Advances in Intelligent Systems and Computing 391

Aleksandra Gruca
Agnieszka Brachman
Stanisław Kozielski
Tadeusz Czachórski *Editors*

Man–Machine Interactions 4

4th International Conference on Man–Machine Interactions, ICMMI 2015 Kocierz Pass, Poland, October 6–9, 2015

Springer

Advances in Intelligent Systems and Computing

Volume 391

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

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

Aleksandra Gruca · Agnieszka Brachman
Stanisław Kozielski · Tadeusz Czachórski
Editors

Man–Machine Interactions 4

4th International Conference on
Man–Machine Interactions, ICMMI 2015
Kocierz Pass, Poland, October 6–9, 2015

Springer

Editors
Aleksandra Gruca
Institute of Informatics
Silesian University of Technology
Gliwice
Poland

Agnieszka Brachman
Institute of Informatics
Silesian University of Technology
Gliwice
Poland

Stanisław Kozielski
Institute of Informatics
Silesian University of Technology
Gliwice
Poland

Tadeusz Czachórski
Institute of Theoretical and Applied Informatics
Polish Academy of Sciences
Gliwice
Poland

and

Institute of Informatics
Silesian University of Technology
Gliwice
Poland

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-23436-6 ISBN 978-3-319-23437-3 (eBook)
DOI 10.1007/978-3-319-23437-3

Library of Congress Control Number: 2015948771

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer International Publishing AG Switzerland is part of Springer Science+Business Media (www.springer.com)

Our technology, our machines, is part of our humanity. We created them to extend ourselves, and that is what is unique about human beings.

Ray Kurzweil

Preface

This volume contains the proceedings of the 4th International Conference on Man–Machine Interactions, ICMMI 2015 which was held at Kocierz Pass, Poland, during October 6–9, 2015. The series of ICMMI conferences is organised biennially since 2009 by Institute of Informatics at the Silesian University of Technology and Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Gliwice, Poland. The first ICMMI was dedicated to the memory of Prof. Adam Mrózek, a distinguished scientist in the area of decision support systems in industrial applications and since 2009 the conference became a permanent fixture in the calendar of scientific meetings, bringing together academic and industrial researchers interested in all aspects of theory and practice of man–machine interactions.

Since the beginning, ICMMI become an international forum providing a unique opportunity to network and find potential collaborators, discuss and exchange innovative ideas, share information and knowledge. This year, ICMMI was organised under the technical co-sponsorship of the IEEE Poland Section ensuring high-quality technical programme of the conference. The interdisciplinary field of man–machine interaction covers the whole spectrum of technological aspects involved in interactions and communications between human users and machines. The discipline relies on novel techniques such as computer vision, speech technology, pattern recognition, expert systems, robotics and biomedical sensor applications. In this volume 6 invited and 54 contributed papers presenting a broad range of topics have been divided into the following sections: human–computer interfaces, robot control, embedded and navigation systems, bio-data analysis and mining, biomedical signal processing, image and motion data processing, decision support and expert systems, pattern recognition, fuzzy systems, algorithms and optimisation, computer networks and mobile technologies, and data management systems.

ICMMI 2015 conference attracted 172 authors from 15 different countries across the world. The review process was conducted by the Programme Committee members with the help of external reviewers. Each paper was subjected to at least two independent reviews. After the peer-review process, 54 high quality papers

were selected for publication in ICMMI 2015 proceedings. Here, we would like to thank all Programme Committee members and associated reviewers for their highly professional work, effort in reviewing papers and valuable comments.

ICMMI 2015 conference had a distinguished panel of keynote speakers. We would like to express our gratitude to Antonis Argyros, Juan Luis Fernández–Marínez, Karmeshu, Hanna Klaudel, Manoj Sharma and Sethu Vijayakumar who agreed to deliver keynote talks and invited papers.

We appreciate the help of the editorial staff at Springer Verlag, in particular we would like to thank Janusz Kacprzyk, the editor of the series, Holger Schape and Thomas Ditzinger who supported the publication of these proceedings in AISC series. We also wish to thank all those who contributed to the organisation of this conference, in particular we are grateful to the members of the Organising Committee for their hard work and efforts in making this conference successful.

Finally, we would like to thank all the authors and participants who contributed to the success of this event. We believe that ICMMI 2015 delivered a high-quality, informative and inspiring scientific programme. We hope that this volume provides an innovative and useful contribution into further research and developments in the field of man–machine interactions.

October 2015

Aleksandra Gruca
Agnieszka Brachman
Stanisław Kozielski
Tadeusz Czachórski

Organization

ICMMI 2015 is organised by the Institute of Informatics, Silesian University of Technology, Gliwice, Poland and the Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Gliwice, Poland.

Conference Chair

Stanisław Kozielski, Silesian University of Technology, Poland

Programme Committee

Ajith Abraham, Machine Intelligence Research Labs, USA
Antonis Argyros, University of Crete, Greece
Gennady Agre, Institute of Information and Communication Technologies, Bulgaria
Francisco Martinez Alvarez, Pablo de Olavide University of Seville, Spain
Annalisa Appice, University of Bari, Italy
Maria Martinez Ballesteros, University of Seville, Spain
Haider Banka, Indian School of Mines, India
Christoph Beierle, University of Hagen, Germany
Orlando Belo, University of Minho, Portugal
Petr Berka, University of Economics in Prague, Czech Republic
Agnieszka Brachman, Silesian University of Technology, Poland
Snehashish Chakraverty, National Institute of Technology Rourkela, India
Bidyut Baran Chaudhuri, Indian Statistical Institute, India
Santanu Chaudhury, Indian Institute of Technology, India
Davide Ciucci, University of Milano Bicocca, Italy
Eliseo Clementini, University of L'Aquila, Italy

Gualberto Asencio Cortes, Pablo de Olavide University of Seville, Spain
Tadeusz Czachórski, Institute of Theoretical and Applied Informatics, PAS, Poland
Claudia d'Amato, University of Bari, Italy
Jeroen de Bruin, Medical University of Vienna, Austria
Kamil Dimililer, Near East University, Cyprus
Dejing Dou, University of Oregon, USA
Antonio Dourado, University of Coimbra, Portugal
Ivo Duntsch, Brock University, Canada
Floriana Esposito, University of Bari, Italy
Erol Gelenbe, Imperial College, UK
William Grosky, University of Michigan, USA
Krzysztof Grochla, Institute of Theoretical and Applied Informatics, PAS, Poland
Aleksandra Gruca, Silesian University of Technology, Poland
Concettina Guerra, Georgia Institute of Technology, USA
Anindya Halder, North-Eastern Hill University Shillong, India
Katarzyna Harezlak, Silesian University of Technology, Poland
Gerhard Heyer, University of Leipzig, Germany
Alfred Inselberg, Tel Aviv University, Israel
Pedro Isaias, Universidade Aberta, Portugal
Richard Jensen, Aberystwyth University, UK
Martti Juhola, University of Tampere, Finland
Janusz Kacprzyk, Polish Academy of Sciences, Poland
Dimitrios Karras, Chalkis Institute of Technology, Greece
Gabriele Kern-Isberner, Technical University of Dortmund, Germany
Petia Koprinkova-Hristova, Bulgarian Academy of Science, Bulgaria
Józef Korbicz, University of Zielona Gora, Poland
Jacek Koronacki, Institute of Computer Science, PAS, Poland
Markus Koskela, Aalto University School of Science, Finland
Stanisław Kozielski, Silesian University of Technology, Poland
Adam Krzyzak, Concordia University, Canada
Amaury Lendasse, Helsinki University of Technology, Finland
Jacek M. Leski, Silesian University of Technology, Poland
Haim Levkowitz, University of Massachusetts Lowell, USA
Yannis Manalopoulos, Aristotle University, Greece
Duoqian Miao, Tongji University, PR China
Evsey Morozov, Karelian Research Centre, Russia
Manuel Ojeda-Aciego, University of Malaga, Spain
Witold Pedrycz, University of Alberta, Canada
Petra Perner, Institute of Computer Vision and Applied Computer Sciences, Germany
Ioannis Pitas, Aristotle University of Thessaloniki, Greece
Joanna Polanska, Silesian University of Technology, Poland
Andrzej Polanski, Silesian University of Technology, Poland
Lech Polkowski, Polish-Japanese Academy of Information Technology, Poland
Arun K. Pujari, University of Hyderabad, India

Sheela Ramanna, University of Manitoba, Canada
Zbigniew W. Ras, University of North Carolina, USA
Jan Rauch, University of Economics in Prague, Czech Republic
Indrajit Saha, University of Wroclaw, Poland
Gerald Schaefer, Loughborough University, UK
Ute Schmid, University of Bamberg, Germany
Rainer Schmidt, University of Rostock, Germany
Kaoru Shimada, Fukuoka Dental College, Japan
Andrzej Skowron, Warsaw University, Poland
Bogdan Smolka, Silesian University of Technology, Poland
Paolo Soda, University Campus Bio-Medico of Rome, Italy
Urszula Stańczyk, Silesian University of Technology, Poland
Andrzej Świerniak, Silesian University of Technology, Poland
Tamas Sziranyi, Budapest University of Technology and Economics, Hungary
Doina Tatar, "Babes-Bolyai" University of Cluj-Napoca, Romania
Ewaryst Tkacz, Silesian University of Technology, Poland
Li-Shiang Tsay, North Carolina A&T State University, USA
Dan Tufis, Research Institute for Artificial Intelligence, Mihai Draganescu, Romania
Brijesh Verma, CQ University, Australia
Krzysztof Walkowiak, Wroclaw University of Technology, Poland
Guoyin Wang, Chongqing University of Posts and Telecommunications, China
Konrad Wojciechowski, Silesian University of Technology, Poland
Michał Woźniak, Wroclaw University of Technology, Poland
Tulay Yildirim, Yildiz Technical University, Turkey

Additional Reviewers

Paweł Foszner
Adam Gudyś
Paweł Kasprowski
Michał Kawulok
Michał Kozielski
Marcin Michalak
Jakub Nalepa
Adam Schmidt
Marek Sikora
Krzysztof Siminski
Przemysław Skurowski
Robert Wójcicki

Organising Committee

Aleksandra Gruca (Chair)
Agnieszka Brachman
Agnieszka Danek
Paweł Foszner
Adam Gudyś
Katarzyna Harezlak

Sponsoring Institutions

Technical co-sponsorship of IEEE Poland Section
IEEE http://www.ieee.org/index.html
IEEE Poland Section http://www.ieee.pl/
IEEE Membership http://www.ieee.org/membership_services/index.html
http://www.ieee.pl/?q=node/51
IEEE Poland Section Chapters http://www.ieee.pl/?q=node/41

Contents

Part I Invited Papers

Modelling and Analysing Mixed Reality Applications 3
Johan Arcile, Tadeusz Czachórski, Raymond Devillers, Jean-Yves Didier, Hanna Klaudel and Artur Rataj

A Generative Approach to Tracking Hands and Their Interaction with Objects 19
Nikolaos Kyriazis, Iason Oikonomidis, Paschalis Panteleris, Damien Michel, Ammar Qammaz, Alexandros Makris, Konstantinos Tzevanidis, Petros Douvantzis, Konstantinos Roditakis and Antonis Argyros

Design of Biomedical Robots for the Analysis of Cancer, Neurodegenerative and Rare Diseases. 29
Juan L. Fernández-Martínez, Enrique J. deAndrés-Galiana and Stephen T. Sonis

Generation of Power Law: Maximum Entropy Framework and Superstatistics. 45
Karmeshu, Shachi Sharma and Sanjeev Kumar

Optimal Control of Multi-phase Movements with Learned Dynamics 61
Andreea Radulescu, Jun Nakanishi and Sethu Vijayakumar

Computable Extensions of Advanced Fractional Kinetic Equation and a Class of Levy-Type Probabilities. 77
Manoj Sharma

Part II Human–Computer Interfaces

A Model-Driven Engineering Approach to the Evaluation of a Remote Controller of a Movement Assistant System 93
Anna Derezińska and Karol Redosz

Neural Network and Kalman Filter Use for Improvement of Inertial Distance Determination 103
Piotr Kopniak and Marek Kaminski

Mobile Activity Plan Applications for Behavioral Therapy of Autistic Children 115
Agnieszka Landowska and Michal Smiatacz

Usability Tests Supporting the Design of Ergonomic Graphical User Interface in Decision-Support Systems for Criminal Analysis 127
Aleksandra Sadowska and Kamil Piętak

Evaluating Informational Characteristics of Oculomotor System and Human–Computer Interfaces 139
Raimondas Zemblys

Part III Robot Control, Embedded and Navigation Systems

On Control of Human Arm Switched Dynamics 151
Artur Babiarz

Incorporating Static Environment Elements into the EKF-Based Visual SLAM 161
Adam Schmidt

Prediction-Based Perspective Warping of Feature Template for Improved Visual SLAM Accuracy 169
Adam Schmidt

Interpolation Method of 3D Position Errors Decreasing in the System of Two Cameras 179
Tadeusz Szkodny

Part IV Bio-Data Analysis and Mining

Parameter Estimation in Systems Biology Models by Using Extended Kalman Filter 195
Michal Capinski and Andrzej Polanski

Nucleotide Composition Based Measurement Bias in High Throughput Gene Expression Studies 205
Roman Jaksik, Wojciech Bensz and Jarosław Smieja

Application of a Morphological Similarity Measure to the Analysis of Shell Morphogenesis in Foraminifera 215
Maciej Komosinski, Agnieszka Mensfelt, Paweł Topa and Jarosław Tyszka

The Resection Mechanism Promotes Cell Survival After Exposure to IR 225
Monika Kurpas, Katarzyna Jonak and Krzysztof Puszynski

Integrative Construction of Gene Signatures Based on Fusion of Expression and Ontology Information 237
Wojciech Łabaj and Andrzej Polanski

Automatic PDF Files Based Information Retrieval System with Section Selection and Key Terms Aggregation Rules 251
Rafal Lancucki and Andrzej Polanski

Influence of Introduction of Mitosis-Like Processes into Mathematical-Simulation Model of Protocells in RNA World 259
Dariusz Myszor

eVolutus: A Configurable Platform Designed for Ecological and Evolutionary Experiments Tested on Foraminifera 269
Paweł Topa, Maciej Komosinski, Maciej Bassara and Jarosław Tyszka

Part V Biomedical Signal Processing

Real-Time Detection and Filtering of Eye Blink Related Artifacts for Brain-Computer Interface Applications 281
Bartosz Binias, Henryk Palus and Krzysztof Jaskot

Application of Dimensionality Reduction Methods for Eye Movement Data Classification 291
Aleksandra Gruca, Katarzyna Harezlak and Pawel Kasprowski

Dynamic Time Warping Based on Modified Alignment Costs for Evoked Potentials Averaging 305
Marian Kotas, Jacek M. Leski and Tomasz Moroń

Principal Component Analysis and Dynamic Time-Warping in Subbands for ECG Reconstruction 315
Tomasz Moroń, Marian Kotas and Jacek M. Leski

Part VI Image and Motion Data Processing

Evaluating of Selected Systems for Colorimetric Calibration of LCD Monitors 329
Artur Bal, Andrzej Kordecki, Henryk Palus and Mariusz Frąckiewicz

Optical Flow Methods Comparison for Video FPS Increase 341
Jan Garus and Tomasz Gąciarz

Towards the Automatic Definition of the Objective Function for Model-Based 3D Hand Tracking 353
Konstantinos Paliouras and Antonis A. Argyros

Optimizing Orthonormal Basis Bilinear Spatiotemporal Representation for Motion Data 365
Przemysław Skurowski, Jolanta Socała and Konrad Wojciechowski

Evaluation of Improvement in Orientation Estimation Through the Use of the Linear Acceleration Estimation in the Body Model 377
Agnieszka Szczęsna, Przemysław Pruszowski, Janusz Słupik, Damian Pęszor and Andrzej Polański

Part VII Decision Support and Expert Systems

Data Cleansing Using Clustering 391
Petr Berka

Estimation of the Joint Spectral Radius 401
Adam Czornik, Piotr Jurgaś and Michał Niezabitowski

AspectAnalyzer—Distributed System for Bi-clustering Analysis 411
Pawel Foszner and Andrzej Polański

Algorithm for Finding Zero Factor Free Rules 421
Grete Lind and Rein Kuusik

Diagnostic Model for Longwall Conveyor Engines 437
Marcin Michalak, Beata Sikora and Jurand Sobczyk

Supporting the Forecast of Snow Avalanches in the Canton of Glarus in Eastern Switzerland: A Case Study 449
Sibylle Möhle and Christoph Beierle

Geospatial Data Integration for Criminal Analysis 461
Kamil Piętak, Jacek Dajda, Michał Wysokiński, Michał Idzik and Łukasz Leśniak

Multivariate Approach to Modularization of the Rule Knowledge Bases 473
Roman Simiński

Part VIII Pattern Recognition

Practical Verification of Radio Communication Parameters for Object Localization Module 487
Karol Budniak, Krzysztof Tokarz and Damian Grzechca

Perturbation Mappings in Polynomiography 499
Krzysztof Gdawiec

PCA Based Hierarchical Clustering with Planar Segments as Prototypes and Maximum Density Linkage 507
Jacek M. Leski, Marian Kotas and Tomasz Moroń

Feature Thresholding in Generalized Approximation Spaces 517
Dariusz Małyszko

Progressive Reduction of Meshes with Arbitrary Selected Points 525
Krzysztof Skabek, Dariusz Pojda and Ryszard Winiarczyk

The Class Imbalance Problem in Construction of Training Datasets for Authorship Attribution 535
Urszula Stańczyk

Part IX Fuzzy Systems

Approximate Reasoning and Fuzzy Evaluation in Code Compliance Checking 551
Ewa Grabska, Andrzej Łachwa and Grażyna Ślusarczyk

Classification Based on Incremental Fuzzy $(1 + p)$-Means Clustering 563
Michal Jezewski, Jacek M. Leski and Robert Czabanski

Imputation of Missing Values by Inversion of Fuzzy Neuro-System . . . 573
Krzysztof Siminski

Memetic Neuro-Fuzzy System with Big-Bang-Big-Crunch Optimisation 583
Krzysztof Siminski

Part X Algorithms and Optimisation

Statistical Methods of Natural Language Processing on GPU 595
Dariusz Banasiak

Profitability Analysis of PV Installation in Combination with Different Time-of-Use Strategies in Poland 605
Agnieszka Brachman and Robert Wojcicki

Bees Algorithm for the Quadratic Assignment Problem on CUDA Platform 615
Wojciech Chmiel and Piotr Szwed

Grammatical Inference in the Discovery of Generating Functions 627
Wojciech Wieczorek and Arkadiusz Nowakowski

Optimization of Decision Rules Relative to Coverage—Comparison of Greedy and Modified Dynamic Programming Approaches 639
Beata Zielosko

Part XI Computer Networks and Mobile Technologies

Characteristic of User Generated Load in Mobile Gaming Environment 653
Krzysztof Grochla, Wojciech Borczyk, Maciej Rostanski and Rafal Koffer

Using Kalman Filters on GPS Tracks 663
Krzysztof Grochla and Konrad Połys

Stability Analysis and Simulation of a State-Dependent Transmission Rate System 673
Evsey Morozov, Lyubov Potakhina and Alexander Rumyantsev

Part XII Data Management Systems

Independent Data Partitioning in Oracle Databases for LOB Structures 687
Lukasz Wycislik

Hybrid Column/Row-Oriented DBMS 697
Małgorzata Bach and Aleksandra Werner

Author Index 709

Part I
Invited Papers

Modelling and Analysing Mixed Reality Applications

Johan Arcile, Tadeusz Czachórski, Raymond Devillers, Jean-Yves Didier, Hanna Klaudel and Artur Rataj

Abstract Mixed reality systems overlay real data with virtual information in order to assist users in their current task. They generally combine several hardware components operating at different time scales, and software that has to cope with these timing constraints. MIRELA, for MIxed REality LAnguage, is a framework aimed at modelling, analysing and implementing systems composed of sensors, processing units, shared memories and rendering loops, communicating in a well-defined manner and submitted to timing constraints. The framework is composed of (i) a language allowing a high level, and partially abstract, specification of a concurrent real–time system, (ii) the corresponding semantics, which defines the translation of the system to concrete networks of timed automata, (iii) a methodology for analysing various real-time properties, and (iv) an implementation strategy. We present here a summary of several of our papers about this framework, as well as some recent extensions concerning probability and non–deterministic choices.

Keywords Mixed reality · Timed automata · Deadlocks · Temporal properties

J. Arcile · J.-Y. Didier · H. Klaudel (✉)
Laboratoire IBISC, Université d'Evry-Val d'Essonne, Évry, France
e-mail: hanna.klaudel@ibisc.fr

J. Arcile
e-mail: johan.arcile@ens.univ-evry.fr

J.-Y. Didier
e-mail: jean-yves.didier@ibisc.fr

R. Devillers
Département d'Informatique, Université Libre de Bruxelles, Brussels, Belgium
e-mail: rdevil@ulb.ac.be

T. Czachórski · A. Rataj
Institute of Theoretical and Applied Computer Science, Gliwice, Poland
e-mail: tadek@iitis.pl

A. Rataj
e-mail: arturrataj@gmail.com

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_1

1 Introduction

The primary goal of a mixed reality (MR) system is to produce an environment where virtual and digital objects coexist and interact in real time. In order to get the global environment and its virtual or physical objects we need specific data, for which we shall use sensors (like cameras, microphones, haptic arms…). But gathering data is not sufficient as we want to see the result in our mixed environment; we then implement a rendering loop that will read the data and express the result in some way that a human can interpret (using senses like sight, hearing, touch). To communicate between those two types of components (sensors and renderers), shared memory units store the data, and processing units process the data received from sensors or processing units, and write them into shared memories or other processing units.

Since a few years, the MIRELA framework [7–9, 14] (for MIxed REality LAnguage) is developed aiming at supporting the development process of applications made of components which have to react within a fixed delay when some events occur inside or outside the considered area. This is the case in mixed reality applications which are evolving in an environment full of devices that compute and communicate with their surrounding context [6]. In such a context, it is difficult to keep control of the end-to-end latency and to minimise it. Classically, mixed reality software frameworks do not rely on formal methods in order to validate the behaviour of the developed applications. Some of them emphasise the use of formal descriptions of components inside applications in order to enforce a modular decomposition, possibly with tool chains to produce the final application [13, 17], and ease future extensions [15] or substitutions of one module by another, like InTML [10, 11]. Such frameworks do not deal with software failure issues related to time. On the contrary, this is the main focus of the MIRELA framework which proposes to use formal methods and automatic tools to analyse and understand potential issues, especially related to time, performance, and various kinds of bad behaviours such as deadlocks, starvations or unbounded waitings.

A typical modelling using MIRELA consists of the following stages. In the first phase, a formal specification of the system is given, in the form of a network of automata, defined using a high level description [8, 9]. Then, depending on the applied approximation of the modelled system, and also on the properties we want to check, we decide how to transform the components. This may include theoretical issues, like the generation of bounding variant systems [7], which contain under– and over–approximating timed automata [1], and practical issues, like which model checker to target, depending on its capabilities. Finally, an implementation skeleton can be produced, e.g. in the form of a looping controller, which has a simple physical realisation, while certain properties of the original network are still met, which may be checked on to the chosen bounding variant systems.

2 MIRELA Framework

Originally, the semantics of a MIRELA specification has been defined and implemented in UPPAAL [18] as a set of timed automata [1–3, 19]. More precisely, we used a subclass called Timed Automata with Synchronised Tasks (TASTs) in order to cope with implementability issues (see [7] for details). TASTs are networks of timed automata, which communicate via urgent communication channels in the producer–consumer manner, and optionally contain wait locations, where the wait time $t \in [\min, \max]$ is non-deterministic. The communication dependencies between the automata form a directed and connected graph. There are also some additional constraints, so that the resulting automaton is non–Zeno, i.e. infinite histories taking no time or a finite time are excluded. For a complete description of TASTs see [9].

If the urgent channels are not available, like in the case of PRISM timed automata, a corresponding transformation is possible, which emulates the urgent channels [5].

Due to the verbosity of TASTs, MIRELA has its own, terse syntax, which can be automatically compiled into TASTs, but then also to representations adequate for other model checking tools than UPPAAL. The current MIRELA specification is presented in detail in [9]. Here we will give a summary.

A network of components is defined as a list of declarations of the form:

$$\mathit{SpecName}: \quad id = \mathit{Comp} \rightarrow \mathit{TList}; \ldots; id = \mathit{Comp} \rightarrow \mathit{TList}.$$

Each declaration $\mathit{Comp} \rightarrow \mathit{TList}$ defines a component *Comp* and its target list of components *TList*, which is an optional (comma separated) list of identifiers indicating to which (target) components information is sent, and in which order. Each component also indicates from which (source) components data are expected. A target t of a component c must have c as a source, but it is not required that a source s of a component c has c as an explicit target: missing targets will be implicitly added at the end of the target list, Any of the components, after an optional initialisation, loops infinitely. Delays within the components use clocks, and clocks are never shared between components.

Here is a list of some of the standard components:

- two kinds of sensors that acquire data from outside and send it to processing units or memories:
  - Periodic($\mathit{min\_start}, \mathit{max\_start}$)[$\mathit{min}, \mathit{max}$] starts with a one–off delay $\langle \mathit{min\_start}, \mathit{max\_start} \rangle$, then loops infinitely, each cycle lasting within $\langle \mathit{min}, \mathit{max} \rangle$;
  - Aperiodic($\mathit{min\_event}$) ascertains that the loop has a minimal delay of $\mathit{min\_event}$, but no maximal delay is specified;
- three kinds of processing units that process data coming from possibly several different inputs (they can be combined in an acyclic hierarchy or form loops):

$Ex1$:
$S1 = \mathsf{Periodic}(50, 75)[75, 100];$
$S2 = \mathsf{Periodic}(200, 300)[350, 400] \rightarrow (F2, F1);$
$S3 = \mathsf{Periodic}(200, 300)[350, 400] \rightarrow (F2, B);$
$F1 = \mathsf{First}(S1, S2[50, 75]);$
$F2 = \mathsf{First}(S2, S3[25, 50]);$
$B = \mathsf{Both}(S3, F2)[25, 50];$
$M = \mathsf{Memory}(F1[25, 50], B[25, 50]);$
$R = \mathsf{Rendering}(50, 75)(M[25, 50]).$

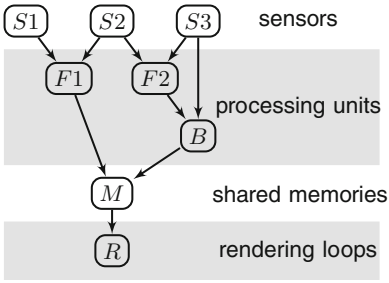


Fig. 1 Specification and information flow representation of our running example

- First(i_1[*min*,*max*], i_2[*min*,*max*], ...) which may have one or more inputs $i_1, i_2, \ldots$ and starts processing when data are received from one of them; the order is irrelevant; the loop delay depends on the input, as seen in the declaration, but a same interval may be distributed on many inputs;
- Both(i_1, i_2)[*min*,*max*] which has exactly two inputs i_1, i_2 and starts processing when both input data are present; the loop delay is $\langle min, max \rangle$;
- Priority(i_m[*min*,*max*], i_s[*min*,*max*]), which has two inputs, master i_m and slave i_s, and starts processing when the master input is ready, possibly using the slave input if it is available before the master one; the delay specified at i_m is realised if the slave was not available; otherwise the delay at i_s is used.

- a shared memory Memory(i_1[*min*,*max*], i_2[*min*,*max*], ...), with reads and writes locked by a common mutex, the write time depends on the input, as seen in the declaration;
- a rendering component Rendering(*min_rg*, *max_rg*)(i_m[*min*,*max*]) is a loop, which consists in reading a memory within a delay specified at i_m, then processes the read data within

This is illustrated with a running example, presented in Fig. 1 along with the corresponding flow of information. The corresponding TAST representation is depicted in Fig. 2.

3 Analysis

We will discuss examples of proving various properties using Mirela, in particular related to bad behaviours, like various kinds of deadlocks and deadlock-like behaviours, which can be distinguished for timed automata.

A complete blocking occurs if a state is reached where nothing may happen: no location change is allowed (because no arc with a true guard is available, or the only ones available lead to locations with a non-valid invariant) and the time is blocked (because the invariant of the present location is made false by time passing).

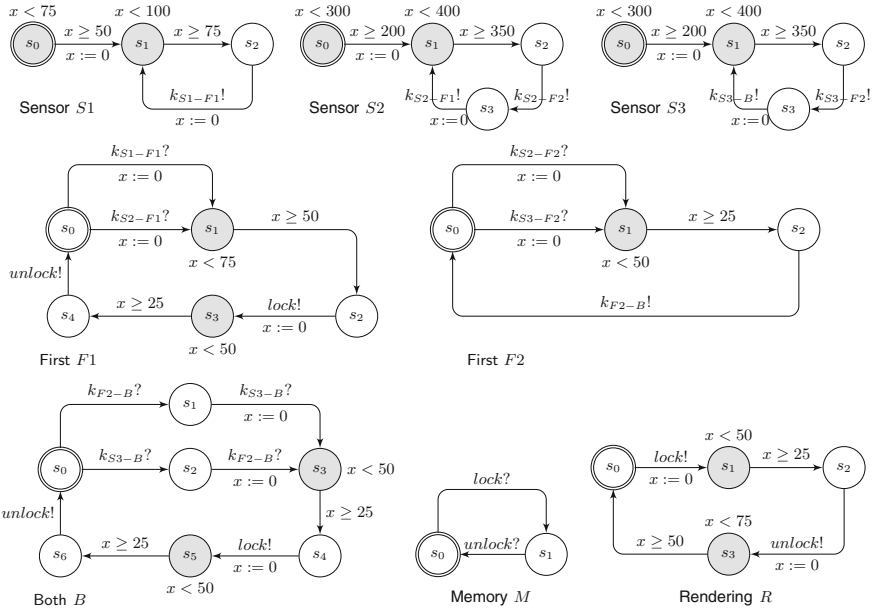


Fig. 2 TAST representation of our running example

A (global) deadlock occurs when only time passing is ever allowed: no location change is nor will be available. A strong Zeno situation occurs when infinitely many location changes may be done without time passing. A weak Zeno situation occurs if infinitely many location changes may occur in a finite time delay. A weaker but potentially unpleasant situation occurs when a location change is available after some time, but this waiting time is unbounded. Those situations are usually considered highly harmful since an actually implemented system cannot meet these theoretical requirements.

For a network of timed automata (hence for TAST systems), a local deadlock occurs if, from some point, no location change is available for some component(s) while other components may evolve normally. A local unbounded waiting occurs if it is certain that a component will evolve, but the time before the component leaves its present state is unbounded. A starvation occurs if a component may be indefinitely blocked in some state (while not being deadlocked); this is different from the previous case, since here the time during which the component is blocked in the present state may really be infinite, and not simply unbounded. Notice however that starvations are not always to be avoided, for instance if the component corresponds to a failure handling.

3.1 Deadlock Detection and Graph Analysis

While MIRELA has translation mechanisms allowing to use model checking tools, we may also take advantage of the specific features of the MIRELA systems, and in particular of the graph structure of the systems (see for example Fig. 1—right).

First, one may observe that no (strong or weak) Zeno behaviour may happen since each loop in a component either contains a reset on some clock x and an arc with a guard $x \geq e$ (with $e > 0$), so that it is impossible to follow it in a null time, and it may be followed only a finite number of times in a finite delay, or contains arcs with input communications (in a memory cell for instance) and it may only progress indefinitely while communicating with a loop of the previous kind.

Moreover, if the system is well-formed, i.e., for each location with an invariant $x < e'$ each input arc resets x and each output arc has a guard $x \geq e$ with $e < e'$ (which is the case for MIRELA systems), it may not block the time.

A global deadlock (hence a complete blocking) may not occur in a complete system, i.e., having at least one memory unit and an associated rendering loop, since the rendering may indefinitely progress while accessing the memory (but starvation may occur if the memory is continually used by other units, and no fairness strategy is applied).

On the contrary local deadlocks may occur. They are intimately connected to the fact that processing units alternate two very distinct phases: first, some signals are received (reception phase), then some signals are emitted (emission phase) in a row (together with some synchronisations with memory units), and then the reception phase is resumed. Typically, in case of a cycle of processing units, it may happen that a processing unit in reception phase waits for a signal, which depends (directly or not) on its own emission, or symmetrically. There are also deadlocks of a mixed nature, combining components in emission and reception phases, that do not need involving any cycle of communicating units, contrary to what happened in the *emission* or *reception* cases. This is due to the fact that, when a component is blocked in its reception phase, the waiting condition corresponds to one or two arcs going in the reverse direction with respect to the flow of information. Hence, a cycle of control may correspond (when there are both emitting and receiving blocked components) to a non cyclic flow of information.

In order to propose guidelines for the detection of local deadlocks, we may observe that a component may only deadlock in a wait location, but never when it waits for a $lock!$, $unlock!$ or $unlock?$. It also means that rendering loops never deadlock, and that memory units may not be the source of a deadlock: a memory unit may only deadlock if it has no rendering loop and all its users are deadlocked while trying to communicate with a non-memory unit.

Definition 1 Let MS be a MIRELA system. An extended system $\overline{MS}$ is a temporal widening of MS if it has the same structure but each (or some) time interval $\mathcal{I}$ is replaced by another one $\overline{\mathcal{I}}$, where $\mathcal{I} \subseteq \overline{\mathcal{I}}$. Symmetrically, an extended system $\underline{MS}$ is a strengthening of MS if it has the same structure but each (or some) interval $\mathcal{I}$ is replaced by another one $\underline{\mathcal{I}}$, where $\underline{\mathcal{I}} \subseteq \mathcal{I}$. **X** 1

Note that, in particular, *MS* is a temporal widening and a strengthening of itself.

Proposition 1 *Let MS be a* MIRELA *system and* $\overline{MS}$ *be any of its temporal widenings. If a component does not deadlock at some location w in* $\overline{MS}$*, the same is true in MS. On the contrary, if* $\underline{MS}$ *is any strengthening of MS and a component deadlocks at some location w in* $\underline{MS}$*, the same is true in MS.* □

This result may be precious because the model checking complexity of timed systems depends of course on the complexity of the system, and on the formulas to be verified, but also on the gcd of the various constants occurring in the timing constraints, and on the various scales of the time intervals. Now, enlarging or restricting those constraints may considerably increase this gcd, or uniformise the various time intervals. Note that, among the special enlargements that may be considered, some or all the upper bounds of time intervals may be replaced by ∞, and some or all the lower bounds of time intervals may be replaced by 1. In fact, it may be observed that lower bounds may even be replaced by 0: this may introduce Zeno behaviours, but does not kill existing deadlocks. Finally, one may observe that going from a constraint $x < e$ to $x \leq e$ is a form of a temporal widening, and going from $x < e$ to $x \leq e - 1$ is a form of strengthening.

Modifying the time intervals may be used to check the presence or absence of bad behaviours, but it may also be used to get a modified system, easier to model check, while maintaining its realistic aspect and its implementability.

Proposition 2 *Let MS be a* MIRELA *system. If a component deadlocks in MS, then so do all its input components, all its output* Both*, all its master output* Priority*, and all its output* First *units having a unique input. A memory component is never the source of a deadlock, but it may incur a deadlock propagation, if all its user components deadlock (which may not occur if there is a corresponding rendering loop).* □

Note that if a component deadlocks while it is a source of a slave input to a Priority, the latter does not necessarily deadlocks; a similar situation may occur if a First component has many inputs and one (or more, but not all) of them deadlocks, since it may still manage inputs from non-deadlocking units. In both cases, the deadlock does not necessarily extend.

While temporal widening and strengthening do not modify the structure of a system, it is also possible to build modified (in general structurally simplified) systems. For instance, since rendering loops never deadlock, one may drop them. We may even also drop the memory units, since they are never the source of a deadlock.

Definition 2 Let *MS* be a MIRELA system and *C* be one of its sensors or processing units. We shall denote by

- $\widehat{MS}$ the system obtained from *MS* by replacing all the time intervals by $[0, \infty)$, i.e., a form of untimed version of *MS*;
- $\widetilde{MS}$ the system obtained from $\widehat{MS}$ by dropping all its rendering loops;

- $C(\widetilde{MS})$ the part of $\widetilde{MS}$ (i.e., the set of components) which, in the abstract scheme, is connected to C without needing to traverse a memory unit; notice that it comprises the memory units.
- $\mathbf{x}2C(\widetilde{\widetilde{MS}})$ the system $C(\widetilde{MS})$ without its memory units.

Proposition 3 *Let MS be a* MIRELA *system and C a sensor or processing unit in it. Let* $MS' = C(\widetilde{MS})$ *and* $MS'' = C(\widetilde{\widetilde{MS}})$. *Then C deadlocks in* $\widehat{MS}$ *iff it deadlocks in* MS' *iff it deadlocks in* MS''. □

Combining Propositions 1 and 3 allows in some circumstances to detect the absence of local deadlocks while reducing the systems to be considered, both in terms of structure and temporal constraints. And if we add Proposition 2, this may even reduce the problem of detecting the absence of a local deadlock to the detection of the absence of global deadlocks in simplified systems, when the deadlock propagation ensures that a local deadlock extends to a global one. This may be useful since there are efficient algorithms to detect (the absence of) global deadlocks, and special commands to check it (in UPPAAL for instance).

Finally, we may observe that a local deadlock, in the original or temporally widened or strengthened model, may be handled by a model checker with query $\psi_w = \text{EF AG } w$, which checks if there is a situation (EF) where the considered component reaches w while there is no way (AG w) to get out of it: this thus corresponds to a local deadlock. Note that this formula pertains to CTL but uses a nesting of path formulas, so that it is not handled by UPPAAL and its optimised implementation. On the contrary, it is accepted by another model checker, PRISM, for which an automatic translation from MIRELA has also been developed (see Sect. 5). It is also possible to detect which components, at which locations, may be blocked simultaneously (recall that local deadlocks are due to many components blocking each others): if w_1 is a wait location in some component C_1, w_2 is in C_2, …, one may use the same formula ψ_w, but where $w = w_1 \wedge w_2 \wedge \ldots \wedge w_k$, to check if $C_1, C_2, \ldots, C_k$ may be blocked simultaneously, in locations $w_1, w_2, \ldots, w_k$, respectively.

In summary, to detect the presence or absence of deadlocks, we may use a procedure, which first tries to get some information from simplified versions of the given system, using Propositions 3 and 2, and next uses Proposition 1 to try to get information on the remaining undecided wait locations.

3.2 Indefinite Waiting Detection

Even if there is no deadlock, it may happen that a component gets stuck in some location. In our case, there are essentially two sources of such an *indefinite waiting*.

For instance, this may occur with an aperiodic sensor, since no upper bound is specified for the time separating two successive data acquisitions. If we do not want that this propagates to other components, we should in particular avoid to use them in a Both unit, as a master to a Priority, or in a First when there is no other kinds of

input. This may be qualified as an *unbounded waiting*, since the assumption is that the time between data acquisitions is finite, but unbounded.

Another kind of situation occurs if many components compete to communicate and, due to the non-deterministic way choices are performed, some of them never succeed. This then corresponds to a potentially infinite waiting, also termed *starvation*.

In a MIRELA system, this may for instance occur when performing a lock on a Memory unit (note that unlocks may only be performed by the components having succeeded in the lock): it may happen that a component (Rendering loop, Sensor or Processing unit) tries to access a Memory, fails because, when the memory is unlocked, the latter is attributed to another requesting component, and due to an unfortunate choice of the timing constraint (intervals), whenever the memory is unlocked, there are (remaining or new) requesting components and the considered one is never chosen, unfortunately, again and again. This also shows that, if we are concerned by starvations, in general it is not a good idea to consider systems simplified by removing Memory and Rendering units, since these can be essential ingredients for inducing infinite waitings.

This may be avoided by an adequate choice of the timing constraints, or by suitable fairness assumptions (and implementations) but in the latter case, ensuring that the waiting time will be finite does not necessarily imply that this time is (upper) bounded.

Proposition 4 *Let MS be a* MIRELA *system, a component may only incur an indefinite waiting in the initial activity location of an aperiodic sensor, or in a wait location, but never while waiting for an unlock.* □

Proposition 5 *Let MS be a deadlock-free* MIRELA *system, a component incurs an indefinite waiting at a location w iff the CTL formula* $\phi_w = \text{EF EG w}$ *is true.* □

If w is the activity location of an aperiodic sensor, we know that there is an unbounded waiting, and it is not necessary to perform the model checking for that. The other interesting cases correspond to wait locations, from which communications $k!$ or $k?$ only are offered (with $k \neq unlock$).

Since ϕ_w is a nesting of two path formulas, like ψ_w it may not be checked with UPPAAL. However, if we already know that w is reachable, instead of using this nested query, it is possible to use equivalently (up to contraposition) a *leads to* property, for which UPPAAL has an efficient algorithm: $\widetilde{\phi}_w = w{-}{-}{>}\neg w$ means that, if true, after w we shall eventually leave it, i.e., we shall have no deadlock and no indefinite waiting in w. Again, instead of working directly on the original system, one may consider temporally widened and/or strengthened versions of it.

Proposition 6 *Let MS be a* MIRELA *system,* $\overline{MS}$ *be one of its temporal widenings (while avoiding to start an interval from* 0*), and* $\underline{MS}$ *be one of its strengthenings. If a component incurs an indefinite waiting at some location w in MS, the same is true in* $\underline{MS}$*, and if a component incurs an indefinite waiting at some location w in* $\underline{MS}$*, the same is true in MS.* □

Note that, here, we may not drop memory units and/or rendering loops, since the latter may be needed ingredients to cause an indefinite waiting. A procedure to detect an indefinite waiting of a component C at location w may thus be proposed. If an indefinite waiting is found for a temporally widened system, nothing may be inferred in general on the original system; however, it may happen that in some circumstances a closer (manual) analysis of the found indefinite wait reveals that the same situation occurs in the original system: then the procedure may be stopped with a positive answer.

If the system presents deadlock situations (checkable with ψ), we could wonder if it also presents indefinite waitings. If ψ_w is false while ϕ_w is true, clearly we have an indefinite waiting at location w. But if ψ_w is true, it could still happen that the system also presents an indefinite waiting at the same location, but for a different environment than the deadlock. This may be checked by a slightly more elaborate CTL formula: $\rho_w = \text{EF EG } (w \wedge (\text{EF } \neg w))$.

Proposition 7 *Let MS be a* MIRELA *system. It presents an indefinite waiting at some location w iff* ρ_w *is true.* □

3.3 Starvation Viz. Unbounded Waiting Detection

If a system has no aperiodic sensor, it is sure that there is no unbounded waiting and that any found indefinite waiting (hence formula ρ_w) corresponds to a starvation phenomenon. The same is true if there are aperiodic sensors, but it is sure their unbounded delays do not propagate. But otherwise we could want to know if there are pure starvations and/or pure unbounded waitings, even for a same location (but for different environments).

Let us assume the considered specification *MS* presents n aperiodic sensors and let us denote by $a_1, a_2, \ldots, a_n$ their respective initial locations (in the TAST translation), and that there is a possible propagation of unbounded waitings.

To check a starvation in a wait location w, we may use the following property formula: $\sigma_w = \text{EF EG } (w \wedge (\text{EF } \neg w) \wedge (\text{F } \neg a_1) \wedge \cdots \wedge (\text{F } \neg a_n))$ which means it is possible to stay indefinitely in w, but also to escape from it, without needing that an aperiodic sensor (or many of them) indefinitely stays in its activity location. Hence, if true, this means there is a pure starvation in w.

To check an unbounded waiting in the same location (or another one), one may use the formula: $\zeta_w = \text{EF } ((\text{EG } w) \wedge \text{A}((\text{G } w) \Rightarrow (\text{FG } a_1) \vee \cdots \vee (\text{FG } a_n)))$ which means it is is possible to stay indefinitely in w, but not without being stuck in some a_i at some point. If this is true, this thus means we have an unbounded waiting in w.

Unfortunately, those last two formulas belong to CTL* and, while it is known that CTL* is decidable, the corresponding decidability algorithm is extremely intricate, and we do not know any implementation of it.

4 Temporal Properties

Another kind of question that may be asked on such systems concerns the minimal and/or maximal durations taken by components to perform their operations. We may be interested in exact values, or in bounds such as: "is this time greater than n units" or "is it between n and m". For instance, one may ask how much time a component may wait in some location until a *rendez-vous* is performed, one may wonder how much time a component takes to perform its (main) loop, or to go from one location to another one. We may need for that to add a new clock y that is reset when we enter the starting location, and then we check the greatest and smallest values of that clock when reaching the goal location s. Since the greatest value is calculated when we leave s, if we want to know that value when we enter it, a general trick to do it in UPPAAL is to add an *urgent location* s^u before s, i.e., a location where time may not progress (an urgent location is time-freezing), while redirecting the input arcs of s to s^u. The same trick will avoid to consider for the minimal value the initial value 0 when the goal location is the initial one. A typical query to do that with UPPAAL is `sup{`$C.s^u$`}:y` (resp. `inf{`$C.s^u$`}:y`) which determines the supremum (resp. infimum) of y when entering location s in component C.

However, in some complex cases such a method reveals inefficient due to the large number of states of the system, and an abstraction method like the following may be useful. The general idea is to consider iteratively some simplified, temporally widened systems, on which the duration bounds estimation is feasible, and to use the obtained bounds to get better temporal widenings, hence to progressively improve the bounds until either no improvement is possible or the obtained bounds are considered satisfactory. To get such simplified systems, the idea is to cut the original system into parts, with some components in the common boundaries, and no communication between components in the interior of different parts (the communications with the exterior must go through the boundaries). Then, one considers iteratively each part (let us call it $\mathcal{P}$) and we isolate it by replacing each connection between $\mathcal{P}$ and the other parts by an activity location associated with an interval encompassing (hence the temporal widening) the interval in which the actual connection takes place. By analysing the durations in the simplified system for $\mathcal{P}$, we shall get intervals encompassing the durations of the interactions between the other parts and $\mathcal{P}$. When we shall consider another part $\mathcal{P}'$, these intervals will be used for abstracting the interactions between $\mathcal{P}'$ and $\mathcal{P}$.

5 Support for PRISM

Besides the translation of MIRELA specifications into TASTs, in a form adequate to use the UPPAAL model checker, another translation mechanism has been devised to the input format of PRISM [12], another model checking framework, able to analyse probabilistic systems but also non-deterministic ones. In particular, the *digital*

clocks engine of PRISM accepts CTL requests that UPPAAL doesn't. However, the translation is less easy than to TAST, due to various characteristics of the models:

- Discrete clocks: the digital clocks engine uses discrete clocks only (and consequently excludes strict inequalities in the logical formulas). This modifies the semantics of the systems, but it may be considered that continuous time, as used by UPPAAL, is a mathematical artefact and that the true evolutions of digital systems are governed by discrete time devices;
- Communication semantics: in UPPAAL, communications are performed through binary (input/output) synchronisations on some channel k. A synchronisation transition triggers simultaneously exactly one pair of edges $k?$ and $k!$, that are available at the same time in two different components. PRISM implements n-ary synchronisations, where an edge labelled $[k]$ may only occur in simultaneity with edges labelled $[k]$ in all components where they are present. Implementing binary communications in PRISM is easy by demultiplying and renaming channels in such a way that a different synchronous channel $[k]$ is attributed to each pair $k?$ and $k!$ of communication labels. In MIRELA specifications, the only labels we have to worry about are the $lock?$ and $unlock?$ labels in each Memory M and the $lock!$ and $unlock!$ labels in the components that communicate with M;
- Urgent channels: UPPAAL offers a modelling facility by allowing to declare some channels as urgent. Delays must not occur if a synchronisation transition on an urgent channel is enabled. PRISM does not have such a facility and thus it should be "emulated" using a specific construct compliant with PRISM syntax. This was done by duplicating some locations and by introducing adequate guards [5].

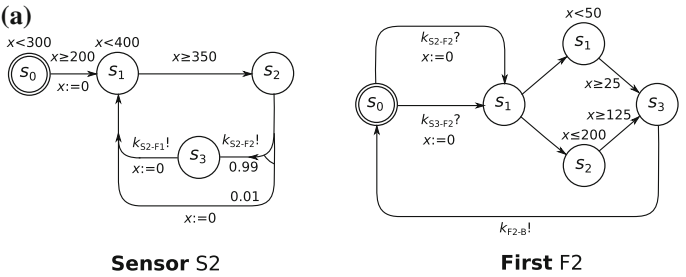
This was realised by adding a dedicated plugin to the translator J2TADD [16] aiming at producing specifications for the PRISM probabilistic model checker.

This may also be a gateway to add probabilities to the framework, hence to turn non-deterministic MIRELA models into stochastic ones. This may be done at two different levels. First, one may replace the time intervals min–max by (continuous or discrete) probability distributions. Next, when many synchronisations are offered to a component at some point (which may happen with a non-null probability when discrete distributions are used), a random drawing mechanism must be defined to perform the choice. It is also possible to mix probabilistic and nondeterministic features, and to allow for alternative paths.

Let us discuss a possible example of such an extension, by modifying two components in Ex1, as illustrated in Fig. 3a. We would thus have a sensor S2 which skips data transmission at a given ratio. Let r be the skipping probability: this could be specified by an optional modifier `drop=`r after the list of output components of the sensor. We also have a processing unit F2, which undergoes, in an unspecified manner, occasional time–consuming clean–up computations. This could be specified by a $+$ operator between the various execution possibilities.

In effect, in the resulting system $Ex1'$, whose specification is shown in Fig. 3b, the sensor S2 now skips every 100th data transmission on average, and the processing unit F2 occasionally, in an unspecified manner, undergoes an additional clean–up computations, which last from 100 to 150 ms.

(a)



(b)

$Ex1'$:

$$S1 = \mathsf{Periodic}(50, 75)[75, 100];$$
$$S2 = \mathsf{Periodic}(200, 300)[350, 400] \rightarrow (F2, F1)\ \mathrm{drop} = 0.01;$$
$$S3 = \mathsf{Periodic}(200, 300)[350, 400] \rightarrow (F2, B);$$
$$F1 = \mathsf{First}(S1, S2[50, 75]);$$
$$F2 = \mathsf{First}(S2, S3[25, 50] + [125, 200]);$$
$$B = \mathsf{Both}(S3, F2)[25, 50];$$
$$M = \mathsf{Memory}(F1[25, 50], B[25, 50]);$$
$$R = \mathsf{Rendering}(50, 75)(M[25, 50]).$$

Fig. 3 **a** An example of Mirela components, which would replace the respective automata in Fig. 2, and **b** the corresponding specification of the modified example

An example of a probabilistic property, against which we could check *Ex1′*, using Prism's parametric model checking, and replacing the constant `drop`=0.01 with and undefined value `drop`=*R*, might be "how the value of *R* affects the maximum probability that a deadlock will occur in the first 10 time units". If the result would be a function with *R* absent or reducible, it would mean that the drop does not affect the probability of the deadlock.

6 Implementation Strategy

Given a verified MIRELA specification *MS*, which satisfies some properties considered important, the idea is to use this specification for producing an implementation aiming at preserving those properties. The approach from [7] considers implementation prototypes, which take the form of a looping controller $\mathcal{C}_{MS,\Delta}$, obtained from the TAST representation of *MS* and parameterised with a well-chosen sampling period Δ. Such a controller may execute zero or several actions in the same period Δ. Obviously, there are semantic differences between the implementation and the specification, coming mostly from the interpretation of the continuous clock values in the sampled world of the implementation and the immediate reaction of the system when a synchronisation becomes possible. For example, one may easily observe that even

if the original specification MS does not reach some error state, the controller $\mathcal{C}_{MS,\Delta}$ may reach it because the sampling allows to evaluate transition conditions potentially larger than it was the case in MS. The approach then proposes an over-approximating model of the implementation in order to check if the essential properties of the original specification are still satisfied by the implementation. The new model "covers" the evolutions of the controller $\mathcal{C}_{MS,\Delta}$, hence "sandwiching" the implementation between the original specification and this auxiliary model. This model, $\overline{MS}$, is very similar to MS, but with relaxed timing constraints, and essentially allows to check if the safety properties of the specification are preserved by the implementation.

7 Conclusion and Future Work

The analysis techniques described above have been checked on various realistic examples, with satisfactory results [4, 5]. For instance, when analysing our running example, the evaluation of $\tilde{\phi}_w$ with UPPAAL takes a few seconds for the various interesting locations w, while the evaluation of ψ_w and ρ_w with PRISM takes a hundred of seconds. Similarly, obtaining bounds for the looping time of each component took a few seconds, with UPPAAL and the abstraction technique.

In the future, we plan to extend the MIRELA specification by approximate probabilistic distributions, and also by explicit state variables.

Acknowledgments This work has been partly supported by French ANR project SYNBIOTIC and Polish-French project POLONIUM.

References

1. Alur, R., Dill, D.L.: Automata for modeling real-time systems. Automata, Languages and Programming. LNCS, vol. 443, pp. 322–335. Springer, Berlin (1990)
2. Alur, R., Dill, D.L.: The theory of timed automata. In: de Bakker, J.W., Huizing, C., de Roever, W.P., Rozenberg, G. (eds.) Real Time: Theory in Practice. LNCS, vol. 600, pp. 45–73. Springer, Berlin (1991)
3. Alur, R., Dill, D.L.: A theory of timed automata. Theor. Comput. Sci. **126**(2), 183–235 (1994)
4. Arcile, J., Didier, J.Y., Klaudel, H., Devillers, R., Rataj, A.: Analysis of real-time properties of mixed reality applications. Submitted paper (2015)
5. Arcile, J., Didier, J.Y., Klaudel, H., Devillers, R., Rataj, A.: Indefinite Waitings in MIRELA systems. In: ESSS 2015. pp. 5–18. Oslo, Norway (2015)
6. Chouiten, M., Domingues, C., Didier, J.Y., Otmane, S., Mallem, M.: Distributed mixed reality for remote underwater telerobotics exploration. In: VRIC 2012. pp. 1:1–1:6. Laval, France (2012)
7. Devillers, R., Didier, J.Y., Klaudel, H.: Implementing timed automata specifications: the "Sandwich" approach. In: ACSD 2013. pp. 226–235. Barcelona, Spain (2013)
8. Didier, J.Y., Djafri, B., Klaudel, H.: The MIRELA framework: modeling and analyzing mixed reality applications using timed automata. J. Virtual Real. Broadcast. **6**(1) (2009)
9. Didier, J.Y., Klaudel, H., Moine, M., Devillers, R.: An improved approach to build safer mixed reality systems by analysing time constraints. In: JVRC 2013 (2013)

10. Figueroa, P., Bischof, W.F., Boulanger, P., Hoover, H.J., Taylor, R.: Intml: a dataflow oriented development system for virtual reality applications. Presence: Teleoperators Virtual Environ. **17**(5), 492–511 (2008)
11. Figueroa, P., Hoover, J., Boulanger, P.: Intml concepts. Technical Report, University of Alberta, Edmonton, Canada (2004)
12. Kwiatkowska, M., Norman, G., Parker, D.: Probabilistic symbolic model checking with PRISM: a hybrid approach. Int. J. Softw. Tools Technol. Transf. **6**(2), 128–142 (2004)
13. Latoschik, M.E.: Designing transition networks for multimodal VR-interactions using a markup language. In: ICMI 2002. p. 411. Pittsburgh, USA (2002)
14. Moine, M.: Implementation tool of Timed Automata specifications. Master's thesis, ENSIIE - Université d'Evry-val d'Essonne (2013)
15. Navarre, D., Palanque, P., Bastide, R., Schyn, A., Winckler, M., Nedel, L.P., Freitas, C.M.: A formal description of multimodal interaction techniques for immersive virtual reality applications. In: Costabile, M.F., Paternó, F. (eds.) Human-Computer Interaction–INTERACT 2005, pp. 170–183. LNCS, Springer, Berlin Heidelberg (2005)
16. Rataj, A., Wozna, B., Zbrzezny, A.: A Translator of Java Programs to TADDs. Fundamenta Informaticae **93**(1–3), 305–324 (2009)
17. Sandor, C., Reicher, T.: CUIML: A Language for the Generation of Multimodal Human-Computer Interfaces. In: UIML 2001. vol. 124 (2001)
18. Uppsala Universitet: Uppaal, http://www.uppaal.org/
19. Waez, M.T.B., Dingel, J., Rudie, K.: Timed automata for the development of real-time systems. Research Report 2011–579, Queen's University, Kingston, Canada (2011)

A Generative Approach to Tracking Hands and Their Interaction with Objects

Nikolaos Kyriazis, Iason Oikonomidis, Paschalis Panteleris, Damien Michel, Ammar Qammaz, Alexandros Makris, Konstantinos Tzevanidis, Petros Douvantzis, Konstantinos Roditakis and Antonis Argyros

Abstract Markerless 3D tracking of hands in action or in interaction with objects provides rich information that can be used to interpret a number of human activities. In this paper, we review a number of relevant methods we have proposed. All of them focus on hands, objects and their interaction and follow a generative approach. The major strength of such an approach is the straightforward fashion in which arbitrarily complex priors can be easily incorporated towards solving the tracking problem and their capability to generalize to greater and/or different domains. The proposed generative approach is implemented in a single, unified computational framework.

Keywords 3D hand tracking · 3D tracking of hand-object interactions · Model-based 3D tracking

1 Introduction

This work discusses the problem of observing and understanding human (inter)action through markerless observations, i.e. in a non noninvasive manner. Human action is considered with focus on human hands. Understanding human hands in action is a theoretically and practically interesting problem. Humans are readily capable of understanding hand actions of their own and of other humans. Interestingly, the idea supported in the literature is that to achieve such understanding, humans employ mental simulation [6, 7], which could tentatively be corresponded to the generative approach in computer vision. On the practical side, achieving the understanding

N. Kyriazis · I. Oikonomidis · P. Panteleris · D. Michel · A. Qammaz
A. Makris · K. Tzevanidis · P. Douvantzis · K. Roditakis · A. Argyros (✉)
Institute of Computer Science, Foundation for Research and Technology, Heraklion, Greece
e-mail: argyros@ics.forth.gr

N. Kyriazis · I. Oikonomidis · K. Tzevanidis · P. Douvantzis · K. Roditakis · A. Argyros
Computer Science Department, University of Crete, Heraklion, Greece

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_2

of human hand action sets the foundation upon which a variety of socially helpful applications can be established, in the fields of safety, medicine, education, industry, entertainment, etc. Substituting occasionally flawed visual scrutiny by humans for systematically robust mechanised processing of images can have a significant positive impact on the success of the underlying tasks.

As demonstrated in our work, rich understanding of hand action and interaction with objects can be successfully built upon robust and detailed 3D tracking of hands and objects, from images captured by noninvasive camera setups. However, the 3D tracking task is not an easy one. The structural complexity of the human hand corresponds to multiple Degrees of Freedom (DoFs) which yield a tracking problem with a difficult to explore high-dimensional search space. Highly frequent and temporally prolonged self-occlusions lead to insufficient observations for full and proper estimation of hand articulation. The uniformity of finger appearance introduces ambiguities in observation, where e.g. one finger can be mistaken for another. It is common that human hands occupy little area in captured images and therefore correspond to observations of low spatial resolution. On top of that, hands may move quite fast, resulting in a relatively low temporal resolution, too, even if observed by cameras operating in 30 fps. Adding more hands and/or objects in 3D tracking only aggravates the aforementioned problems and also increases computational complexity.

2 The Proposed Generative Approach

We present our generative approach in problems pertaining to 3D tracking of hands in isolation Sect. 2.1 and hands in interaction with objects Sect. 2.2. A unified approach [11] is employed in all of the presented work.

The presented work is founded on the premise that there exists a simulation process which can synthesize data similar to acquired observations. The configuration of the simulation that best matches some observations is the "explanation" or understanding of the observations. Simulations involve synthesizing appearance, through 3D rendering, and even behaviour, if physics-based simulation is employed.

Assuming temporal continuity, tracking objects in 3D amounts to searching for the 3D configuration of the objects whose simulation best matches actual observations. Searching is performed in the vicinity of the tracking result for the previous frame. For each tracking frame the simulation error $\mathbf{E}$ is minimized:

$$\mathbf{E}(x, o, h) = \|\mathbf{M}(x, h) - \mathbf{P}(o)\| + \lambda \mathbf{L}(x, h), \tag{1}$$

where x is the state of all tracked objects, o are the observations of the current frame, h is the tracking history, $\mathbf{M}$ is the simulation function, $\mathbf{P}$ is the pre-processing function which maps observation to the same feature space as $\mathbf{M}$ and $\mathbf{L}$ is a prior on the 3D configuration of objects. The first term is a notational abstraction of a data term, i.e. a quantification of the discrepancy between a given hypothesis and actual

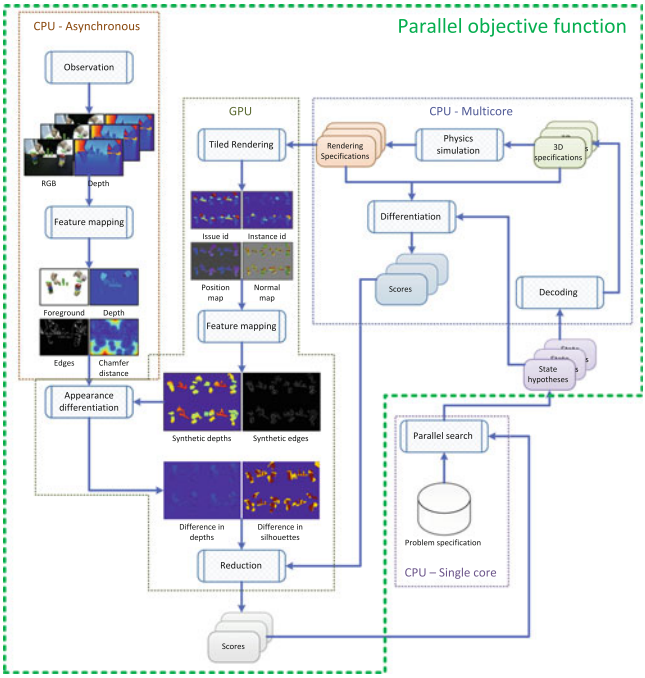


Fig. 1 The outline of the implementation for the generative 3D tracking framework. For more details the reader is referred to [11]

measurements. The prior term and data term are balanced with a weight λ that is empirically defined.

The tracking solution s for the current frame amounts to minimizing (1):

$$s \triangleq \arg\min_{x} \mathbf{E}(x, o, h). \tag{2}$$

During search for the optimal x, (1) is invoked several times. To favor speed, parallel search techniques are incorporated and the implementation of (1) exploits GPU acceleration. An outline of the implementation of the generic approach is depicted in Fig. 1.

2.1 Tracking Hands in Isolation

There is significant interest in pursuing fast and robust 3D tracking of a single hand. However, as already discussed, tracking a single hand in 3D is already a formidable challenge. We hereby present a series of successful 3D single hand tracking efforts.

Relevant Work The problem of 3D hand tracking has received significant attention. There exist several approaches to solving the problem, with every one including a top-down component and a bottom-up component. According to the contribution weight of each component, approaches can be characterized as top-down (dominating top-down component), bottom-up (dominating bottom-up component) and hybrid (equally contributing top-down and bottom-up components). Top-down methods work by establishing a model of the hand and fitting it to moderately preprocessed observations, through search [3, 14, 29, 31, 35, 38, 42, 43, 48]. The methods presented in this paper call within this category too. Bottom-up methods work by learning, via machine learning tools, the mapping from the image space to the configuration space, from large training corpora [1, 4, 8, 9, 16, 34, 36, 44, 46]. Hybrid methods work by fusing top-down models with strong bottom-up features, which are a product of learning [30, 40, 41, 49].

Contribution We have investigated the 3D tracking problem for a hand in isolation and under different cases of camera setups, representations and optimization tools [5, 15, 20–22, 24].

With respect to camera setups we have explored two options, namely a set of calibrated RGB cameras and a single RGBD camera (MS Kinect, ASUS Xtion). What is differentiated between the two classes is the feature space used while computing (1). For the case of multiple RGB cameras we have employed per camera 2D features (silhouettes, edges) which are fused in 3D information as a result of incorporation in an objective function (see 1) defined over 3D configuration [24]. We have also employed 3D features (visual hulls) which, conversely, where computed as a fusion of multiple 2D cues prior to their incorporation in the computations of the objective function [21]. The latter [21] presents favorable traits for the small baseline stereo (2 cameras) case and is otherwise similar to the former [24]. For the RGBD case we have employed both 2D (silhouettes) and 3D (depth) features in (1) [5, 15, 20, 22]. The benefit of employing a single RGBD camera instead of multiple RGB cameras is the significant decrease of data which require processing while maintaining the required 3D information. This allows for 3D tracking of a single hand which can currently be performed at a rate of 30 fps.

The default representation of a hand in all presented work has been a kinematics tree (skeleton) of 27 parameters, redundantly encoding 26 DoFs. However, it is noticed that hand motion is often modulated by an underlying task. We investigated whether by conditioning on tasks, hand motion could be described with a reduced amount of parameters. In [5] we came up with per-task sub-spaces of hand motion, which we successfully employed in 3D hand tracking.

Last but not least, we have employed different optimization schemes for computing (2). In most of the hand tracking work [5, 20, 21, 24] (2) was computed using a variant (see [20] for details) of the Particle Swarm Optimization algorithm [28]. Introducing a custom evolutionary technique for search [22] and replacing PSO with it dramatically decreased the amount of required invocations to (1) for computing (2), with respect to [20], while preserving the accuracy level. By substituting the find-best optimization schemes of [20, 22] with a probability density propagation scheme,

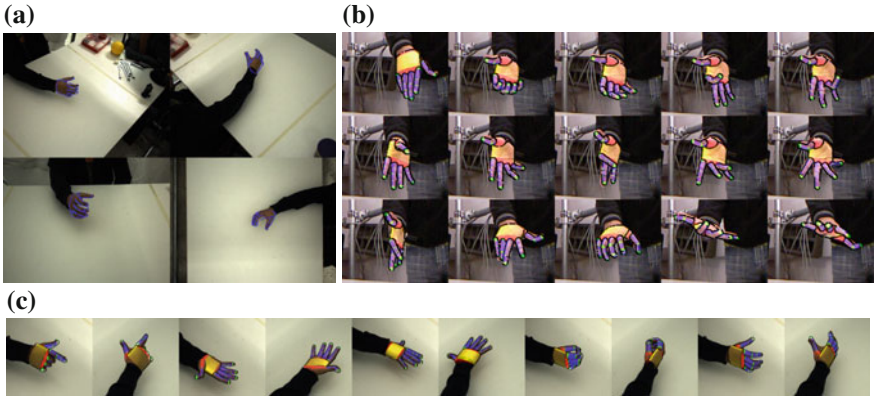


Fig. 2 Exemplar results of 3D tracking of a single hand from **a** multiple 2D cues [24], **b** a single RGBD camera [20] and **c** convex hulls computed from multiple views [20]

namely Hierarchical Particle Filtering [15], not only 3D hand tracking became even faster (90 fps) but it also became more robust, due to the persistence of several hypotheses across frames, instead of a single one as in [5, 20, 22, 24]. Indicative results of the 3D hand tracking methods are provided in Fig. 2. For more details the reader is referred to the corresponding papers.

2.2 Tracking Interacting Hands

In a significant subset of the scenarios which involve observation of a human, hands are not isolated. They are usually found interacting with other hands or with objects in the surrounding of the subject. The amount of objects manipulated over time can vary from one (explore or use a single object) to many (preparing multi-ingredient food). Tracking such scenes in 3D inherits the list of single hand tracking problems, in aggravated forms, accentuated by the cardinality of the scenes.

Relevant Work There exist some approaches in the literature to handle 3D tracking of hand-object interactions, that can be corresponded to the categorization of top-down [47], bottom-up [33] and hybrid [2] methods. However, they all regard a single object. On the other hand, there exist approaches that tackle the problem of tracking multiple objects in 3D, but do not include a hand [10, 37, 45].

Contribution We have performed work which spans across several choices over setups, representations, etc. Even more importantly, the corresponding methods also regard different scene scales. We will present the corresponding methods focusing on the axis of scene scale and denoting important points on the way.

The multi-camera 3D hand tracking method of [24] has been extended in [19] to also include an object of known parametric shape. The extension amounted to introducing the DoFs of the object and its 3D shape in the computations of (1) and (2). Thus, both the hand and the object where tracked, by requiring a computational budget which was a bit larger than that used in [24]. In [23] we showed that this extension can successfully scale up to the case of the object being as complex as another hand, increasing the DoFs to 54 (27 for each hand) and also increasing the computational budget. In both cases, the state of the hand and the other hand or object were considered jointly, as required in order to model constraints such as, for example, that two physical objects cannot occupy the same physical space. PSO was used in both cases and succeeded, given enough computational budget.

However, generalizing to larger scenes is problematic, as the total DoFs dramatically increase. Because of lack of gradient or any other type of focus, PSO cannot effectively explore arbitrarily large search spaces with limited resources. To introduce a sense of focus, the joint problem of tracking many objects and/or hands in 3D was carefully decomposed to several semi-independent tracking problems, one for each entity [13]. Instead of a single tracker for the entire scene, an Ensemble of Collaborative Trackers is incorporated. Each of these trackers treats the last known state of all others as static and thus becomes independent of them, while at the same time it incorporates their state in its computations. This allows to simultaneously decompose the problem but also preserve reasoning over the entire state in each tracker. The decomposition, along with the parallel implementation (CPU and GPU) yield sub-linear increase in computational time as the number of objects increases and significantly outperforms schemes such as the joint optimization of [19, 23] and truly independent trackers (e.g. multiple instances of [20]).

The ECT method in [13] was also endowed with a hand-object manipulation prior which allowed for even computing forces from vision alone [27]. When a hand and an object are tracked by an ECT, erroneous measurements or optimization hiccups can lead to physically implausible manipulation estimations. By tracking the acceleration of the manipulated object and consulting a per finger force prior (learned through training) the hand-object estimation can be rectified so as to become physically plausible. At the same time, the actual forces exerted by the subject are also computed.

The notion of physical plausibility has also been exploited in [12] to enable scalable tracking of a scene comprising a hand and several objects. The observation that objects move only due to the motion of the hand leads to the formulation of a hand-objects tracking problem whose DoFs are the same as the hand's alone. Otherwise stated, the amount of passive objects makes no difference in the size of the search space, since in order to track any subset of moving objects it suffices to only search for a hand motion whose physical consequences would have the objects move. Some results of the 3D hand-object(s) tracking methods can be viewed at Fig. 3. For more details the reader is referred to the corresponding papers.

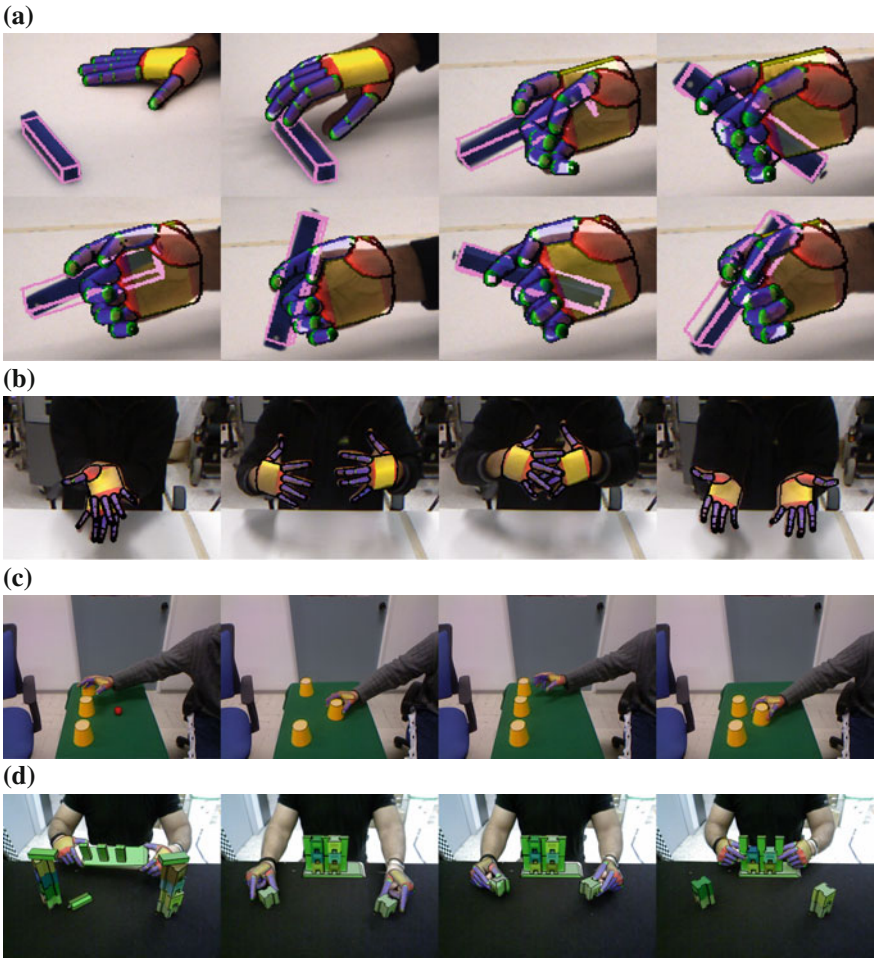


Fig. 3 Exemplar results of 3D tracking of hands and objects in interaction: **a** hand-object interaction observed from a multi-camera system [18], **b** two strongly interacting hands observed from an RGBD camera [23], **c** a hand interacting with a few objects observed from an RGBD camera [12] and **d** two hands interacting with many objects observed from an RGBD camera [13]

3 Discussion

We presented a series of methods to solving the problem of tracking hands in action or in interaction with objects, in 3D. All the works followed the generative approach, as implemented in a single unified and modular architecture [11]. The generative approach has successfully led to tackling the 3D tracking problems and establishing state-of-the-art solutions. At the same time, the generative approach, as opposed to discriminative or hybrid approaches, has straightforward means to facilitate

generalization to larger and/or different domains. As an example, the same approach used in the presented work has also been successfully employed in problems of tracking hands with markers [32], tracking head motion [25] and tracking the full body [17]. The results of the presented 3D hand-object tracking methods have also been employed to fuel even higher-level inference. Specifically, reconstructed and detailed 3D trajectories of hands manipulating objects have been used to establish a high-level action grammar for everyday tasks [26] and also to enable the inference of the manipulation intent a subject has while approaching an object, even before manipulation is actually observed [39].

Acknowledgments This work was partially supported by the by the EU ISTFP7-IP-288533 project RoboHow.Cog and by the EU FP7-ICT-2011-9-601165 project WEARHAP.

References

1. Athitsos, V., Sclaroff, S.: Estimating 3d hand pose from a cluttered image. In: CVPR 2003. vol. 2, pp. 432–439. Madison, USA (2003)
2. Ballan, L., Taneja, A., Gall, J., Van Gool, L., Pollefeys, M.: Motion Capture of Hands in Action Using Discriminative Salient Points. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7577, pp. 640–653. Springer, Heidelberg (2012)
3. Bray, M., Koller-Meier, E., Van Gool, L.: Smart particle filtering for high-dimensional tracking. Comput. Vis. Image Underst. **106**(1), 116–129 (2007)
4. de Campos, T., Murray, D.: Regression-based hand pose estimation from multiple cameras. In: CVPR 2006. vol. 1, pp. 782–789. New York, USA (2006)
5. Douvantzis, P., Oikonomidis, I., Kyriazis, N., Argyros, A.: Dimensionality reduction for efficient single frame hand pose estimation. In: Chen, M., Leibe, B., Neumann, B. (eds.) Computer Vision Systems. LNCS, vol. 7963, pp. 143–152. Springer, Heidelberg (2013)
6. Gallese, V., Goldman, A.: Mirror neurons and the simulation theory of mind-reading. Trends Cogn. Sci. **2**(12), 493–501 (1998)
7. Grezes, J., Decety, J.: Functional anatomy of execution, mental simulation, observation, and verb generation of actions: a meta-analysis. Hum. Brain Mapp. **12**(1), 1–19 (2001)
8. Keskin, C., Kirac, F., Kara, Y., Akarun, L.: Real time hand pose estimation using depth sensors. In: ICCV 2011. pp. 1228–1234. Barcelona, Spain (2011)
9. Keskin, C., Kıraç, F., Kara, Y., Akarun, L.: Hand pose estimation and hand shape classification using multi-layered randomized decision forests. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7577, pp. 852–863. Springer, Heidelberg (2012)
10. Kim, K., Lepetit, V., Woo, W.: Keyframe-based modeling and tracking of multiple 3D objects. In: ISMAR 2010. pp. 193–198. Seoul, Japan (2010)
11. Kyriazis, N.: A computational framework for observing and understanding the interaction of humans with objects of their environment. Ph.D. thesis, University of Crete (2014)
12. Kyriazis, N., Argyros, A.: Physically plausible 3D scene tracking: The single actor hypothesis. In: CVPR 2013. pp. 9–16. Portland, Oregon, USA (2013)
13. Kyriazis, N., Argyros, A.: Scalable 3D tracking of multiple interacting objects. In: CVPR 2014. pp. 3430–3437. Columbus, Ohio, USA (2014)
14. MacCormick, J., Isard, M.: Partitioned sampling, articulated objects, and interface-quality hand tracking. In: In: Proceedings of the 6th EuropeanConference on Computer Vision-Part II, ECCV 2000. LNCS, pp. 3–19. Springer, Heidelberg (2000)

15. Makris, A., Kyriazis, N., Argyros, A.: Hierarchical particle filtering for 3d hand tracking. In: CVPR 2015. pp. 8–17. Boston, USA (2015)
16. Malassiotis, S., Strintzis, M.G.: Real-time hand posture recognition using range data. Image Vis. Comput. **26**(7), 1027–1037 (2008)
17. Michel, D., Panagiotakis, C., Argyros, A.A.: Tracking the articulated motion of the human body with two RGBD cameras. Mach. Vis. Appl. **26**(1), 41–54 (2013)
18. Oikonomidis, I., Kyriazis, N., Argyros, A.: Full dof tracking of a hand interacting with an object by modeling occlusions and physical constraints. In: ICCV 2011. pp. 2088–2095. Barcelona, Spain (2011)
19. Oikonomidis, I., Kyriazis, N., Argyros, A., et al.: Full dof tracking of a hand interacting with an object by modeling occlusions and physical constraints. In: ICCV 2011. pp. 2088–2095. Barcelona, Spain (2011)
20. Oikonomidis, I., Kyriazis, N., Argyros, A.A.: Efficient model-based 3D tracking of hand articulations using kinect. In: BMVC 2011. p. 3. Dundee, UK (2011)
21. Oikonomidis, I., Kyriazis, N., Tzevanidis, K., Argyros, A., et al.: Tracking hand articulations: relying on 3D visual hulls versus relying on multiple 2D cues. In: ISUVR 2013. pp. 7–10. Daejeon, South Korea (2013)
22. Oikonomidis, I., Lourakis, M., Argyros, A., et al.: Evolutionary quasi-random search for hand articulations tracking. In: CVPR 2014. pp. 3422–3429. Columbus, Ohio, USA (2014)
23. Oikonomidis, I., Kyriazis, N., Argyros, A.A.: Tracking the articulated motion of two strongly interacting hands. In: CVPR 2012. pp. 1862–1869. Providence, Rhode Island (2012)
24. Oikonomidis, I., Kyriazis, N., Argyros, A.: Markerless and efficient 26-DOF hand pose recovery. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) Computer Vision—ACCV 2010. LNCS, vol. 6494, pp. 744–757. Springer, Heidelberg (2011)
25. Padeleris, P., Zabulis, X., Argyros, A., et al.: Head pose estimation on depth data based on particle swarm optimization. In: CVPRW 2012. pp. 42–49. Providence, USA (2012)
26. Patel, M., Ek, C.H., Kyriazis, N., Argyros, A., Miro, J.V., Kragic, D.: Language for learning complex human-object interactions. In: ICRA 2013. pp. 4997–5002. Karlsruhe, Germany (2013)
27. Pham, T.H., Kheddar, A., Qammaz, A., Argyros, A.A.: Towards force sensing from vision: observing hand-object interactions to infer manipulation forces. In: CVPR 2015. pp. 1893–1902. Boston, USA (2015)
28. Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization. Swarm Intell. **1**(1), 33–57 (2007)
29. Qian, C., Sun, X., Wei, Y., Tang, X., Sun, J.: Realtime and robust hand tracking from depth. In: CVPR 2014. pp. 1106–1113. Columbus, USA (2014)
30. Qian, C., Sun, X., Wei, Y., Tang, X., Sun, J.: Realtime and robust hand tracking from depth. In: CVPR 2014. pp. 1106–1113. Ohio, USA (2014)
31. Rehg, J.M., Kanade, T.: Visual tracking of high DOF articulated structures: an application to human hand tracking. In: ECCV 1994. pp. 35–46. Springer, London, UK (1994)
32. Roditakis, K., Argyros, A.: Quantifying the effect of a colored glove in the 3D tracking of a human hand. In: Nalpantidis, L., Krüger, V., Eklundh, J.O., Gasteratos, A. (eds.) Computer Vision Systems. LNCS, vol. 9163, pp. 404–414. Springer, Heidelberg (2015)
33. Romero, J., Kjellström, H., Ek, C.H., Kragic, D.: Non-parametric hand pose estimation with object context. Image Vis. Comput. **31**(8), 555–564 (2013)
34. Romero, J., Kjellström, H., Kragic, D.: Monocular real-time 3D articulated hand pose estimation. In: IEEE-RAS 2009. pp. 87–92. Paris, France (2009)
35. Ros, G., del Rincon, J., Mateos, G.: Articulated particle filter for hand tracking. In: ICPR 2012. pp. 3581–3585. Stockholm, Sweden (2012)
36. Rosales, R., Athitsos, V., Sigal, L., Sclaroff, S.: 3D hand pose reconstruction using specialized mappings. In: ICCV 2001. vol. 1, pp. 378–385. Vancouver, Canada (2001)
37. Salzmann, M., Urtasun, R.: Physically-based motion models for 3D tracking: a convex formulation. In: ICCV 2011. pp. 2064–2071. Barcelona, Spain (2011)

38. Schmidt, T., Newcombe, R., Fox, D.: Dart: Dense articulated real-time tracking. In: RSS 2014. vol. 2. Berkeley, USA (2014)
39. Song, D., Kyriazis, N., Oikonomidis, I., Papazov, C., Argyros, A., Burschka, D., Kragic, D.: Predicting human intention in visual observations of hand/object interactions. In: ICRA 2013. pp. 1608–1615. Karlsruhe, Germany (2013)
40. Sridhar, S., Oulasvirta, A., Theobalt, C.: Interactive markerless articulated hand motion tracking using RGB and depth data. In: ICCV 2013. pp. 2456–2463. Sydney, Australia (2013)
41. Sridhar, S., Rhodin, H., Seidel, H.P., Oulasvirta, A., Theobalt, C.: Real-time hand tracking using a sum of anisotropic gaussians model. In: 3DV 2014. Tokyo, Japan (2014)
42. Stenger, B., Mendonça, P.R., Cipolla, R.: Model-based 3D tracking of an articulated hand. In: CVPR 2001. vol. 2, pp. 310–315. Kauai, HI, USA (2001)
43. Sudderth, E., Mandel, M., Freeman, W., Willsky, A.: Visual hand tracking using nonparametric belief propagation. In: CVPRW 2004. pp. 189–189. Washington, USA (2004)
44. Tompson, J., Stein, M., Lecun, Y., Perlin, K.: Real-time continuous pose recovery of human hands using convolutional networks. ACM Trans. Graph. **33**(5), 169–169 (2014)
45. Vo, B., Vo, B., Pham, N., Suter, D.: Joint detection and estimation of multiple objects from image observations. IEEE Trans. Signal Process. **58**(10), 5129–5141 (2010)
46. Wang, R., Paris, S., Popovic, J.: 6D hands: markerless hand-tracking for computer aided design. In: UIST 2011. pp. 549–558. UIST '11, ACM, New York, USA (2011)
47. Wang, Y., Min, J., Zhang, J., Liu, Y., Xu, F., Dai, Q., Chai, J.: Video-based hand manipulation capture through composite motion control. ACM Trans. Graph. (TOG) **32**(4), 43:1–43:14 (2013)
48. Wu, Y., Lin, J.Y., Huang, T.S.: Capturing natural hand articulation. In: ICCV 2001. vol. 2, pp. 426–432. Vancouver, USA (2001)
49. Xu, C., Cheng, L.: Efficient hand pose estimation from a single depth image. In: ICCV 2013. pp. 3456–3462. Sydney, Australia (2013)

Design of Biomedical Robots for the Analysis of Cancer, Neurodegenerative and Rare Diseases

Juan L. Fernández-Martínez, Enrique J. deAndrés-Galiana and Stephen T. Sonis

Abstract Studies of genomics make use of high throughput technology to discover and characterize genes associated with cancer and other illnesses. Genomics may be of particular value in discovering mechanisms and interventions for neurodegenerative and rare diseases in the quest for orphan drugs. To expedite risk prediction, mechanism of action and drug discovery, effectively, analytical methods, especially those that translate to clinical relevant outcomes, are highly important. In this paper, we define the term biomedical robot as a novel tool for genomic analysis in order to improve phenotype prediction, identifying disease pathogenesis and significantly defining therapeutic targets. The implementation of a biomedical robot in genomic analysis is based on the use of feature selection methods and ensemble learning techniques. Mathematically, a biomedical robot exploits the structure of the uncertainty space of any classification problem conceived as in an ill-posed optimization problem, that is, given a classifier several equivalent low scale signatures exist providing similar prediction accuracies. As an example, we applied this method to the analysis of three different expression microarrays publically available concerning Chronic Lymphocytic Leukemia, Inclusion Body Myositis/Polimyositis (IBM-PM) and Amyotrophic Lateral Sclerosis (ALS). Using these examples we showed the value of the biomedical robot concept to improve knowledge discovery and provide decision systems in order to optimize diagnosis, treatment and prognosis. The goal of the FINISTERRAE project is to leverage publically available genetic databases of rare and neurodegenerative diseases and construct a relational database with the genes and genetic pathways involved, which can be used to accelerate translational research in this domain.

J.L. Fernández-Martínez (✉) · E.J. deAndrés-Galiana
Mathematics Department, University of Oviedo, Asturias, Spain
e-mail: jlfmuniovi@gmail.com

E.J. deAndrés-Galiana
Artificial Intelligence Centre, Machine Learning Group,
University of Oviedo, Asturias, Spain

S.T. Sonis
Biomodels LLC, Watertown, MA, USA

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_3

Keywords Biomedical robots · Genomics · Cancer · Rare and Neurogenerative diseases

1 Introduction

Studies of genomics make use of high throughput technology to discover and characterize genes associated with cancer and other illnesses. Genomics may be of particular value in discovering mechanisms and interventions for rare diseases. Cancer genomics is a sub-field of genomics that employs high throughput technology to discover and characterize genes associated with cancer. It could be also defined as the application of genomics and personalized medicine to cancer research. The goal of oncogenomics is to identify new genes related to cancer that may improve diagnosis, treatment and prognosis [13]. While much has been published about genomics and cancer development, limited mutations are only associated with 5–10 % of cancers which limits the utility of genetic testing to determine cancer risk for most malignancies. Furthermore, although genetic testing for cancer risk has received a huge amount of attention, a recent report seems to undermine its importance, concluding that much of cancer risk is due to random causes. Nonetheless, the use of genomics in cancer may certainly contribute to understanding how mutations impact the expression of different important cancer risk genes and also how patients may or may not respond to specific forms of cancer therapy.

Rare diseases, when taken together, are not that rare at all, since according to the National Institutes of Health (NIH) they affect 30 million Americans, and also almost 5 per 10.000 citizens of the European Union, that is, 30 million of people as well. There are nearly 7000 diseases that are officially catalogued as rare. The classification of a disease as rare is not ubiquitous. In the U.S. rare classification requires an incidence of less than 200.000 individuals, whereas it is defined as less than 1 per 2000 individuals in Europe. The majority of rare diseases are of genetic origin and half impact children. Both cancer and neurodegenerative diseases are high priorities for the biopharmaceutical industry as they both represent significant and growing unmet clinical needs. Cancer incidence is increasing and is likely to accelerate in the near future secondary to the aging population and better therapies for other late-in-life diseases. Similarly improved diagnostic tools and therapies for neurodegenerative diseases are priorities. In Europe 16 % of the population is older than 65 years old; of neurodegenerative diseases, Alzheimer disease alone affects more than 7 million Europeans and is expected to double every 20 years.

"As noted by Eli and Edythe of the Broad Institute for Biomedical Research of Harvard and MIT we have a historic opportunity and responsibility to transform medicine by using systematic approaches to dramatically accelerate the understanding and treatment of disease". Here we present the concept of "Biomedical Robots" and their genomic analytic applications.

Biomedical Robots are a set of bioinformatic algorithms derived from a combination of applied mathematics, statistics and computer science that are capable of analyzing dynamically (as a function of time) very high dimensional data, discovering knowledge, generating new biomedical working hypotheses, and supporting medical decision-making with its corresponding uncertainty assessment. The structure of the paper is as follows: first the oncogenomics problem is formally introduced. Secondly we describe the methodology to address the problem and the techniques used for the design of a biomedical robot. This formalism is applied to a synthetic dataset to get some preliminary conclusions (sensitivity analysis). Finally we apply this set up to different datasets concerning the analysis of Chronic Lymphocytic Leukemia (CLL), Inclusion Body Myositis (IBM)-Polimyositis(PM), and Amyotrophic Lateral Sclerosis (ALS).

2 The Oncogenomics Problem

The main target in oncogenomics is identifying the set of genes that are related to cancer (oncogenes) or to the development of any other illness in order to improve diagnosis, treatment and prognosis. For that purpose the problem will be posed as a classification problem in order to find the sets of genes that provide an optimal classification of a given phenotype. Typically the phenotype discrimination will consist in finding the sets of genes that optimally differentiate between a person developing the illness from others control samples corresponding to healthy individuals. The classification problem of phenotype discrimination does not need necessarily to be binary, that is, it could be multiclass. Also the problem could consist in predicting toxicity risks in patients, identifying biological and molecular pathways differences between normal and cancer cells, or investigating drug mechanisms of action. Accordingly, the corresponding machine learning oncogenomics problem can be established as a nonlinear classification problem, since the classifier and the genetic features that serve to achieve an optimum prediction of the phenotype are unknown, because no physical relation is at disposal to predict the phenotype based on the genetic information. Therefore, as a first step, a given type of classifier (nearest-neighbor, neural networks, SVM, etc.) should be built ad-hoc. This can be considered as a first source of uncertainty. Let us imagine that we have at disposal a set of expressions of n genes (or probes) for a set of m samples whose phenotype classes are provided by a medical expert annotation. Without loss of generality, let us also suppose that the classification problem is binary, that is, the phenotype of the samples belong either to class 1 or class 2. This information is typically organized in the expression matrix $E \in M_{m\times n}(\mathbb{R})$ with $m \ll n$ and in the class vector $\mathbf{c}^{obs}$. The classifier L^* can be formally defined as an application between the set of genetic features $\mathbf{g} \in M \subset \mathbb{R}^n$ and the set of classes $C = \{c_1, c_2\}$:

$$L^* : M \to C = \{c_1, c_2\}. \tag{1}$$

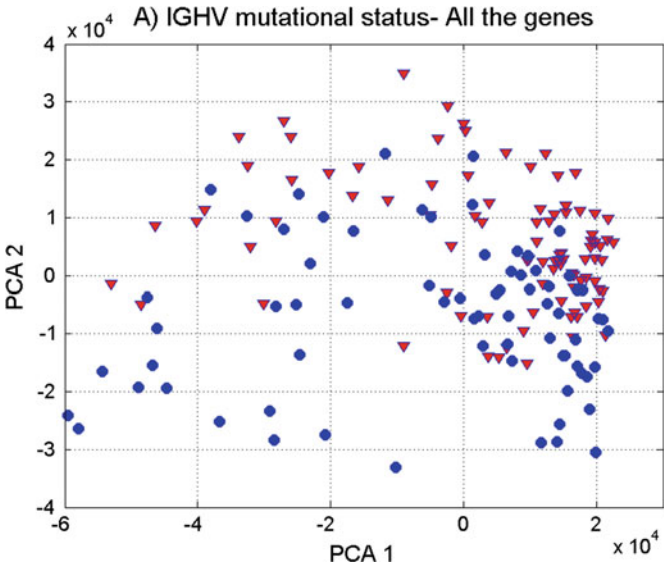


Fig. 1 IgVH classification in CLL: considering all the samples the problem is nonlinear

Nevertheless not all the genes (or genetic probes) are involved in the oncogenomics inverse problem. Furthermore, when all the genetic information from samples is considered, the corresponding classification problem becomes nonlinear separable. This can be easily viewed performing clustering in the PCA base deduced taking into account all the genetic information that has been sampled and showing that it is impossible to find a hyper plane separating both classes. As an example of this fact, Fig. 1 shows the PCA clustering for the case of Chronic Lymphocytic Leukemia (CLL), where we tried to classify the IgVH mutational status, that is crucially important to predict prognosis in CLL patients. As it was already mentioned, it can be observed that using the entire genetic information of the patients, the classification problem is nonlinearly separable.

Nevertheless, it is possible to discard irrelevant features, that is, those genes that do not provide any useful information for the phenotype discrimination, since these features might introduce noise/ambiguity in the classification. Therefore, the relevant genes would be the ones that minimize a given target function $O(\mathbf{g})$ related to the class prediction:

$$\mathbf{g} : O(\bar{\mathbf{g}}) = \min_{\mathbf{g}\in\mathbb{R}^S} O(\mathbf{g}), \tag{2}$$

$$O(\mathbf{g}) = ||\mathbf{L}^*(\mathbf{g}) - \mathbf{c}^{obs}||_p \tag{3}$$

$$\mathbf{L}^*(\mathbf{g}) = (L^*(\mathbf{g}_1), \ldots, L^*(\mathbf{g}_i), \ldots, L^*(\mathbf{g}_m)), \tag{4}$$

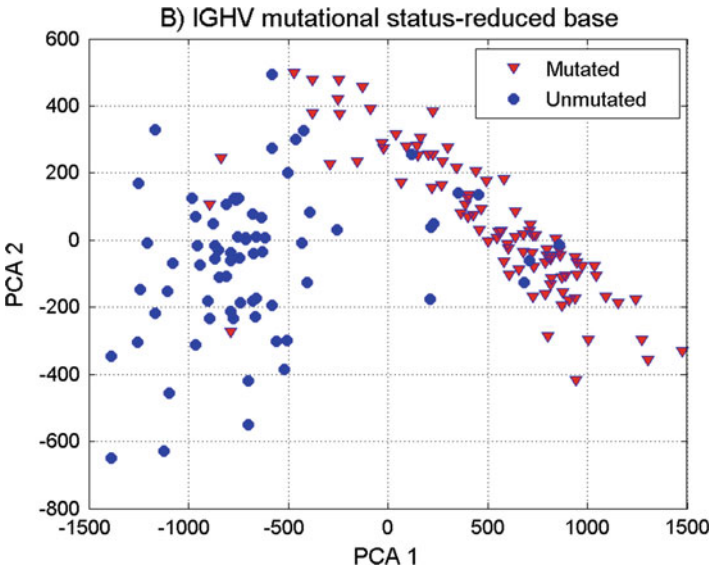


Fig. 2 IgVH classification in CLL: using the most discriminatory genes the classification problem becomes linearly separable

where $\mathbf{c}^{obs}$ is the set of observed classes, p is the norm applied in the distance criterion, $\mathbf{L}^*(\mathbf{g})$ is the set of predicted classes and $\mathbf{g}_i \in \mathbb{R}^{N^S}$ is the genetic signature corresponding to sample i.

An important conclusion is that the selected relevant genes make the classification problem to be linear separable, as it is shown in Fig. 2 for the CLL problem. In this case we have only used the most discriminatory genes (13) to make the PCA projection. It can be observed that classification becomes almost linearly separable and only a few samples are misclassified.

Several considerations are important at this point:

- **Ill-posed character**: the oncogenomic problem as any inverse problem is ill-posed, that is, several genetic signatures with the same number of genes exist, explaining the phenotype class equally well (similar predictive accuracy). Also if we allow the number of genes to vary, the number of equivalent genetic signatures increases. It is possible to introduce the parsimony principle, which consists in finding small scale signatures by introducing the concept of redundant feature. Given a genetic signature $\mathbf{g} \in \mathbb{R}^s$ characterized by its class prediction accuracy and length s, redundant features (or genes) are those that provide no additional information than the currently selected features, that is, the prediction accuracy does not increase by adding these genetic features to $\mathbf{g}$ in the classifier.
- **High degree of underdeterminacy**: the ill-posed character of the classification is due to the high underdetermined character of the inverse problem involved, since the number of samples m is much lower that the total number of probes n.

Fernández-Martínez et al. [2, 4] analyzed the uncertainty space of linear and nonlinear inverse (and classification) problems showing that the topography of the cost function $O(\mathbf{g})$ in the region of lower misfits (or higher predictive accuracies) correspond to flat elongated valleys with null gradients, where high predictive genetic signatures of the same length s reside. This valley is unique and rectilinear if the classification/inverse problem is linear, and bends and might be composed of several disconnected basins if the inverse problem is nonlinear. Obviously the topography changes if the space where the optimization is performed ($\mathbb{R}^s$) is changed. Also, if we are somehow able to define the discriminatory power of the different genes, a classification problem could be interpreted as the Fourier expansion of a signal, that is, there will be genes that provide alone high accuracy for the classification problem (head genes), while others will provide the high frequency details (helper genes) to improve the prediction accuracy. Nevertheless, there is a time when adding more details to the classifier does not increase its predictive accuracy. The smallest scale signature is the one that has the least number of highest discriminatory genes. This knowledge could be important for diagnosis.

- **Noise in data**: the presence of noise in genomic data will impact the classification. There are two main sources of noise:
  - Noise in the genes expressions that is introduced by the process of measurement.
  - Noise in the class assignment, that is due to an incorrect labeling of the sample by the experts (or medical doctors). For instance, sometimes the classification problem is parameterized as binary and in reality more than two classes exist. Therefore, assigning two different classes to the samples will provide noise in the classification. In this case, finding a predictive accuracy lower than 100 % would be an expected result, otherwise the algorithm will find a wrong genetic signature in order to fit (or explaining) the wrong class assignment. Obviously this situation is always difficult to detect, since the normal strategy usually consists in achieving a perfect prediction.

In presence of these types of noise the genetic signature with the highest predictive accuracy (and therefore the lowest misfit error) will never perfectly coincide with the genetic signature(s) that explains the disease (noise-free phenotype classification problem). For that reason it is desirable to look also for genetic signatures with lower predictive accuracy than the optimum.
To show this fact, let us imagine that we introduce noise $\delta\mathbf{c}$ in the class assignment, that is, $\mathbf{c}^{obs} = \mathbf{c}^{true} + \delta\mathbf{c}$, then:

$$\begin{aligned} O^p(\mathbf{g}) &= ||\mathbf{L}^*(\mathbf{g}) - \mathbf{c}^{obs}||_p \\ &= ||\mathbf{L}^*(\mathbf{g}) - \mathbf{c}^{true} + \delta\mathbf{c}||_p \\ &= ||\mathbf{L}^*(\mathbf{g}) - \mathbf{c}^{true}||_p + \delta\mathbf{L}^*(\mathbf{g}) \\ &= O^t(\mathbf{g}) + O^p(\mathbf{g}), \end{aligned} \tag{5}$$

where $O^p(\mathbf{g})$, $O^t(\mathbf{g})$ stand respectively for the perturbed and noise-free cost functions, and $\delta\mathbf{c}$ for the noise in modeling induced by the noise in data. For instance, in the case where the Euclidean norm is used ($p = 2$) we have:

$$\delta\mathbf{L}^*(\mathbf{g}) = (\mathbf{L}^*(\mathbf{g}) - \mathbf{c}^{true})^T \delta\mathbf{c} + \delta\mathbf{c}^T \delta\mathbf{c}. \quad (6)$$

Therefore it is straight forward to conclude that $O^p(\mathbf{g})$ and $O^t(\mathbf{g})$ will achieve their corresponding minima (in case they exist and are unique) in different $\mathbf{g}$ models. Besides, the classifier $\mathbf{L}^*$ is built ad-hoc and it is just a mathematical abstraction used to discover the genes that are involved in the phenotype discrimination, but it is not the reality itself.

For the reasons mentioned above uncertainty analysis of the oncogenomic classification is always needed. The ensemble of genetic signatures that predict the class of the samples with a similar high accuracy will be therefore related to the disease development. These sets of genes will serve to explore the genetic pathways that are responsible for the illness development. Moreover, from this analysis it is possible to deduce the correlation networks, that is how the most discriminatory genes are interrelated taking into their differential expressions.

3 Biomedical Robots

We define a Biomedical robot as the set of bioinformatic algorithms coming from applied mathematics, statistics and computer science that are capable of analyzing dynamically (as a function of time) very high dimensional data, discovering knowledge, generating new biomedical working hypothesis, and supporting medical decision making with its corresponding uncertainty assessment. In this definition the data does not need to be of genetic type or even homogeneous, that is, several types of medical data could be mixed for decision-making purposes. In the present case the data come from microarray differential expression analysis, between individuals that develop the illness and others that do not.

Generating new working hypothesis in this case includes the analysis of biomarkers and mechanisms of action involved in the illness development, and finding existing drugs that could target the main actionable genes. Also, a benefit of this approach could be the design of intelligent systems to support medical doctors/researchers in the decision making process of new incoming uncatalogued samples to decide questions relative to their diagnosis, treatment and prognosis before any decision is taken. These techniques can help for instance in segmenting patients with respect to drug response based on genetic signatures, to predict the development of induced toxicities, to predict the surgical risk, etc, among many different applications that we can imagine.

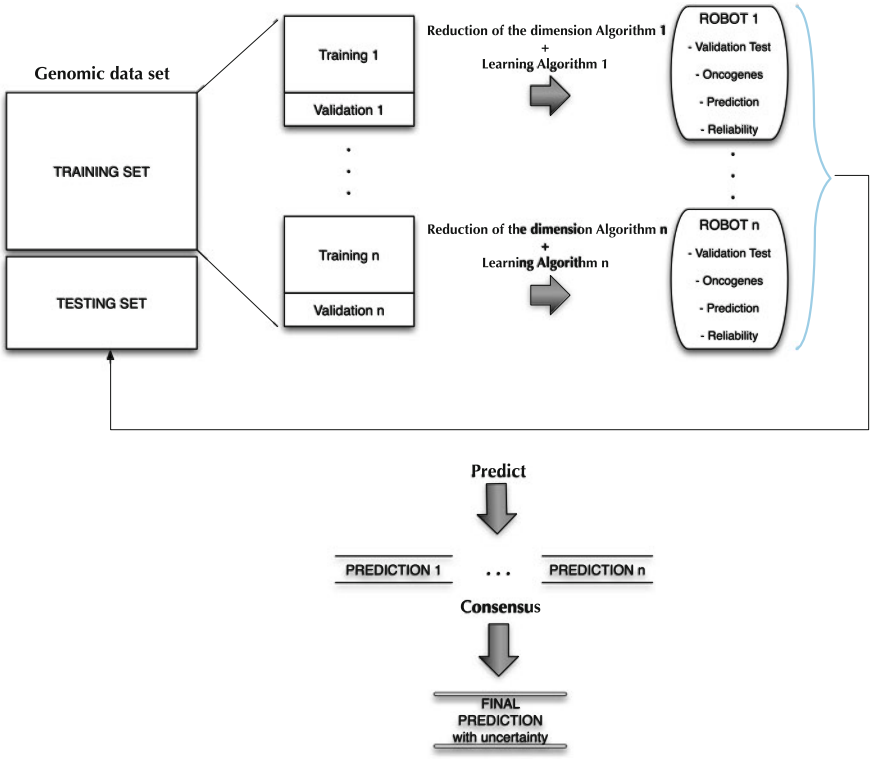


Fig. 3 Biomedical robot scheme

A scheme of a Biomedical robot is depicted in Fig. 3. From a training data set we built N_r robots. The robot in the present case is in fact a classifier characterized by its small scale genetic signature **g** and its corresponding set of parameters **p** needed to perform the classification of samples. These robots will be deduced from the dataset by applying different supervised filter feature selection methods and dimensional reduction algorithms. Each robot will also be characterized by its predictive accuracy according to the classification cost function $O(\mathbf{g})$. The design of the cost function is important because the set of genetic signatures found depend on that design. In this paper we have used a LOOCV (Leave-One-Out-Cross-Validation) average error because it makes use of most of the sample information that is available, and also mimics the process that we will find in real practice.

It is important to remark that we are not only interested in building a black-box predicting approach, but also being able of inferring the mechanism of action and the genetic pathways involved. The final decision approach is as follows: given a new incoming sample, each of the equivalent robots will perform a prediction. A final prediction with its uncertainty assessment will be given using all these predictions and a consensus strategy (majority voting). This approach has been used by Fernández-Martínez and Cernea [3] in the face recognition problem, obtaining very high accuracies and stable results. Ensemble classification is related to majority vote

decisions that are based on Condorcet's jury theorem, which is a political science theorem about the probability of a given group of individuals arriving at a correct decision. In the context of biomedical robots and ensemble learning, it implies that the probability of being correct for a majority of independent voters is higher than the probability of any of the individual voters, and tends to 1 when the number of voters (or weak classifiers) tends to infinity. In this case the weak classifiers are any of the biomedical robots of the ensemble that have a high predictive accuracy.

An important question in this design is how to measure the discriminatory power of a gene with respect to a given phenotype. There is not a unique answer to this question. Several methods exist to assign the discriminatory power of a gene: Fold-Change, p-value, Fisher's ratio, entropy, mutual information, percentile distance between statistical distributions, etc. Generally speaking high discriminatory genes are those that have very different distributions in both classes (in a binary problem) and whose expression remains quite stable or homogeneous within each class.

The algorithm used in this paper is similar to the one that was presented in [1] and [12], and consists in several steps (see Fig. 3):

1. Applying one (or several) filter feature selection methods to find different lists of high discriminatory genes.
2. Finding the predictive accuracy of the ranked list of genes by Leave-One-Out-Cross-Validation (LOOCV) using a k-Nearest-Neighbor (k-NN) classifier. Others classifiers can be also used.
3. Obtaining different biomedical robots from these lists with their associated predictive accuracy. One possibility is applying Backwards Feature Elimination.
4. Selecting robots above a certain predictive accuracy (or below a given error tolerance) and performing the consensus prediction through a majority voting.

According to the definitions stated in (1), (2), (3), (4) we can formally define a biomedical robot as the set of classifiers:

$$L_{tol} = \{L^*(\mathbf{g}_k) : k = 1, \ldots, \mathrm{N}_r\}, \tag{7}$$

fulfilling that the number of misclassified samples is less than a given bound tol. The oncogenomic problem with uncertainty estimation consists in, giving an incoming sample $\mathbf{s}_{new}$, apply the set of Biomedical robots L_{tol} with predictive accuracy higher than $(100 - tol)$ %, and performing the consensus classification. Following the rules of importance sampling, and supposing that the uncertainty analysis was correctly performed, then the probability of $\mathbf{s}_{new}$ to belong to class c_1 is calculated as the number of robots that predicted the sample to belong to class c_1 divided by the total number of selected robots in the set L_{tol}.

In this paper we apply this concept to the analysis of three kinds of diseases: Cancer (CLL), Neurodegenerative (ALS) and Rare diseases (IBM-PM). Although the concept is theoretically correct, before applying it to omics data, we have analyzed its robustness and stability. For such purpose, we have generated a synthetic data set with different types of noise and performing sensitivity analysis of the methodology that has been proposed.

3.1 Sensitivity Analysis

We have generated different synthetic data sets using three types of noise: additive Gaussian noise, lognormal noise, and noise in the class assignation. These last two types belong to the category of non-Gaussian noise, since they are multiplicative and systematic random noises. The method consists in building a synthetic dataset with a predefined number of differentially expressed discriminatory genes, and subsequently introducing different types of noise, and determining the predictive accuracy (Acc) as a function of the number of used robots ($\#R$). The synthetic dataset was built using the OCplus package available for The Comprehensive R Archive Network [11]. Table 1 shows the results obtained for the sensitivity analysis, where δ_c, δ_g and δ_{lg} represent the level of noise imputed in these 3 cases.

The results can be summarized as follows:

- The predictive accuracy using the biomedical robots systematically decreases when a higher level of the noise is added in the class assignment. This result is very interesting and suggests that the use of the biomedical robots do not dramatically over fit the expression data in order to fulfill a wrong class assignment.
- The predictive accuracy of the biomedical robots generally improves when Gaussian and non-Gaussian noise is added to the expression data, meaning that the biomedical robots are robust with respect to the presence of noise in the expressions. This result also suggests that noise acts as regularization, as it has been theoretically proved by Fernández-Martínez et al. [5, 6] in inverse problems. An additional corollary of this fact is that working with raw data would have some benefits with respect of working with filtered microarray data. Filtering is a step that it is usually performed in microarray data and might have important consequences with respect to the conclusions that are obtained. Future research will be devoted to this important subject.

Therefore we can conclude that if the biomedical robots are unable to improve the accuracy of the best prediction, the dataset must have some wrong class assignment that prevents achieving a perfect classification. Other possibility is that parameterization of the samples is incorrect, that in the present case would mean that none of the genes that have been measured bring enough information to achieve a good phenotype discrimination.

Table 1 Noise results

Class assignation			Gaussian			Log Gaussian		
$\delta_c(\%)$	$Acc_c(\%)$	$\#R_c$	$\delta_g(\%)$	$Acc_g(\%)$	$\#R_g$	$\delta_{lg}(\%)$	$Acc_{lg}(\%)$	$\#R_{lg}$
10	90.18	98	20	100.00	11	20	100.00	43
15	87.12	3	30	99.39	7	30	100.00	14
20	80.98	1	40	98.77	24	40	88.96	8
25	77.30	10	50	98.77	71	50	100.00	52
30	73.62	43	60	100.00	1	60	100.00	11

3.2 Chronic Lymphocytic Leukemia

B-cell Chronic Lymphocytic Leukemia (CLL) is a complex and molecular heterogeneous disease, which is the most common adult Leukemia in western countries. In our cohort DNA analyses served to distinguish two major types of CLL with different survival times based on the maturity of the lymphocytes, as discerned by the Immunoglobulin Heavy chain Variable-region (IgVH) gene mutation status [7]. In this first example we had at disposal a microarray data set consisting of 163 samples and 48807 probes. The IgVH mutational status was predicted with 93.25 % accuracy using small-scale signature composed by 13 genes: LPL (2 probes), CRY1, LOC100128252(2 probes), SPG20 (2 probes), ZBTB20, NRIP1, ZAP-70, LDOC1, COBLL1 and NRIP1.

Table 2 shows the results of applying the methodology of biomedical robots to this problem. In this case the highest prediction accuracy obtained by the set of biomedical robots equal the accuracy provided by the best robot (93.25 %). This result implies that some samples are behavioral outliers or might be misclassified. This happened with 11 samples. Recently it has been suggested the possibility of existence of a third group of CLL patients.

The pathway analysis deduced from the biomedical robots has revealed the importance of the ERK signaling super pathway that includes ERK signaling, ILK signaling, MAPK signaling, Molecular Mechanisms of cancer and Rho Family GTPases pathway. These pathways control Proliferation, Differentiation, Survival and Apoptosis. Also, other important pathways found were Allograft Rejection, the Inflammatory Response Pathway, CD28 Co-stimulation, TNF-alpha/NF-kB Signaling Pathway, Akt Signaling, PAK Pathway and TNF Signaling. The presence of some of these pathways opens the hypothesis of viral infection as a cause for CLL.

Table 2 CLL, IBM and PM and ALS results

CLL			IBM and PM			ALS		
Acc(%)	*tol*	#*R*	*Acc*(%)	*tol*	#*R*	*Acc*(%)	*tol*	#*R*
92.64	85.89	488	87.50	82.50	223	84.71	83.53	547
92.64	86.50	487	87.50	85.00	159	85.88	84.71	441
92.64	89.57	486	90.00	87.50	138	87.06	85.88	241
92.64	90.18	479	90.00	90.00	71	88.24	87.06	197
92.64	90.80	446	92.50	92.50	32	90.59	88.24	134
92.64	91.41	373	100.00	95.00	2	91.76	89.41	96
93.25	92.02	255	97.50	97.50	1	90.59	90.59	54
93.25	92.64	120				92.94	91.76	32
93.25	93.25	22				95.29	92.94	20
						94.12	94.12	10
						95.29	95.29	6
						96.47	96.47	1

3.3 Inclusion Body Myositis and Polymiositis

Myositis means muscle inflammation, and can be caused by infection, injury, certain medicines, exercise, and chronic disease. Some of the chronic, or persistent, forms are idiopathic inflammatory myopathies whose cause is unknown. We have dealt with the IBM/PM dataset published by Greenberg et al. [8]. The data consisted in the microarray analysis of 23 patients with IBM, 6 with PM and 11 samples corresponding to healthy control. The classification of the IBM+PM versus control has obtained a predictive accuracy of 97.5 % using a reduced base with only 20 probes.

Table 2 shows the results using the biomedical robots methodology. In this case we are able to hit the 100 % of the samples with two robots, improving the results of the best robot. The genes belonging to the highest predictive small-scale genetic signature are HLA-C (3 probes), HLA-B (4 probes), TMSB10, S100A6, HLA-G, STAT1, TIMP1, HLA-F, IRF9, BID, MLLT11 and PSME2. It can be observed the presence of different HLA-x genes of the major histocompatibility. Particularly the function of the gene HLA-B would explain alone the genesis of IBM: "HLA-B (major histocompatibility complex, class I, B) is a human gene that provides instructions for making a protein that plays a critical role in the immune system. HLA-B is part of a family of genes called the human leukocyte antigen (HLA) complex. The HLA complex helps the immune system distinguish the body's own proteins from proteins made by foreign invaders such as viruses and bacteria".

The analysis of biological pathways has revealed the importance of viral infections, mainly in IBM patients: Allograft Rejection, Influenza A, Class I MHC Mediated Antigen Processing and Presentation, Staphylococcus Aureus Infection, Interferon Signaling, Immune Response IFN Alpha/beta Signaling Pathway, Phagosome, Tuberculosis, Cell Adhesion Molecules (CAMs), Epstein-Barr Virus Infection, and TNF Signaling. It can be observed several viral infections appearing in this list. Interesting, it has been found that 75 % of the cases of viral myositis are due to Staphylococcus Aureus infection.

Finally, Fig. 4 shows the correlation network of the most discriminatory genes found in this analysis. This is an interesting tool to understand how the most discriminatory genes are interrelated and regulate the expression of other genes. The head of graph is formed by the genes that are highly correlated to the class array.

3.4 Amyotrophic Lateral Sclerosis

ALS is a motor neuron disease that characterized by stiff muscles, muscle twitching, and gradually worsening weakness. Between 5 and 10 % of the cases are inherited from a relative, and for the rest of cases, the cause is still unknown [10]. It is a progressive disease that the average survival from onset to death is three to four years, in which most of them die from a respiratory failure. There is no cure yet.

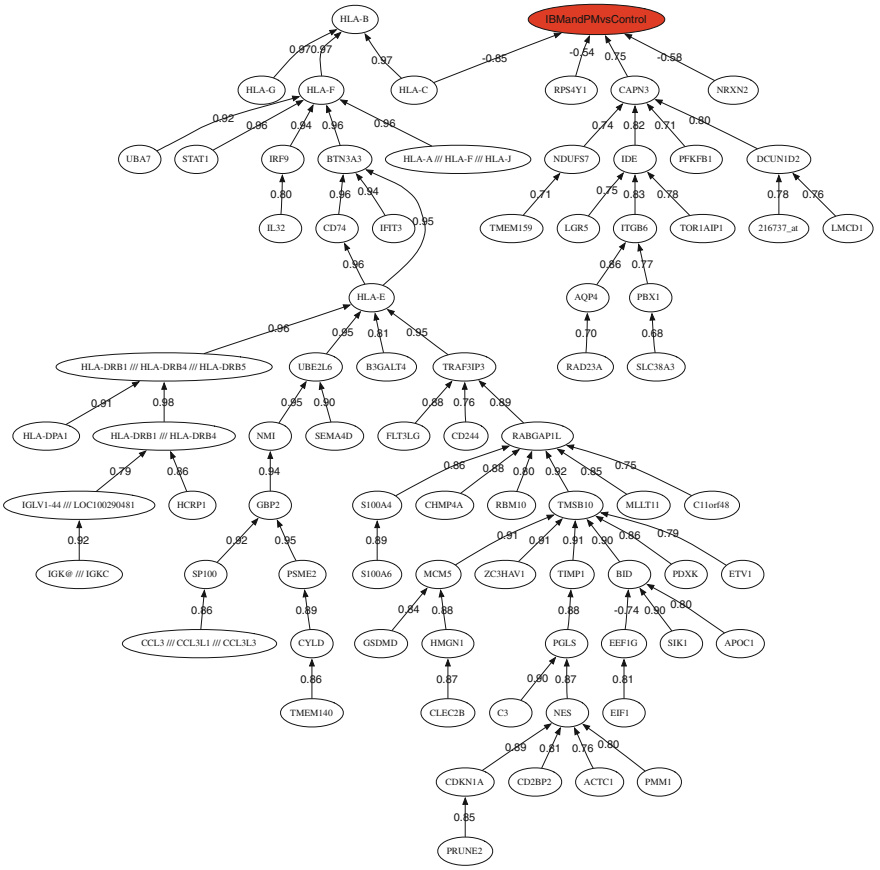


Fig. 4 Correlation network for IBM and PM

We reinterpreted a dataset published by Lincecum et al. [9] consisting of 57 ALS cases and 28 healthy controls. The best result yields an accuracy of 96.5 % with small scale signature involving the following genes: CASP1, ZNF787 and SETD7. Table 2 shows the results of applying this methodology to this problem. The biomedical robots in this case did not improve this prediction. The pathway analysis has revealed the importance of the GPCR Pathway, RhoA Signaling Pathway, EPHB Forward Signaling, EphrinA-EphR Signaling, EBV LMP1 Signaling, and Regulation of Microtubule Cytoskeleton. These pathways have different important signaling roles and suggest a possible link to the Epstein-Barr virus (EBV).

4 Conclusion

In this paper we have introduced the concept of biomedical robots and its application to the analysis of cancer, rare and neurodegenerative diseases in a project that we have named FINISTERRAE: the use of genomics as a crucial tool for pharmacological discovery. The concept of biomedical robot is based in exploring the uncertainty space of the phenotype classification problem involved, and using the structure of the uncertainty space to adopt decisions and inferring knowledge. The application to a synthetic dataset has shown that the biomedical robots are robust in the presence of different kind of noise in the expressions and class-assignment. The synthetic experiment has proved that the biomedical robots do not overfit the expression data to justify a wrong class assignment. Finally we have shown the application of this novel concept to 3 different illnesses: CLL, IBM-PM and ALS, proving that it is possible to infer both, high discriminatory small scale signatures and the description of the biological pathways involved.

References

1. Fernández-Martínez, J.L., Luaces, O., del Coz, J., Fernández, R., Solano, J., Nogués, E., Zanabilli, Y., Alonso, J., Payer, A., Vicente, J., et al.: On the prediction of Hodgkin lymphoma treatment response. In: Clinical and Translational Oncology, pp. 1–8 (2015)
2. Fernández-Martínez, J.L., Fernandez Muniz, M.Z., Tompkins, M.J.: On the topography of the cost functional in linear and nonlinear inverse problems. Geophysics **77**(1), W1–W15 (2012)
3. Fernández-Martínez, J.L., Cernea, A.: Exploring the uncertainty space of ensemble classifiers in face recognition. Int. J. Pattern Recognit. Artif. Intell. **29**(03), 1556002 (2015)
4. Fernández-Martínez, J.L., Fernández-Muñiz, Z., Pallero, J., Pedruelo-González, L.M.: From Bayes to tarantola: new insights to understand uncertainty in inverse problems. J. Appl. Geophys. **98**, 62–72 (2013)
5. Fernández-Martínez, J.L., Pallero, J., Fernández-Muñiz, Z., Pedruelo-González, L.M.: The effect of noise and Tikhonov regularization in inverse problems. Part II: the nonlinear case. J. Appl. Geophys. **108**, 186–193 (2014)
6. Fernández-Martínez, J.L., Pallero, J.L.G., Fernandez-Muniz, Z.: The effect of noise and tikhonov regularization in inverse problems. Part I: the linear case. J. Appl. Geophys. **108**, 176–185 (2014)
7. Ferreira, P.G., Jares, P., Rico, D., Gómez-López, G., Martínez-Trillos, A., Villamor, N., Ecker, S., González-Pérez, A., Knowles, D.G., Monlong, J., et al.: Transcriptome characterization by RNA sequencing identifies a major molecular and clinical subdivision in chronic lymphocytic leukemia. Genome Res. **24**(2), 212–226 (2014)
8. Greenberg, S., Bradshaw, E., Pinkus, J., Pinkus, G., Burleson, T., et al.: Plasma cells in muscle in inclusion body myositis and polymyositis. Neurology **65**(11), 1782–1787 (2005)
9. Lincecum, J.M., Vieira, F.G., Wang, M.Z., Thompson, K., De Zutter, G.S., et al.: From transcriptome analysis to therapeutic anti-CD40L treatment in the SOD1 model of amyotrophic lateral sclerosis. Nature Genet. **42**(5), 392–399 (2010)
10. National Institute of Neurological Disorders and Stroke: Motor neuron diseases. Fact Sheet (2010)
11. Pawitan, Y., Ploner, A.: OCplus: Operating characteristics plus sample size and local FDR for microarray experiments, R package, version 1.40.0

12. Saligan, L.N., Fernández-Martínez, J.L., et al.: Supervised classification by filter methods and recursive feature elimination predicts risk of radiotherapy-related fatigue in patients with prostate cancer. Cancer Inform. **13**, 141–152 (2014)
13. Strausberg, R.L., Simpson, A.J., Old, L.J., Riggins, G.J.: Oncogenomics and the development of new cancer therapies. Nature **429**(6990), 469–474 (2004)

Generation of Power Law: Maximum Entropy Framework and Superstatistics

Karmeshu, Shachi Sharma and Sanjeev Kumar

Abstract The ubiquitous nature of power law in a variety of systems is a well-known phenomenon. We describe various mechanisms responsible for emergence of power law behavior. Maximum entropy based framework with appropriate moment constraints provides a useful approach for generating power law. It is found that maximization of Shannon entropy with either geometric or shifted geometric mean yields power tail behavior. Tsallis entropy maximization with arithmetic mean constraint also results in long tail distributions. A new framework based on superstatistics is discussed which also has the capability to generate heavy tail distributions. Illustrative examples from communication systems, computational neuroscience, Brownian motion in state dependent random force and social systems are briefly discussed.

Keywords Heavy tail · Communication networks · Spiking neurons · Inter spike interval distribution · Learning-forgetting

1 Introduction

It is being increasingly observed that systems ranging from physical to engineering to biological depict power law tails [7, 18, 23, 34, 37, 45]. In addition to these systems, man made systems like internet [8, 29, 49] and natural phenomenon related to intensity of earthquake [14], turbulence [36], occurrence of floods [26] are also known to exhibit heavy long tail behavior. Existence of rare events is a characteristic feature of distributions with heavy tail, implying that rare events are the ingredient of power law [3]. Several mechanisms have been discussed in the literature to explain the underlying mechanisms which are responsible for generation of such a phenomenon. Interestingly, this behavior is also seen in social science [38], Zipf's law for statistical

Karmeshu (✉) · S. Kumar
School of Computer and Systems Sciences, Jawaharlal Nehru University, New Delhi, India
e-mail: karmeshu@gmail.com

S. Sharma
IBM Research Laboratory, New Delhi 110070, India

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_4

structure of language [27, 28, 51], financial market [11], city size distribution [25] etc. Mathematically, a continuous random variable X with probability density function $f(x)$ is said to follow power law when

$$f(x) = Ax^{-\alpha}, \qquad x \geq x_{min} \tag{1}$$

where A is normalization constant. A discrete random variable X with probability mass function $p(x)$ obeys power law when

$$P(X = x) = p(x) = Cx^{-\alpha}, \qquad x = 1, 2, \ldots \tag{2}$$

where C is normalization constant.

Besides systems showing power law, there are many other systems that depict $1/f$ noise [2], a phenomena also intimately connected with power law behavior. As pointed out by Milotti [30], "the outstanding feature of $1/f$ noise is that it is scale invariant i.e. it looks the same for any choice of frequency or time unit,". Montroll and Shlesinger [31, 32] have argued that log normal distribution over a certain range can mimic $1/f$ noise. Many situations can be described by a multiplicative mechanism where it would be possible to show by invoking central limit theorem in probability that the resulting distribution follows a log normal distribution. For large variance, it is easy to see that the log normal density function can mimic power law tail. The frequent presence of log normal distribution in various scenarios is on account of the process being characterized by multiplicative mechanism. The appreciation of this aspect lends justification to the log normal distribution observed in many other situations which may appear to be totally unconnected but are moderated by an implicit multiplicative mechanism. For example, we consider problems related to income distribution [24], fading phenomenon in communication network [17], ultra sound fading back scattered envelope in ultrasound imaging [13], scientific productivity of researchers [43] are totally unconnected, the underlying mechanism is multiplicative.

In this paper, we describe various mechanisms for generation of power law. We consider application of maximum entropy principle (MEP) to communication networks. Depending on the choice of entropy measures, we show that emergence of heavy tail probability distribution is a consequence of appropriate moments of variables of interest. Section 2 presents a brief introduction of extensive Shannon entropy and non-extensive Tsallis entropy measures. This section also discusses the maximum entropy principle. Section 3 provides MEP based mechanisms for generating power law in the context of communication networks. Section 4 provides an overview to obtain power law by representing it through a mixture of exponentials. In Sect. 5, we discuss the power law behavior in Brownian motion of a particle in presence of state dependent random force. Section 6 explains power law in social systems. As an illustration of superstaistical framework in computational neuroscience, we describe in Sect. 7 generation of power law in inter-spike interval distribution when an ensemble of neurons group together and fire together. Last section contains concluding remarks.

2 Entropy Measures and Maximum Entropy Principle

Entropy can be regarded as a measure of uncertainty or randomness in a system. The word entropy is well-known in statistical thermodynamics and it measures amount of disorder. Shannon [40] in the context of mathematical theory of communication, introduced the concept of average amount of uncertainty associated with a probabilistic system. Recognizing the analogy with statistical thermodynamics, Shannon called this measure of average uncertainty (or average amount of information) as entropy. For a discrete random variable X which can take n values such that $P(X = x_i) = p_i, i = 1, 2, \ldots, n$, Shannon entropy of random variate X is defined as

$$H(X) = -\sum_{i=1}^{n} p_i \ln_2 p_i \tag{3}$$

Accounting for an impossible event $p = 0$, one takes $0 \log 0 = 0$. In case of independent random variables X and Y, the joint entropy $H(X, Y)$ equals the sum of entropy of independent random variables [21]

$$H(X, Y) = H(X) + H(Y) \tag{4}$$

and Shannon entropy satisfies properties of extensive systems.

Tsallis Entropy: Tsallis [46] proposed non-extensive entropy with parameter q

$$S_q = K \frac{1 - \sum_i p_i^q}{q - 1}, \qquad i = 1, 2, \ldots \tag{5}$$

This entropy for two independent systems A and B satisfies quasi-additivity property [12],

$$S_q(A, B) = S_q(A) + S_q(B) + (1 - q) S_q(A) S_q(B) \tag{6}$$

Tsallis entropy is non-extensive which in the limit $q \to 1$ yields additivity property and the entropy measure (5) reduces to Shannon entropy [12].

MEP and Shannon Entropy: Jaynes' [15] proposed that the most objective probability distribution of a system consistent with partial information, usually available in the form of moments of random variables describing the system, is the one which maximizes entropy. This principle is known as Maximum Entropy Principle (MEP). Based on Shannon entropy, MEP has been applied to study variety of queueing systems as well [21, 22]. When the knowledge about first moment viz. queue size is available i.e.

$$\sum_{i=1}^{\infty} i p_i = A \tag{7}$$

Maximization of Shannon entropy (3) by Lagrange's method subject to (7) and normalization constraint

$$\sum_{i=0}^{\infty} p_i = 1 \tag{8}$$

yields

$$p_i = \frac{e^{-\beta i}}{\sum_{i=0}^{\infty} e^{-\beta i}}, \qquad i = 1, 2, \ldots \tag{9}$$

where β is Lagrange's parameter. Equation (9) on substitution $\rho = e^{-\beta}$ gives result for equilibrium distribution of M/M/1 queueing system

$$p_i = (1 - \rho)\rho^i, \qquad i = 1, 2, \ldots \tag{10}$$

It is important to note that arithmetic mean is the characterizing moment to describe M/M/1 queueing system through MEP framework. Further, it is worth noting that maximization of Shannon entropy results in exponential class of distributions.

In next section, we discuss how these two entropy measures enable to generate power law in communication networks.

3 MEP and Power Law

We discuss MEP based approach and methods to generate power law. For the purpose of illustration, we consider a single server queueing system driven by power law input. Such a queueing system emulates a node of the communication network and for performance study it is important to examine the behavior of such a queueing system. Our objective is to obtain analytical closed form solution for such a queueing system driven by input having power law distribution.

3.1 MEP and Tsallis Entropy

Kameshu and Shachi [19] have studied the applicability of Tsallis entropy to study power law behavior in broadband communication network. It is known that the distributions showing power law may not possess finite first moment. It can be seen that the mean value or first moment of power law distributed random variable X may not exist. In such a scenario, fractional moment of the variable of interest is assumed to exist. The kth fractional moment can be defined as

$$\sum_{i=1}^{\infty} i^k p_i = A_k, \qquad 0 < k < 1 \tag{11}$$

When Tsallis entropy (5) is maximized subject to fractional moment constraint (11) and normalization (8), the optimization problem can be stated as follows:

$$Max \;\; S_q = K \frac{1 - \sum_i p_i^q}{q - 1}, \qquad n = 1, 2, \ldots \tag{12}$$

subject to

$$\sum_{i=1}^{\infty} i^k p_i = A_k, \qquad 0 < k < 1 \tag{13}$$

and

$$\sum_{i=0}^{\infty} p_i = 1 \tag{14}$$

Solving constrained optimization problem, the Lagrangian function is,

$$L_q = \frac{S_q}{K} - \alpha(1 - \sum_{i=0}^{\infty} p_i) + \alpha\beta(q-1)(A_k - \sum_{i=1}^{\infty} i^k p_i) \tag{15}$$

where α and β are Lagrange's parameters. Differentiating L_q with respect to p_i and equating the result to zero yields

$$p_i = \frac{\left[1 + \beta(1-q)i^k\right]^{\frac{1}{q-1}}}{\sum_{i=0}^{\infty}\left[1 + \beta(1-q)i^k\right]^{\frac{1}{q-1}}}, \qquad q > 1 - k \tag{16}$$

which asymptotically follows power law i.e.

$$p_i \sim i^{-k/(1-q)} \tag{17}$$

Further, it is also observed that buffer over flow probability exhibits power law

$$P(i > x) \sim x^{-\frac{k}{1-q}+1}, \qquad \frac{k}{1-q} > 1 \tag{18}$$

In the limit $q \to 1$, the overflow probability becomes

$$P(i > x) \sim e^{-\beta x^k} \tag{19}$$

corresponding to tail of Weibull distribution. This is similar to results derived by Norros [35] in the context of fractional Brownian motion input in broadband networks. In case of finite buffer system [41], loss probability also asymptotically follows power law. Thus, Tsallis entropy maximization subject to fractional moment constraints provides a useful method to study power law in communication networks.

3.2 MEP, Geometric Mean and Shannon Entropy

It has been observed that power law emerges if average of logarithm of the variables of interest is specified. Also, since the power law is connected to log normal distribution which is symmetrical and unimodal on log scale, it can thus be argued that geometric mean may be good measure of central tendency for characterizing distributions with power law. In this light, Singh and Karmeshu [44] in a recent paper have revisited Shannon entropy to capture power law. They note the wider applicability of maximization of Shannon entropy to obtain power law in queueing systems when either geometric mean or shifted geometric mean is prescribed as constraints.

We consider moment constraint in the form of geometric mean is available. Maximization of Shannon entropy (3) subject to geometric mean constraint

$$\sum_{i=0}^{\infty} p_i \, log \, i = log \, G \tag{20}$$

normalization constraint

$$\sum_{i=0}^{\infty} p_i = 1 \tag{21}$$

and system utilization

$$h(i) = \begin{cases} 0, & i = 0 \\ 1 & i \neq 1 \end{cases} \tag{22}$$

results in

$$p_i = e^{-\alpha - \beta h(i)} i^{-\gamma}, \qquad i \geq 1 \tag{23}$$

where α, $\beta$and γ are Lagrange's parameters that can be calculated from the constraints. The probability distribution $\{p_i, i \geq 1\}$ can also be re-written as

$$p_i = (1 - p_0) i^{-\gamma} \zeta(\gamma), \qquad i \geq 1 \tag{24}$$

which clearly follows power law for large i

$$p_i \sim i^{-\gamma}, \qquad \gamma > 1 \tag{25}$$

Singh and Karmeshu [44] have shown that if geometric mean with a shift parameter **a** is available i.e.

$$Q = \left(\prod_i (i + a) \right)^{1/i} - a \tag{26}$$

then maximization of Shannon entropy (3) subject to (8) and (26) gives

$$p_i = \frac{(i + a)^{\gamma}}{\sum_i (i + a)^{\gamma}}, \qquad i = 1, 2, \ldots \tag{27}$$

Asymptotically, this results in power law distribution

$$p_i \sim i^{-\gamma}, \qquad \gamma > 1 \tag{28}$$

The probability that large buffer size x is exceeded, is given by

$$P(i > x) \sim x^{-\gamma+1} \tag{29}$$

On introducing parameters $s \neq 1$ such that $a = 1/(s-1)\beta$ and $\gamma = \alpha/s - 1$, (27) becomes

$$p(i; s) = \frac{[1 + (s-1)\beta i]^{\frac{\alpha}{1-s}}}{\sum_{i=0}^{\infty} [1 + (s-1)\beta i]^{\frac{\alpha}{1-s}}} \tag{30}$$

which is similar to (16) when $k = 1$. Thus, the equivalence between maximum Shannon and Tsallis entropy frameworks holds by appropriate choice of constraints. In other words, Tsallis entropy parameter q and shifted geometric mean parameter **a** get connected.

4 Power Law: Weighted Mixture of Exponentials

In the context of video traffic and internet traffic, Feldman and Whitt [9] suggest that it is possible to approximate long tail distribution, like Pareto, Weibull, log normal by a weighted mixture of the exponentials in the region of interest. One can write

$$f(x) = \sum_{i=1}^{k} w_i e^{-\lambda_i x} \tag{31}$$

where $f(x)$ is distribution following power law, k is the number of exponentials and $w_i \geq 0$ are weights such that $\sum_i^k w_i = 1$. Their finding is based on the Bernestein theorem [5] which states that every completely monotone pdf f is a mixture of exponential pdfs

$$f(x) = \int_0^\infty \lambda e^{-\lambda t} dG(\lambda), \qquad x > 0 \tag{32}$$

for some CDF G.

Consider Levy distribution having the form

$$G_r(x) = C_r \Big[1 - (1-r)\frac{x}{\lambda}\Big]^{1/(1-r)} \tag{33}$$

which in the limit $r \rightarrow 1$ converges to

$$G_{r=1} = g(x) = c.e^{-\frac{x}{\lambda}} \tag{34}$$

Wilk and Wlodarczyk [48] prove that when the distribution $f\ (1/\lambda)$ of random variable $1/\lambda$ is gamma, then exponential distribution (34) leads to Levy distribution (33). Let $1/\lambda_0$ be the mean value of fluctuations then

$$G_r(x;\lambda_0) = C_r\left(1 + \frac{x}{\lambda_0}\frac{1}{\alpha}\right)^{-\alpha} = C_r \int_o^\infty e^{-x/\lambda} f\left(\frac{1}{\lambda}\right)\, d\left(\frac{1}{\lambda}\right) \tag{35}$$

where $\alpha = 1/(r-1)$. Using Euler gamma function in (35), one can immediately derive the distribution of $f(1/\lambda)$ as gamma distribution

$$f\left(\frac{1}{\lambda}\right) = \frac{1}{\Gamma(\alpha)}(\alpha\lambda_0)\left(\frac{\alpha\lambda_0}{\lambda}\right)^{\alpha-1} e^{-\frac{\alpha\lambda_0}{\lambda}} \tag{36}$$

In other words, the distribution of λ is given by inverse gamma distribution.

5 Brownian Motion in State Dependent Random Force: Emergence of Power Law

Brownian motion in state dependent random force can be modeled by stochastic differential equation (SDE) with additive and multiplicative noise sources. The motion of the particle is described by Langevin equation which is given as

$$\frac{du(t)}{dt} = -[\beta + \alpha(t)]u(t) + \frac{1}{m}F(t) \tag{37}$$

where $u(t)$ is the velocity of the Brownian particle, m the mass and $F(t)$ is random driving force. The damping is stochastically perturbed with mean part β and stochastic part $\alpha(t)$. It is assumed that both $\alpha(t)$ and $F(t)$ are independent white noise sources such that

$$\begin{aligned} E[\alpha(t)] &= 0, \qquad E[\alpha(t_1)\alpha(t_2)] = 2\lambda\delta(t_1 - t_2) \\ E[F(t)] &= 0, \qquad E[F(t_1)F(t_2)] = S\delta(t_1 - t_2) \end{aligned} \tag{38}$$

The Fokker-Planck equation corresponding to SDE (37) is

$$\frac{\partial P(u,t)}{\partial t} = \frac{\partial^2}{\partial u^2}\left[\left(\lambda u^2 + \frac{S}{2m^2}\right)P(u,t)\right] + \frac{\partial}{\partial u}[(\beta - \lambda)uP(u,t)] \tag{39}$$

Following Wong [50], Karmeshu [16] has obtained explicit time-dependent solution as given by

$$\begin{aligned} P(u,t;u_0) = &\sqrt{\left(\frac{2m^2\lambda}{S}\right)}\left(a + \frac{2m^2\lambda}{S}u^2\right)^{(-\alpha+\frac{1}{2})} \\ &\left\{\frac{1}{\pi}\sum_{n=0}^{N}\frac{\alpha - n}{n!\Gamma(2\alpha + 1 - n)}e^{-n\lambda(2\alpha-n)t}\theta_n\left(\sqrt{\left(\frac{2m^2\lambda}{S}\right)}u_0\right)\right. \\ &\theta_n\left(\sqrt{\left(\frac{2m^2\lambda}{S}\right)}u\right) + \frac{1}{2\pi}\int_0^\infty e^{-\lambda(\alpha^2+\mu^2)t} \\ &\left[\psi\left(\mu, \sqrt{\left(\frac{2m^2\lambda}{S}\right)}u_0\right)\psi\left(-\mu, \sqrt{\left(\frac{2m^2\lambda}{S}\right)}u\right)\right. \\ &\left.\left.+\psi\left(-\mu, \sqrt{\left(\frac{2m^2\lambda}{S}\right)}u_0\right)\psi\left(-\mu, \sqrt{\left(\frac{2m^2\lambda}{S}\right)}u\right)\right]d\mu\right\} \end{aligned} \tag{40}$$

The pdf $P(u,t)$ in the limit of large t yields stationary pdf given by

$$P(u,\infty) = P(u) = \sqrt{\left(\frac{2m^2\lambda}{S}\right)}\frac{\Gamma(\alpha + \frac{1}{2})}{\Gamma\left(\frac{1}{2}\right)\Gamma(\alpha)}\left(1 + \frac{2m^2\lambda}{S}u^2\right)^{(-\alpha+\frac{1}{2})} \tag{41}$$

For large u,

$$P(u) \sim u^{-2\alpha+1} \tag{42}$$

displaying power law behavior. For $\alpha = 1/2$, $P(u)$ reduces to Cauchy density function,

$$P(u) = \sqrt{\left(\frac{2m^2\lambda}{S}\right)}\frac{1}{\pi}\left(1 + \frac{2m^2\lambda}{S}u^2\right)^{-1} \tag{43}$$

It is known that no integral moment of Cauchy density function exist except its probability density function. The mechanism of generation of power law in the presence of both additive and multiplicative noises has also been discussed by Anteneodo and Tsallis [1].

6 Power Laws in Social Systems

There are several problems in social sciences where heavy tail distribution arise. As illustration, we discuss following two examples.

6.1 Power Law in Learning/Forgetting

The learning process entails forgetting which over time with practice declines. The rate of learning or forgetting varies across the population. The probability of forgetting declines as a person progresses during the process of learning. Murre and Chessa [33] note that probability of committing errors in a test reduces with duration of study time. They assume exponential learning curve with

$$p(t|\mu) = e^{-\mu t}, \qquad \mu > 0, t \geq 0 \tag{44}$$

This may reflect fundamental cognitive process. Murre and Chessa [33] assume that learning rate μ of individual participants follow gamma probability distribution $g(\mu)$. Averaging over random variable μ, yields

$$\begin{aligned} p(t) &= \int_0^\infty p(t|\mu)g(\mu)d\mu \\ &= \int_0^\infty e^{-\mu t}\frac{1}{\Gamma(a)b^a}\mu^{a-1}e^{\frac{\mu}{b}}d\mu \\ &= (1+bt)^{-a} \end{aligned} \tag{45}$$

For large t, we have

$$p(t) \sim t^{-a} \tag{46}$$

exhibiting power law. Murre and Chessa [33] argue that power law arises as a result of data aggregation.

6.2 Frequency of Causalities in a Deadly Conflict

The number of casualties during Iraq conflict is found to follow a heavy tail distribution. Clauset, Young and Gleditsch [6] note that the relatively peaceful days with low number of killed were occasionally punctuated by large number of killed. It has been observed that on some days there have been hundreds of casualties and on certain days this number went up to thousands of casualties. The rare events resulting in large number of deaths are characteristic feature of power law.

7 ISI Distribution and Power Law: Superstatistics

In recent years researchers have empirically observed power law behavior in the spiking patterns of the neuronal system. Salinas and Sejnowski [39] note the existence of high variability in the spike train data due to presence of temporal correlation in the membrane potential dynamics of the neuron. Feng [10] suggests that such high variability could be on account of correlated firing in an ensemble of similar neurons which group together and fire together. Beck [4] propose superstatistical framework which deals with the long-term stationary states of non-equilibrium systems characterized by parameter fluctuating spatiotemporally. They find that averaging over these fluctuations will lead to an infinite set of more general statistics called 'superstatistics'. Sharma and Karmeshu [42] have proposed a mechanism based on superstatistical framework for the emergence of power law behavior in the inter spike interval (ISI) data of integrate-fire (IF) neurons. The dynamics of the membrane potential of perfect IF model as obtained by diffusion approximation is governed by

$$dV = \mu dt + \sigma dW(t), \qquad V(0) = V_0 \tag{47}$$

where $W(t)$ is Wiener process representing noise in the system. The drift μ and variance σ^2, are obtained in terms of excitatory λ_e and inhibitory λ_i rates and their jump magnitudes a_e an a_i respectively. We have

$$\mu = \lambda_e a_e - \lambda_i a_i \tag{48}$$

and

$$\sigma^2 = \lambda_e a_e^2 + \lambda_i a_i^2 \tag{49}$$

The drift is taken to be nonnegative to ensure that membrane potential reaches the threshold with probability one. The first passage time (FPT) can be defined as

$$T = \inf \{t \geq 0 | V(t) > V_{th}, V(0) < V_{th}\} \tag{50}$$

FPT distribution of IF neuronal model follows inverse Gaussian distribution [47] with pdf

$$f(t) = \frac{(V_{th} - V_0)}{\sqrt{2\pi\sigma^2 t^3}} \exp\left[-\frac{(V_{th} - V_0 - \mu t)^2}{2\sigma^2 t}\right], \qquad t > 0 \tag{51}$$

where V_{th} corresponds to the threshold value and V_0 is resting potential value.

Sharma and Karmeshu [42] consider an ensemble of neurons when several nearby neurons with similar functions group together and fire together [47]. Each neuron is characterized by rates which may vary spatiotemporally throughout the population of neurons. Accordingly, we may regard the rates to be randomly distributed. Thus, the pdf $f(.)$ of the first passage time becomes a conditional pdf for a given realization of these rates, i.e.

$$f(t|\lambda_e, \lambda_i, a_e, a_i) = \frac{V_{th} - V_0}{\sqrt{2\pi(\lambda_e a_e^2 + \lambda_i a_i^2)t^3}}$$
$$\times \exp\left[\frac{(V_{th} - V_0 - (\lambda_e a_e - \lambda_i a_i)t)^2}{2(\lambda_e a_e^2 + \lambda_i a_i^2)t}\right], t > 0 \tag{52}$$

Averaging over random rates yields the pdf for ISI distribution for IF model, viz.,

$$f(t) = \int_0^\infty \int_0^\infty f(t|\lambda_e, \lambda_i, a_e, a_i) \times g(\lambda_e, \lambda_i) d\lambda_e \, d\lambda_i \tag{53}$$

For studying the effect of random rates on ISI distribution, we assume that the random rates are governed by independent gamma random variates. Averaging over the gamma distributed rates of EPSP and IPSP, the asymptotic behavior of ensemble average of first passage time distribution of IF model is given by

$$f(t) \sim t^{-\xi} \tag{54}$$

with $\xi = n + c + 1$, here n and c are scale parameters of gamma distributed rates of EPSP and IPSP. This clearly shows that the spiking pattern asymptotically follows a power law. Similarly, one can observe power law behavior in the ISI distribution of Leaky integrate and fire (LIF) model when the inverse membrane decay constant is considered to be gamma distributed [20]. Karmeshu and Sharma [20] have also shown that the emergence of power law behavior is not on account of particular distribution or parameters of the model but due to averaging process over the ensemble of several similar neurons which group together and fire together. Superstatistical framework thus provides a mechanism for generation of power law of spiking pattern of neurons.

8 Conclusion

We have discussed some possible mechanisms which can lead to emergence of power law. Maximization of different entropy measures viz. Shannon and Tsallis entropies, with appropriate moment constraints are shown to result in the generation of heavy tail probability distributions. An important aspect which needs to be underlined is that maximization of extensive entropy in conjunction with non-extensive moment constraints and vice-versa yields distributions with power law. This establishes equivalence of maximization of Shannon entropy with geometric/shifted geometric mean and maximization of Tsallis entropy with arithmetic mean. Another approach based on superstatistics also results in generation of power law distributions. An area of future enquiry would be to examine the underlying mechanisms which may provide some deeper connection between different mechanisms.

Acknowledgments The authors would like to thank Dr. Sudheer Kumar Sharma for useful suggestions.

References

1. Anteneodo, C., Tsallis, C.: Multiplicative noise: a mechanism leading to nonextensive statistical mechanics. J. Math. Phys. **44**(5194), 5194–5203 (2003)
2. Bak, P., Tang, C., Wiesenfeld, K.: Self-organized criticality: an explanation of the $1/f$ noise. Phys. Rev. Lett. **59**, 381–384 (1987)
3. Barabasi, A.L.: Bursts: The Hidden Patterns Behind Everything We Do, from Your E-mail to Bloody Crusades. Penguin Group, New York (2010)
4. Beck, C.: Superstatistics: theory and applications. Contin. Mech. Thermodyn. **16**(3), 293–304 (2004)
5. Bernstein, S.: Sur les fonctions absolument montones. Acta Math. **51**(1), 1–66 (1928)
6. Clauset, A., Young, M., Gleditsch, K.S.: On the frequency of severe terrorist events. Confl. Res. **51**(1), 58–88 (2007)
7. Concas, G., Marchesi, M., Pinna, S., Serra, N.: Power-laws in a large object-oriented software system. IEEE Trans. Softw. Eng. **33**(10), 687–708 (2007)
8. Crovella, M.E., Bestavros, A.: Self-similarity in world wide web traffic-evidence and possible causes. IEEE/ACM Trans. Netw. **5**(6), 835–846 (1997)
9. Feldmann, A., Whitt, W.: Fitting mixtures of exponentials to long-tail distributions to analyze network performance models. Perform. Eval. **31**, 245–279 (1998)
10. Feng, J.: Is the integrate and fire model good enough? Neural Netw. **14**(6), 955–975 (2001)
11. Gabaix, X., Gopikrishnan, P., Plerou, V., Stanley, H.E.: A theory of power-law distributions in financial market fluctuations. Nature **423**(6937), 267–270 (2003)
12. Gell-Mann, M., Tsallis, C.: Nonextensive Entropy Interdisciplinary Applications. Oxford University Press, Oxford (2004)
13. Gupta, A.: Karmeshu: study of compound generalized Nakagami-generalized inverse Gaussian distribution and related densities: application in ultrasound imaging. Comput. Stat. **30**, 81–96 (2015)
14. Gutenberg, B., Richter, R.F.: Frequency of earthquakes in California. Bull. Seismolog. Soc. Am. **34**(4), 185–188 (1944)
15. Jaynes, E.T.: On the rationale of maximum entropy methods. Proc. IEEE **70**(9), 939–952 (1982)

16. Karmeshu: Motion of a particle in a velocity dependent random force. J. Appl. Prob. **13**, 684–695 (1976)
17. Karmeshu, Agrawal, R.: On efficacy of Rayleigh-inverse Gaussian distribution over K-distribution for wireless fading channels. Wireless Communications and Mobile Computing **7**(1), 1–7 (2006)
18. Karmeshu, Sharma, S.: Power law and Tsallis entropy: network traffic and applications. Chaos Nonlinearity Complex. Stud. Fuzziness Soft Comput. **206**, 162–178 (2006)
19. Karmeshu, Sharma, S.: Queue length distribution of network packet traffic: Tsallis entropy maximization with fractional moments. IEEE Commun. Lett. **10**(1), 34–36 (2006)
20. Karmeshu, Sharma, S.K.: Ensemble of LIF neurons with random membrance decay constant: emergence of power-law behavior in ISI distribution. IEEE Trans. Nanobiosci. **13**(3), 308–314 (2014)
21. Karmeshu (ed.): Entropy Measures, Maximum Entropy Principle and Emerging Applications. Springer, Berlin (2003)
22. Kouvatsos, D.D.: In: Potier, D. (ed.) Modelling Techniques and Tools for Performance Analysis. North-Holland, Amsterdam (1985)
23. Lathora, V., Rapisarda, A., Raffo, S.: Lyapunov instability and finite size effects in a system with long-range forces. Phys. Rev. Lett. **80**(4), 692–695 (1998)
24. Levy, M., Solomon, S.: New evidence for the power-law distribution of wealth. Phys. A: Stat. Mech. Appl. **242**(1), 90–94 (1997)
25. Makse, H., Havlin, S., Stanley, H.E.: Modeling urban growth patterns. Nature **377**(6550), 608–612 (1995)
26. Malamud, B.D., Turcotte, D.L.: The applicability of power-law frequency statistics to floods. J. Hydrol. **322**(1), 168–180 (2006)
27. Mandelbrot, B.: An informational theory of the statistical structure of languages. Commun. Theory **84**, 486–502 (1953)
28. Manin, D.Y.: Mandelbrot's model for Zipf's law: can Mandelbrot's model explain Zipf's law for language? J. Quant. Ling. **16**(3), 274–285 (2009)
29. Medina, A., Matta, I., Byers, J.: On the origin of power laws in internet topologies. ACM SIGCOMM Comput. Commun. Rev. **30**(2), 18–28 (2000)
30. Milotti, E.: $1/f$ noise: a pedagogical review. arxiv.org/pdf/physics/0204033 (2002)
31. Montroll, E.W., Shlesinger, M.F.: On $1/f$ noise and other distributions with long tails. Proc. Natl. Acad. Sci. **79**, 3380–3383 (1982)
32. Montroll, E.W., Shlesinger, M.F.: Maximum entropy formalism, fractals, scaling phenomena and $1/f$ noise: a tale of tails. J. Stat. Phys. **32**, 209–230 (1983)
33. Murre, M.J., Chessa, A.G.: Power laws from individual differences in learning and forgetting: mathematical analyses. Psychon. Bull. Rev. **18**(3), 592–597 (2011)
34. Newman, M.E.J.: Power laws, Pareto distributions and Zipf's law. Contemp. Phys. **46**(5), 323–351 (2005)
35. Norros, I.: A storage model with self-similar input. Queueing Syst. **16**(4), 387–396 (1994)
36. Oboukhov, A.M.: Some specific features of atmospheric tubulence. J. Fluid Mech. **13**(1), 77–81 (1962)
37. Peng, C.K., Buldyrev, S., Goldberger, A., Havlin, S., Sciortino, F., Simons, M., Stanley, H.E.: Long-range correlations in nucleotide sequences. Nature **356**(6365), 168–171 (1992)
38. Roberts, D.C., Turcotte, D.L.: Fractality and self orgaized crticality of wars. Fractals **6**(4), 351–357 (1998)
39. Salinas, E., Sejnowski, T.J.: Integrate-and-fire neurons driven by correlated stochastic input. Neural Comput. **14**(9), 2111–2155 (2002)
40. Shannon, C.E.: A mathematical theory of communication. Bell Syst. Tech. J. **27**, 379–423 (1984)
41. Sharma, S., Karmeshu: Power law characteristics and loss probability: finite buffer queueing systems. IEEE Commun. Lett. **13**(12), 971–973 (2009)
42. Sharma, S.K., Karmeshu: Power law in IF model with random excitatory and inhibitory rates. IEEE Trans. Nanobiosci. **10**(3), 172–176 (2011)

43. Shockley, W.: On the statistics of individual variations of productivity in research laboratories. Proc. IRE **45**(3), 279–290 (1957)
44. Singh, A.K., Karmeshu: Power law behavior of queue size: maximum entropy principle with shifted geometric mean constraint. IEEE Commun. Lett. **18**(8), 1335–1338 (2014)
45. Suki, B., Barabasi, A.L., Hantos, Z., Petak, F., Stanley, H.E.: Avalanches and power law behaviour in lung inflation. Nature **368**(6472), 615–618 (1994)
46. Tsallis, C.: Possible generalization of Boltzmann-Gibbs statistics. Stat. Phys. **52**(1–2), 479–487 (1988)
47. Tuckwell, H.C., Feng, J.: The Theoretical Overview in the Computaional Neuroscience. Chapman & Hall, London (2004)
48. Wilk, G., Wlodarczyk, Z.: Interpretation of the nonextensitivity parameter q in some applications of Tsallis statistics and Levy distributions. Phys. Rev. Lett. **84**(13), 2770–2774 (2000)
49. Willinger, W., Taqqu, M.S., Sherman, R., Wilson, D.V.: Self-similarity through high variability: statistical analysis of ethernet LAN traffic at source level. IEEE/ACM Trans. Netw. **5**(1), 71–86 (1997)
50. Wong, E.: The construction of a class of stationary Markoff processes. In: Proceedings of the Symposium on Applied Mathematics, pp. 264–276 (1964)
51. Zipf, G.K.: Selective Studies and the Principle of Relative Frequency in Language. Harvard University Press, Cambridge (1937)

Optimal Control of Multi-phase Movements with Learned Dynamics

Andreea Radulescu, Jun Nakanishi and Sethu Vijayakumar

Abstract In this paper, we extend our work on movement optimisation for variable stiffness actuation (VSA) with multiple phases and switching dynamics to incorporate scenarios with incomplete, complex or hard to model robot dynamics. By incorporating a locally weighted nonparametric learning method to model the discrepancies in the system dynamics, we formulate an online adaptation scheme capable of systematically improving the multi-phase plans (stiffness modulation and torques) and switching instances while improving the dynamics model on the fly. This is demonstrated on a realistic model of a VSA brachiating system with excellent adaptation results.

Keywords Optimal control · Variable stiffness actuators · Adaptive dynamics learning · Adaptive control

1 Introduction

The accuracy of model-based control is significantly dependent on that of the models themselves. Traditional robotics employs models obtained from mechanical engineering insights. Kinematic equations will provide accurate information about the evolution of a rigid body configuration, given a precise knowledge of its geometry. Similarly, the dynamics equation can incorporate well modelled factors such as inertia, Coriolis and centrifugal effect or external forces.

However, there are certain elements that cannot be fully captured by these models, such as friction from the joints or resulting from cable movement [22], which can vary in time. The introduction of flexible elements in the structure of a system increases

A. Radulescu · S. Vijayakumar (✉)
Institute of Perception, Action and Behaviour, University of Edinburgh, Edinburgh, UK
e-mail: sethu.vijayakumar@ed.ac.uk

J. Nakanishi
Institute for Cognitive Systems, Technical University of Munich, Munich, Germany

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_5

the complexity of the model and makes the identification of accurate dynamics significantly more difficult. Additionally, during operation, the robot can suffer changes in its mechanical structure due to wear and tear or due to the use of a tool (which modifies the mechanical chain structure) [23].

On-line adaptation of models can provide a solution for capturing all these properties. Early approaches, such as on-line *parameter identification* [21], which tunes the parameters of a predefined model (dictated by the mechanical structure) using data collected during operation, proved sufficient for accurate control and remained a popular approach for a long time [1, 7]. The increased complexity of latest robotic systems demands novel approaches capable of accommodating significant non-linear and unmodelled robot dynamics. Successful *non-parametric model learning* methods use supervised learning to perform system identification with only limited prior information about its structure—removing the restriction to a fixed model structure, allowing the model complexity to adapt in a data driven manner.

In this work, we will build on our significant prior efforts to engage this techniques in the context of robot control [11, 13, 17] and apply this in the context of multiphase variable impedance movements. Indeed, adaptive model learning has been used successfully in a wide range of scenarios such as inverse dynamics control [18, 26], inverse kinematics [5, 24], robot manipulation and locomotion [6, 20].

1.1 Adaptive Learning for Optimal Control

Classical OC (Optimal Control) is formulated using an analytic dynamics model, but recent work [2, 10] has shown that combining OC with dynamics learning can produce a powerful and principled control strategy for complex systems with redundant actuation.

In [10], using online (non-parametric) supervised learning methods, an adaptive internal model of the system dynamics is learned. The model is afterwards used to derive an optimal control law. This approach, named *iterative Linear Quadratic-Gaussian (iLQG) method with learned dynamics (iLQG-LD)*, proved efficient in a variety of realistic scenarios, including problems where the analytic dynamics model is difficult to estimate accurately or subject to changes and the system is affected by noise [10, 12].

In iLQG-LD the update of the dynamics model takes place on a trial-by-trial basis [12]. The operating principle (depicted in Fig. 1) is to (i) compute the iLQG solution, (ii) run the obtained control law on the plant and collect data, (iii) use the plant data to update the dynamics model.

The initial state and the cost function (which includes the desired final state) are provided to the iLQG planner, alongside a preliminary model of the dynamics. An initial (locally optimal) command sequence $\bar{\mathbf{u}}$ is generated, together with the corresponding state sequence $\bar{\mathbf{x}}$ and feedback correction gains $\mathbf{L}$. Applying the feedback controller scheme, at each time step the control command is corrected by

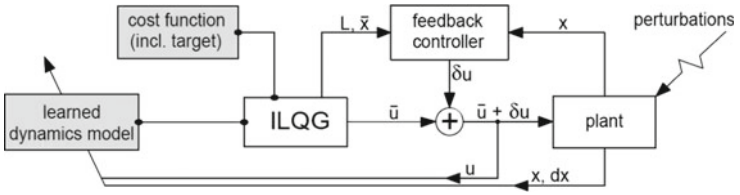


Fig. 1 The iLQG-LD learning and control scheme as first introduced in [12]

$\delta\mathbf{u} = \mathbf{L}(\mathbf{x} - \bar{\mathbf{x}})$, where $\mathbf{x}$ is the true state of the plant. The model of the dynamics is updated using the information provided by the applied command $\mathbf{u} + \delta\mathbf{u}$ and observed state $\mathbf{x}$.

This methodology employs the *Locally Weighted Projection Regression (LWPR)* [8] as the nonparametric learning scheme of choice to train a model of the dynamics in an incremental fashion. In LWPR, the regression function is constructed by combining local linear models. During training the parameters of the local models (locality and fit) are updated using incremental *partial least squares (PLS)*. PLS projects the input on to a small number of directions in the input space along the directions of maximal correlation with the output and then performs linear regression on the projected inputs. This makes LWPR suitable for high dimensional input spaces. Local models can be pruned or added on an as-need basis (e.g., when training data is generated in previously unexplored regions). The areas of validity (receptive fields) of each local model are modelled by Gaussian kernels. LWPR keeps a number of variables that hold sufficient statistics for the algorithm to perform the required calculations incrementally.

We incorporate the iLQG-LD scheme into our approach involving learning the dynamics of a brachiation system with VSA (variable stiffness actuator) capabilities and employing it in planning for locomotion tasks.

2 Problem Formulation

In our previous work [14], we introduced a general formulation of optimal control problems for tasks with multiple phase movements including switching dynamics and discrete state transition arising from iterations with an environment. Given a rigid body dynamics formulation of a robot with a VSA model, a hybrid dynamics representation with a composite cost function is introduced to describe such a task. In this section we briefly describe this approach, for details we refer the interested reader to [14]. We also introduce the changes dictated by the use of the LWPR method in the context of iLQG-LD, for integration within our approach.

2.1 Hybrid Dynamics with Time-Based Switching and Discrete State Transition

We employ the following hybrid dynamics representation to model multi-phase movements having interactions with an environment [4]:

$$\dot{\mathbf{x}} = \mathbf{f}_{i_j}(\mathbf{x}, \mathbf{u}), \quad T_j \leq t < T_{j+1} \tag{1}$$

$$\mathbf{x}(T_j^+) = \boldsymbol{\Delta}^{i_{j-1}, i_j}(\mathbf{x}(T_j^-)) \tag{2}$$

with $j = 0, \ldots, K$ for (1) and $j = 1, \ldots, K$ for (2) and where $\mathbf{f}_i : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ is the ith subsystem, $\mathbf{x} \in \mathbb{R}^n$ is a state vector, $\mathbf{u} \in \mathbb{R}^m$ is a control input vector.

When the dynamics switch from subsystem i_{j-1} to i_j at $t = T_j$, we assume that instantaneous discrete (discontinuous) state transition is introduced, which is denoted by a map $\boldsymbol{\Delta}^{i_{j-1}, i_j}$ in (2). The terms $\mathbf{x}(T_j^+)$ and $\mathbf{x}(T_j^-)$ denote the post- and pre-transition states, respectively. In this case, the sequence of switching is assumed to be given, e.g., $(1, 2, \ldots, K, K+1)$ or $(1, 2, 1, 2, \ldots)$. Figure 2 depicts a schematic diagram of a hybrid system we consider in this work.

2.2 Robot Dynamics with Variable Stiffness Actuation

To describe multi-phase movements, we consider multiple sets of robot dynamics, as described by (1). An individual rigid body dynamics model is defined for each associated phase of the movement as a subsystem. The servo motor dynamics in the VSA are modelled as a critically damped second order dynamics:

$$\mathbf{M}_i(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_i(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}_i(\mathbf{q}) + \mathbf{D}_i\dot{\mathbf{q}} = \boldsymbol{\tau}_i(\mathbf{q}, \mathbf{q}_m) \tag{3}$$

$$\ddot{\mathbf{q}}_m + 2\boldsymbol{\alpha}_i\dot{\mathbf{q}}_m + \boldsymbol{\alpha}_i^2\mathbf{q}_m = \boldsymbol{\alpha}_i^2\mathbf{u} \tag{4}$$

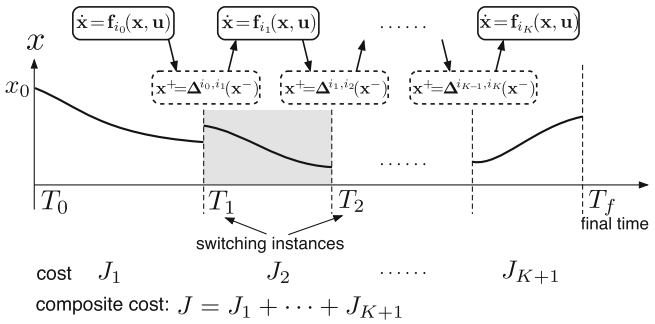


Fig. 2 A hybrid system with time-based switching dynamics and discrete state transition with a known sequence. The objective is to find an optimal control command $\mathbf{u}$, switching instances T_i and final time T_f which minimises the composite cost J

where i denotes the ith subsystem, $\mathbf{q} \in \mathbb{R}^n$ is the joint angle vector, $\mathbf{q}_m \in \mathbb{R}^m$ is the motor position vector of the VSA, $\mathbf{M} \in \mathbb{R}^{n\times n}$ is the inertia matrix, $\mathbf{C} \in \mathbb{R}^n$ is the Coriolis term, $\mathbf{g} \in \mathbb{R}^n$ is the gravity vector, $\mathbf{D} \in \mathbb{R}^{n\times n}$ is the viscous damping matrix, and $\boldsymbol{\tau} \in \mathbb{R}^n$ are the joint torques from the variable stiffness mechanism. In the equations above, (3) denotes the rigid body dynamics of the robot and (4) denotes the servo motor dynamics in the variable stiffness actuator. In (4), $\boldsymbol{\alpha}$ determines the bandwidth of the servo motors[1] and $\mathbf{u}$ is the motor position command [3]. We assume that the range of control command $\mathbf{u}$ is limited between $\mathbf{u}_{min}$ and $\mathbf{u}_{max}$.

In this work, we consider a VSA model in which the joint torques are given in the form

$$\boldsymbol{\tau}(\mathbf{q}, \mathbf{q}_m) = \mathbf{A}^T(\mathbf{q}, \mathbf{q}_m)\mathbf{F}(\mathbf{q}, \mathbf{q}_m) \tag{5}$$

where $\mathbf{A}$ is the moment arm matrix and $\mathbf{F}$ is the forces by the elastic elements [3] and the joint stiffness is defined as $\mathbf{K} = -\frac{\partial \boldsymbol{\tau}}{\partial \mathbf{q}}$.

We consider the state space representation as the combined plant dynamics consisting of the rigid body dynamics (3) and the servo motor dynamics (4):

$$\dot{\mathbf{x}} = \mathbf{f}_i(\mathbf{x}, \mathbf{u}) \tag{6}$$

where

$$\mathbf{f}_i = \begin{bmatrix} \mathbf{x}_2 \\ \mathbf{M}_i^{-1}(\mathbf{x}_1)\left(-\mathbf{C}_i(\mathbf{x}_1, \mathbf{x}_2)\mathbf{x}_2 - \mathbf{g}_i(\mathbf{x}_1) - \mathbf{D}_i\dot{\mathbf{x}}_2 + \boldsymbol{\tau}_i(\mathbf{x}_1, \mathbf{x}_3)\right) \\ \mathbf{x}_4 \\ -\boldsymbol{\alpha}_i^2\mathbf{x}_3 - 2\boldsymbol{\alpha}_i\mathbf{x}_4 + \boldsymbol{\alpha}_i^2\mathbf{u} \end{bmatrix} \tag{7}$$

and $\mathbf{x} = [\mathbf{x}_1^T,\ \mathbf{x}_2^T,\ \mathbf{x}_3^T,\ \mathbf{x}_4^T]^T = [\mathbf{q}^T,\ \dot{\mathbf{q}}^T,\ \mathbf{q}_m^T,\ \dot{\mathbf{q}}_m^T]^T \in \mathbb{R}^{2(n+n)}$ is the state vector consisting of the robot state and the servo motor state.

Employing the iLQG-LD framework we aim to create an accurate model of the dynamics model of the real hardware using supervised learning. We assume the existence of a preliminary analytic dynamics model which takes the form presented in (3), (4), which is inaccurate (due to various factors such as: the inability of the rigid body dynamics to incorporate all the elements of the system's behaviour or changes suffered during operation).

We use the LWPR method to model the error between the true behaviour of the system and the initial model provided. Thus we replace the dynamics $\mathbf{f}_i$ in (6) with the composite dynamics model $\mathbf{f}_{\mathbf{c}i}$:

$$\dot{\mathbf{x}} = \mathbf{f}_{\mathbf{c}i}(\mathbf{x}, \mathbf{u}) = \tilde{\mathbf{f}}_i(\mathbf{x}, \mathbf{u}) + \bar{\mathbf{f}}_i(\mathbf{x}, \mathbf{u}) \tag{8}$$

where $\tilde{\mathbf{f}}_i \in \mathbb{R}^{2(n+n)}$ is the initial inaccurate model and $\bar{\mathbf{f}} \in \mathbb{R}^{2(n+n)}$ is the LWPR model mapping the discrepancy between $\tilde{\mathbf{f}}_i \in \mathbb{R}^{2(n+n)}$ and the behaviour of the

[1] $\boldsymbol{\alpha} = \text{diag}\{a_1, \ldots, a_m\}$ and $\boldsymbol{\alpha}^2 = \text{diag}\{a_1^2, \ldots, a_m^2\}$ for notational convenience.

system. We note that the changes introduced by iLQG-LD only affect the dynamics modelling in (1), while the instantaneous state transition mapped by $\mathbf{\Delta}$ in (2) remains unchanged.

2.3 Movement Optimisation of Multiple Phases

For the given hybrid dynamics, in order to describe the full movement with multiple phases, we consider the following composite cost function:

$$J = \phi(\mathbf{x}(T_f)) + \sum_{j=1}^{K} \psi^j(\mathbf{x}(T_j^-)) + \int_{T_0}^{T_f} h(\mathbf{x}, \mathbf{u}) dt \tag{9}$$

where $\phi(\mathbf{x}(T_f))$ is the terminal cost, $\psi^j(\mathbf{x}(T_j^-))$ is the via-point cost at the jth switching instance and $h(\mathbf{x}, \mathbf{u})$ is the running cost.

2.4 Optimal Control of Switching Dynamics and Discrete State Transition

In brief, the iLQR method solves an optimal control problem of the locally linear quadratic approximation of the nonlinear dynamics and the cost function around a nominal trajectory $\bar{\mathbf{x}}$ and control sequence $\bar{\mathbf{u}}$ in discrete time, and iteratively improves the solutions.

In order to incorporate switching dynamics and discrete state transition with a given switching sequence, the hybrid dynamics (1) and (2) are linearised in discrete time around the nominal trajectory and control sequence as

$$\delta\mathbf{x}_{k+1} = \mathbf{A}_k \delta\mathbf{x}_k + \mathbf{B}_k \delta\mathbf{u}_k \tag{10}$$

$$\delta\mathbf{x}_{k_j}^+ = \mathbf{\Gamma}_{k_j} \delta\mathbf{x}_{k_j}^- \tag{11}$$

$$\mathbf{A}_k = \mathbf{I} + \Delta t_j \left.\frac{\partial \mathbf{f}_{i_j}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_k}, \quad \mathbf{B}_k = \Delta t_j \left.\frac{\partial \mathbf{f}_{i_j}}{\partial \mathbf{u}}\right|_{\mathbf{u}=\mathbf{u}_k} \tag{12}$$

$$\mathbf{\Gamma}_{k_j} = \left.\frac{\partial \mathbf{\Delta}^{i_{j-1}, i_j}}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_{k_j}^-} \tag{13}$$

where $\delta\mathbf{x}_k = \mathbf{x}_k - \bar{\mathbf{x}}_k$, $\delta\mathbf{u}_k = \mathbf{u}_k - \bar{\mathbf{u}}_k$, k is the discrete time step, Δt_j is the sampling time for the time interval $T_j \leq t < T_{j+1}$, and k_j is the jth switching instance in the discretised time step.

When using the composite model of the dynamics ($\mathbf{f_c}$) introduced in (8) the linearisation of the dynamics is provided in two parts. The linearisation of $\tilde{\mathbf{f}}$ is obtained

by replacing $\mathbf{f}$ with $\tilde{\mathbf{f}}$ in (10) and (12). The derivatives of the learned model ($\bar{\mathbf{f}}$) are obtained analytically by differentiating with respect to the inputs $\mathbf{z} = (\mathbf{x}; \mathbf{u})$ as suggested in [2]:

$$\frac{\partial \bar{\mathbf{f}}(\mathbf{z})}{\partial \mathbf{z}} = \frac{1}{W}\sum_k \left(\frac{\partial w_k}{\partial \mathbf{z}}\psi_k(\mathbf{z}) + w_k\frac{\partial \psi_k}{\partial \mathbf{z}}\right) - \frac{1}{W^2}\sum_k w_k(\mathbf{z})\psi_k(\mathbf{z})\sum_l \frac{\partial w_l}{\partial \mathbf{z}}$$

$$= \frac{1}{W}\sum_k (-\psi_k w_k \mathbf{D}_k(\mathbf{z}-\mathbf{c}_k) + w_k\mathbf{b}_k) + \frac{\bar{\mathbf{f}}(\mathbf{z})}{W}\sum_k w_k\mathbf{D}_k(\mathbf{z}-\mathbf{c}_k) \quad (14)$$

where

$$\frac{\partial \bar{\mathbf{f}}(\mathbf{z})}{\partial \mathbf{z}} = \begin{pmatrix} \partial\bar{\mathbf{f}}/\partial\mathbf{x} \\ \partial\bar{\mathbf{f}}/\partial\mathbf{u} \end{pmatrix}. \quad (15)$$

Since there are no changes on the encoding of the instantaneous state transition (2) the equations in (11) and (13) remain unchanged for the iLQG-LD framework.

The composite cost function (9) is locally approximated in a quadratic form as

$$\begin{aligned} \Delta J \; &= \delta\mathbf{x}_N^T\phi_\mathbf{x} + \frac{1}{2}\delta\mathbf{x}_N^T\phi_{\mathbf{xx}}\delta\mathbf{x}_N + \sum_{j=1}^{K}\left((\delta\mathbf{x}_{k_j}^-)^T\psi_\mathbf{x}^j + \frac{1}{2}(\delta\mathbf{x}_{k_j}^-)^T\psi_{\mathbf{xx}}^j\delta\mathbf{x}_{k_j^-}\right) \\ &+ \sum_{k=1}^{N}\left(\delta\mathbf{x}_k^T h_\mathbf{x} + \delta\mathbf{u}_k^T h_\mathbf{u} + \frac{1}{2}\delta\mathbf{x}_k^T h_{\mathbf{xx}}\delta\mathbf{x}_k + \frac{1}{2}\delta\mathbf{u}_k^T h_{\mathbf{uu}}\delta\mathbf{u}_k + \delta\mathbf{u}_k h_{\mathbf{ux}}\delta\mathbf{x}_k\right)\Delta t_j \end{aligned} \quad (16)$$

and a local quadratic approximation of the optimal cost-to-go function is

$$v_k(\delta\mathbf{x}_k) = \frac{1}{2}\delta\mathbf{x}_k^T\mathbf{S}_k\delta\mathbf{x}_k + \delta\mathbf{x}_k^T\mathbf{s}_k. \quad (17)$$

For notational convenience, note that in (16), $\phi_\mathbf{x}$ and $\phi_{\mathbf{xx}}$ denote $\phi_\mathbf{x} = \frac{\partial\phi}{\partial\mathbf{x}}$ and $\phi_{\mathbf{xx}} = \frac{\partial^2\phi}{\partial\mathbf{x}^2}$, respectively. Similar definitions apply to other partial derivatives.

The local control law $\delta\mathbf{u}_k$ of the form

$$\delta\mathbf{u}_k = \mathbf{l}_k + \mathbf{L}_k\delta\mathbf{x}_k \quad (18)$$

is obtained from the Bellman equation

$$v_k(\delta\mathbf{x}_k) = min_{\delta\mathbf{u}}\{h_k(\delta\mathbf{x}_k, \delta\mathbf{u}_k) + v_{k+1}(\delta\mathbf{x}_{k+1})\} \quad (19)$$

by substituting (10) and (17) into the Eq. (19), where h_k is the local approximation of the running cost in (16) (see [9] for details).

Once we have a locally optimal control command $\delta\mathbf{u}$, the nominal control sequence is updated as $\bar{\mathbf{u}} \leftarrow \bar{\mathbf{u}} + \delta\mathbf{u}$. Then, the new nominal trajectory $\bar{\mathbf{x}}$ is computed by running the obtained control $\bar{\mathbf{u}}$ and the above process is iterated until convergence.

In order to optimise the switching instances and the total movement duration, we introduce a scaling parameter and sampling time for each duration between switching as (cf. (12) and (16)):

$$\Delta t'_j = \frac{1}{\beta_j}\Delta t_j \quad \text{for} \quad T_j \leq t < T_{j+1}, \quad \text{where} \, j = 0, \ldots, K. \tag{20}$$

By optimising the vector of temporal scaling factors $\boldsymbol{\beta} = [\ \beta_0,\ \ldots,\ \beta_K\ ]^T$ via gradient descent [19] we obtain each switching instance T_{j+1} and the total movement duration T_f. This approach was applied previously [15, 16] to optimise the frequency of the periodic movement and the movement duration of swing locomotion in a brachiation task.

In the complete optimisation, computation of optimal feedback control law and temporal scaling parameter update are iteratively performed until convergence. A pseudocode of the complete algorithm is summarised in Algorithm 1.

3 Brachiation System Dynamics

We evaluate the effectiveness of the approach on a robot brachiation task which incorporates switching dynamics and multiple phases of the movement in a realistic VSA actuator model. We consider a two-link underactuated brachiating robot with a MACCEPA [25] variable stiffness actuator. The equation of motion of the system used takes the standard form of rigid body dynamics where only the second joint is actuated[2]:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{D}\dot{\mathbf{q}} = \begin{bmatrix} 0 \\ \tau(\mathbf{q}, \mathbf{q}_m) \end{bmatrix} \tag{21}$$

where $\mathbf{q} = [q_1,\ q_2]^T$ is the joint angle vector, $\mathbf{M}$ is the inertia matrix, $\mathbf{C}$ is the Coriolis term, $\mathbf{g}$ is the gravity vector, $\mathbf{D}$ is the viscous damping matrix, τ is the joint torque acting on the second joint given by the VSA, and $\mathbf{q}_m$ is the motor positions vector in the VSA as described below.

The MACCEPA actuator is equipped with two position controlled servo motors, $\mathbf{q}_m = [q_{m1},\ q_{m2}]^T$, which control the equilibrium position and the spring

[2]For notational simplicity, the subscript i is omitted.

Algorithm 1 Complete optimisation algorithm for hybrid dynamics with temporal optimisation

1: **Input:**
   – Timed switching plant dynamics $\mathbf{f}_i$ (1 or 8), discrete state transition $\mathbf{\Delta}^{i_{j-1},i_j}$ (2) and switching sequence
   – Composite cost function J (9)
2: **Initialise:**
   – Nominal switching instance and final time $T_1, \cdots, T_K$ and T_f
   – Nominal control sequence $\bar{\mathbf{u}}$ and corresponding $\bar{\mathbf{x}}$
3: **repeat**
4: **repeat**
5: **Optimise control sequence $\bar{\mathbf{u}}$:**
   – Obtain linearised time-based switching dynamics (10 or 14) and state transition (11) around $\bar{\mathbf{x}}$ and $\bar{\mathbf{u}}$ in discrete time with current Δt_j
   – Compute quadratic approximation of the composite cost (16)
   – Solve local optimal control problem to obtain $\delta\mathbf{u}$ (18)
   – Apply $\delta\mathbf{u}$ to the linearised hybrid dynamics (10) and (16)
   – Update nominal control sequence $\bar{\mathbf{u}} \leftarrow \bar{\mathbf{u}} + \delta\mathbf{u}$, trajectory $\bar{\mathbf{x}}$ and cost J
6: **until** convergence
7: **Temporal optimisation: update Δt_j:**
   – Update the vector of temporal scaling factor $\boldsymbol{\beta}$ and corresponding sampling time $\Delta t_0, \cdots, \Delta t_K$ in (20) via gradient descent [19].
8: **until** convergence
9: **Output:**
   – Optimal feedback control law $\mathbf{u}(\mathbf{x}, t)$: forward optimal control sequence $\mathbf{u}_{opt}$, optimal trajectory $\mathbf{x}_{opt}(t)$ and optimal gain matrix $\mathbf{L}_{opt}(t)$:
   $\mathbf{u}(\mathbf{x}, t) = \mathbf{u}_{opt}(t) + \mathbf{L}_{opt}(t)(\mathbf{x}(t) - \mathbf{x}_{opt}(t))$
   – Optimal switching instance $T_1, \cdots, T_K$ and final time T_f
   – Optimal composite cost J

pre-tension,[3] respectively. The servo motor dynamics are approximated by a second order system with a PD feedback control, as mentioned in (4):

$$\ddot{\mathbf{q}}_m + 2\boldsymbol{\alpha}\dot{\mathbf{q}}_m + \boldsymbol{\alpha}^2\mathbf{q}_m = \boldsymbol{\alpha}^2\mathbf{u} \tag{22}$$

where $\mathbf{u} = [u_1,\ u_2]^T$ is the motor position command, $\boldsymbol{\alpha}$ determines the bandwidth of the actuator. In this study, we use $\boldsymbol{\alpha} = \text{diag}(20, 25)$. The range of the commands of the servo motors are limited as $u_1 \in [-\pi/2, \pi/2]$ and $u_2 \in [0, \pi/2]$.

We use the model parameters shown in Table 1 and the MACCEPA parameters with the spring constant $\kappa = 771$ N/m, the lever length $B = 0.03$ m, the pin displacement $C = 0.125$ m and the drum radius $r_d = 0.01$ m (Fig. 3).

[3] Which is used to modulate the stiffness of the joint, for details see [25].

Table 1 Model parameters of the two-link brachiating robot

Robot parameters		i = 1	i = 2	$i = 1$
Mass	m_i (kg)	1.390	0.527	*1.240*
Moment of inertia	I_i (kg m^2)	0.0297	0.0104	*0.0278*
Link length	l_i (m)	0.46	0.46	*0.46*
COM location	l_{ci} (m)	0.362	0.233	*0.350*
Viscous friction	d_i (Nm/s)	0.03	0.035	*0.03*

The final column shows the change of parameters of the first link of the system under the changed mass distribution described in Sect. 4

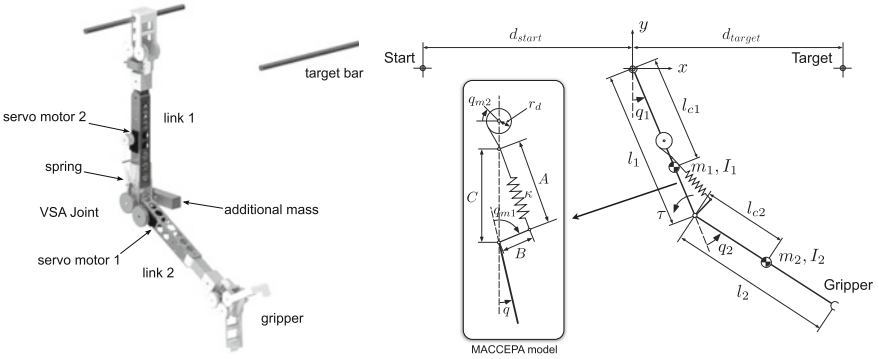


Fig. 3 Two-link brachiating robot model with the VSA joint with the inertial and geometric parameters. The parameters of the robot are given in Table 1, where the indices i denote the link number in this figure and Table 1

4 Experimental Setup

To test the efficiency of our approach we create a scenario where the difference between the true and the assumed model is caused by a change in the mass (and implicitly mass distribution) on one the links (i.e. the mass of the true model is smaller by 150 g (located at the joint) on link $i = 1$). The changed model parameters are shown in the right column of Table 1.[4]

Due to the nature of the discrepancy introduced, the error in the dynamics manifests itself only in the joint accelerations. Thus, we require to map the error just in those two dimensions, reducing the dimension of the output of the LWPR model $\bar{\mathbf{f}}$ from $n = 8$ to 2, where the predictions are added on the corresponding dimension of $\tilde{\mathbf{f}}$ within $\mathbf{f_c}$ (8). Note that different discrepancies will necessitate estimation of the full 8-dim state error.

In line with previous work, we will demonstrate the effectiveness of the proposed approach on a multi-phase, asymmetric swing-up and brachiation task with a VSA

[4]The MACCEPA parameters are the same as described in the previous section.

while incorporating *continuous, online model learning*. Specifically, in the multi-phase task, the robot swings up from the suspended posture to the target at $d_1 = 0.40$ m and subsequently moves to the target located at $d_2 = 0.42$ m and $d_3 = 0.46$ m, respectively.

Since the system has an asymmetric configuration and the state space of the swing up task is significantly different from that of a brachiation movement we proceed by first learning a separate error model for each phase. The procedure used is briefly described in Algorithm 2. The initial exploration loop is performed in order to pre-train the LWPR model $\bar{\mathbf{f}}_i$ (as an alternative to applying motor babbling), the later loop is using iLQG-LD to refine the model in an online fashion. In our experiments the training data is obtained by using a simulated version of the true dynamics, which is an analytic model incorporating the discrepancy.

Algorithm 2 Description of the learning and exploration procedure

Given:
- analytic dynamics for one configuration $\tilde{\mathbf{f}}_i$ and start state $\mathbf{x}_0$
- thresholds for target reaching ε_T and model accuracy ε_M
- the associated cost function J (including desired target $\mathbf{x}_T$)
- p number of initial exploration training episodes

Initialise
- $\bar{\mathbf{f}}_i(\mathbf{x}, \mathbf{u})$; $\mathbf{f}_{\mathbf{c}i}(\mathbf{x}, \mathbf{u}) = \tilde{\mathbf{f}}_i(\mathbf{x}, \mathbf{u}) + \bar{\mathbf{f}}_i(\mathbf{x}, \mathbf{u})$

repeat
  generate $\bar{\mathbf{u}}, \bar{\mathbf{x}}, \mathbf{L}$ using $\tilde{\mathbf{f}}_i(\mathbf{x}, \mathbf{u})$ for an artificial target (a new target at each iteration obtained by sampling around $\mathbf{x}_T$)
  apply the solution to the true dynamics and train the model on the collected data
until p **training episodes have been performed**

repeat
  apply iLQG-LD for target $\mathbf{x}_T$
until ε_T **and** ε_M **conditions are met**

4.1 Individual Phase Learning

Using the traditional OC framework in the presence of an accurate dynamics model, the multi-phase task described previously was achieved with a position error of just 0.002 m. Once the discrepancy detailed in Sect. 4 is introduced, the planned solution is no longer valid and the final position deviates from the desired target (Fig. 5, blue line). We deploy the iLQG-LD framework in order to learn the new behaviour of the system and recover the task performance.

As a measure of the model accuracy we use the *normalised mean square error* (nMSE) of the model prediction on the true optimal trajectory (if given access to the analytic form of the true dynamics of the system). The nMSE is defined as $nMSE(y, \tilde{y}) = \frac{1}{n\sigma_y^2} \sum_{i=1}^{n} (y_i - \tilde{y}_i)^2$ where y is the desired output data set of size n

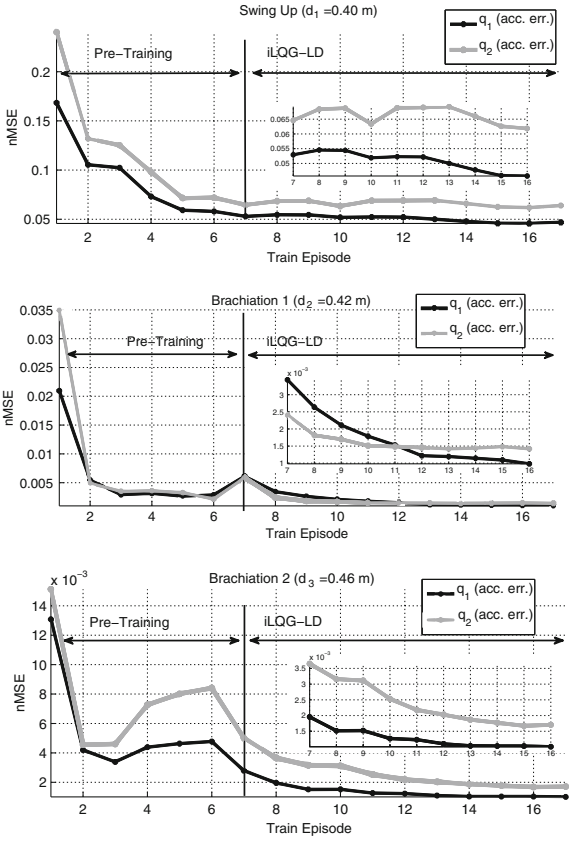


Fig. 4 Evolution of the nMSE for each phase of the movement, at each episode

and $\tilde{y}$ represents the LWPR predictions. The evolution of the nMSE at each stage of the training for every phase is shown in Fig. 4.

In the first part (pre-training phase in Fig. 4) we generate random targets around the desired x_T. A movement is planned for these targets using the assumed model ($\tilde{\mathbf{f}}$). The obtained command solution is then applied to the simulated version of the true dynamics, using a closed loop control scheme. We repeat the procedure for a set of 10 noise contaminated versions of the commands. The collected data is used to train the model.

This pre-training phase seeds the model with information within the region of interest, prior to using it for planning. This reduces the load on the iLQG-LD by lowering the number of iterations required for convergence. For each phase of the movement, at the end of the procedure, the planned trajectory matched the behaviour obtained from running the command solution on the real plant (the final *nMSE* has an order of magnitude of 10^{-4}).

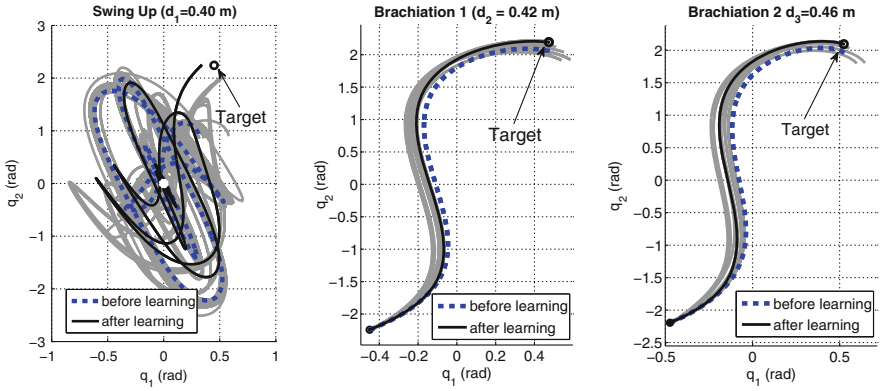


Fig. 5 Phase plot: comparison of the final position achieved (for each individual phase) when using the initial planning (erroneous model—*blue*) and the final planning (composite model—*black*). Intermediary solutions obtained at each step of the iLQG-LD run are depicted in *grey*

Overall the discrepancy is small enough to allow reaching the desired end effector position within a threshold of $\varepsilon_T = 0.040$ m accuracy.[5] Figure 5 shows the effect of the learning by comparing the performance of the planning with the erroneous model and with the composite model obtained after training.

4.2 Multi-phase Performance

In the previous section we showed that our approach to iLQG-LD is able to cope with the requirements of the task in each phase of the movement. For a full solution we use the newly learned models from each phase to obtain the global solution for the multi-phase task wrt. the composite cost function J (9). We use the phase optimal solutions obtained at the previous stage as the initial command sequence, the resulting behaviour is displayed in Fig. 6. The planner is able to use the learned model to achieve the intermediary and final goals, while the expected behaviour provides a reliable match to the actual system's behaviour.[6] The cost of multi-phase optimised solution ($J = 39.17$) is significantly lower than the sum of the costs of the individual phase solutions ($J = 58.23$).

[5]The error in the swing up task is 0.033 m, while for brachiations the value is 0.004 m. In future work we aim to bring the former value to the same magnitude.

[6]We consider that if the position at the end of each phase is within our prescribed threshold $\varepsilon_T = 0.040$ of the desired target the system is able to start the next phase from the ideal location, thus resembling the effect of the grabber on the hardware.

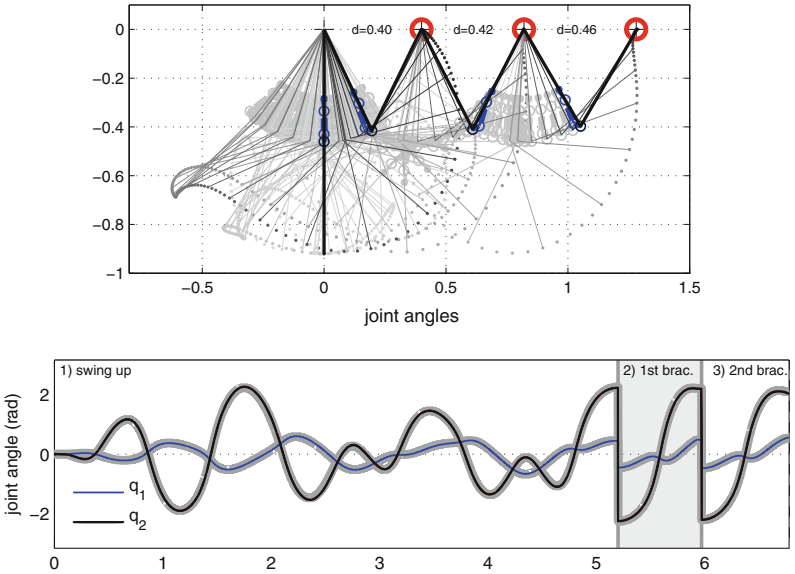


Fig. 6 Performance of the fully optimised multi-phase task using the composite model. *Thick grey lines* planned movement. *Black* and *blue lines* actual system movement

4.3 Performance of Learning

In the previous experiment, we investigated a single (arbitrarily chosen) mass distribution discrepancy. Next we investigate the capacity of our approach to cope with a wider range of mismatched dynamics. For this, we consider the magnitude of the change that is bounded by the capability of the altered system to achieve all the phases of the movement presented in Fig. 6. We define these bounds as the limit values of mass change that allow the same accuracy in task execution, under the same cost function (9). The corresponding values for these limits, found empirically, are -0.200 kg and $+0.300$ kg, respectively.[7]

We apply our framework to the scenarios where the mass has been altered to these boundary values and demonstrate the result on just one of the phases, namely the first brachiation move, to study the relative effects.

Figure 7a shows the effect of the alteration introduced, when executing the commands resulting from the initial planning (using the erroneous model). After training and re-planning the model starts approximating the true behaviour of the system, such that in less than 10 episodes, the system is once again able to reach the desired target, as depicted in Fig. 7b, c.

[7]We note that in our experiments, we assume that modulating the mass distribution does not affect the motor dynamics—this represents a simplified scenario. In the real hardware, the speed of the motor dynamics (4) is indeed a function of the overall load distribution.

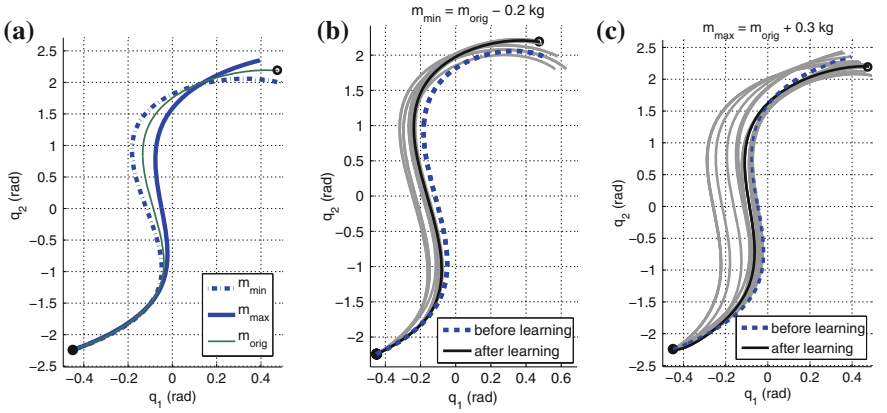


Fig. 7 Phase plot: **a** Effect of the discrepancy introduced by the limit values on the mass modulation (*blue lines*). The behaviour when the model match is correct is depicted for comparison (*green line*). **b–c** Comparison of the final position achieved (for each individual phase) when using the initial planning (erroneous model—*blue*) and the final planning (composite model—*black*). Intermediary solutions obtained at each step of the iLQG-LD run are depicted in *grey*. Results for the boundary value discrepancies on the mass distribution

5 Conclusion

In this work we have presented an extension of our methodology for movement optimisation with multiple phases and switching dynamics, including variable impedance actuators. We broaden the approach by incorporating adaptive learning, which allows for adjustments to the dynamics model, based on changes occurred to the system's behaviour, or when the behaviour cannot be fully capture by a rigid body dynamics formulation. In future work we aim to investigate a wider range of model discrepancies and show the performance of the extended approach on a hardware implementation.

References

1. Atkeson, C.G., An, C.H., Hollerbach, J.M.: Estimation of inertial parameters of manipulator loads and links. Int. J. Robot. Res. **5**(3), 101–119 (1986)
2. Atkeson, C., Moore, A., Schaal, S.: Locally weighted learning for control. Artif. Intell. Rev. **11**(1–5), 75–113 (1997)
3. Braun, D., Howard, M., Vijayakumar, S.: Optimal variable stiffness control: formulation and application to explosive movement tasks. Auton. Robots **33**(3), 237–253 (2012)
4. Caldwell, T.M., Murphey, T.D.: Switching mode generation and optimal estimation with application to skid-steering. Automatica **47**(1), 50–64 (2011)
5. Herbort, O., Butz, M., Pedersen, G.: The sure_reach model for motor learning and control of a redundant ARM: from modeling human behavior to applications in robotics. In: Sigaud, O.,

Peters, J. (eds.) From Motor Learning to Interaction Learning in Robots, Studies in Computational Intelligence. vol. 264, pp. 85–106. Springer, Heidelberg (2010)
6. Kalakrishnan, M., Buchli, J., Pastor, P., Schaal, S.: Learning locomotion over rough terrain using terrain templates. In: IROS 2009, pp. 167–172. St. Louis, USA (2009)
7. Khalil, W., Dombre, É.: Modeling, Identification and Control of Robots. Kogan Page Science, London (2004)
8. Klanke, S., Vijayakumar, S., Schaal, S.: A library for locally weighted projection regression. J. Mach. Learn. Res. **9**, 623–626 (2008)
9. Li, W., Todorov, E.: Iterative linearization methods for approximately optimal control and estimation of non-linear stochastic system. Int. J. Control **80**(9), 1439–1453 (2007)
10. Mitrovic, D., Klanke, S., Vijayakumar, S.: Adaptive optimal feedback control with learned internal dynamics models. In: Sigaud, O., Peters, J. (eds.) From Motor Learning to Interaction Learning in Robots, Studies in Computational Intelligence, vol 264, pp. 65–84 Springer, Berlin, Heidelberg (2010)
11. Mitrovic, D., Klanke, S., Howard, M., Vijayakumar, S.: Exploiting sensorimotor stochasticity for learning control of variable impedance actuators. In: Humanoids, pp. 536–541. Nashville, USA (2010)
12. Mitrovic, D., Klanke, S., Vijayakumar, S.: Optimal control with adaptive internal dynamics models. In: ICINCO 2008, Madeira, Portugal (2008)
13. Nakanishi, J., Farrell, J.A., Schaal, S.: Composite adaptive control with locally weighted statistical learning. Neural Netw. **18**(1), 71–90 (2005)
14. Nakanishi, J., Radulescu, A., Vijayakumar, S.: Spatio-temporal optimization of multi-phase movements: Dealing with contacts and switching dynamics. In: IROS 2013, pp. 5100–5107. Tokyo, Japan (2013)
15. Nakanishi, J., Rawlik, K., Vijayakumar, S.: Stiffness and temporal optimization in periodic movements: an optimal control approach. In: IROS, pp. 718–724. San Francisco, USA (2011)
16. Nakanishi, J., Vijayakumar, S.: Exploiting passive dynamics with variable stiffness actuation in robot brachiation. In: Robotics: Science and Systems, Sydney, Australia, July 2012
17. Nguyen-Tuong, D., Peters, J.: Model learning for robot control: a survey. Cogn. Process. **12**(4), 319–340 (2011)
18. Nguyen-Tuong, D., Seeger, M., Peters, J.: Model learning with local gaussian process regression. Adv. Robot. **23**(15), 2015–2034 (2009)
19. Rawlik, K., Toussaint, M., Vijayakumar, S.: An approximate inference approach to temporal optimization in optimal control. In: NIPS 2010, pp. 2011–2019. Vancouver, Canada (2010)
20. Schaal, S., Atkeson, C.G., Vijayakumar, S.: Scalable techniques from nonparametric statistics for real time robot learning. Appl. Intell. **17**(1), 49–60 (2002)
21. Siciliano, B., Sciavicco, L., Villani, L., Oriolo, G.: Robotics: Modelling, Planning and Control. In: Springer Science & Business Media. Berlin, Germany (2009)
22. Siciliano, B., Khatib, O.: Springer Handbook of Robotics. Springer, Berlin (2008)
23. Sigaud, O., Salaün, C., Padois, V.: On-line regression algorithms for learning mechanical models of robots: a survey. Robot. Auton. Syst. **59**(12), 1115–1129 (2011)
24. Ting, J.A., Kalakrishnan, M., Vijayakumar, S., Schaal, S.: Bayesian kernel shaping for learning control. In: NIPS 2009, pp. 1673–1680. Vancouver, Canada (2009)
25. Van Ham, R., Vanderborght, B., Van Damme, M., Verrelst, B., Lefeber, D.: MACCEPA, the mechanically adjustable compliance and controllable equilibrium position actuator: Design and implementation in a biped robot. Robot. Auton. Syst. **55**(10), 761–768 (2007)
26. Vijayakumar, S., D'souza, A., Shibata, T., Conradt, J., Schaal, S.: Statistical learning for humanoid robots. Auton. Robots **12**(1), 55–69 (2002)

Computable Extensions of Advanced Fractional Kinetic Equation and a Class of Levy-Type Probabilities

Manoj Sharma

Abstract In recent year's fractional kinetic equation are studied due to their usefulness and importance in mathematical physics, especially in astrophysical problems. In Astrophysics kinetic equations designate a system of differential equations, describing the rate of change of chemical composition of a star for each species in terms of the reaction rates for destruction and production of that species. Methods for modeling processes of destruction and production of stars have been developed for bio-chemical reactions and their unstable equilibrium states and for chemical reaction networks with unstable states, oscillations and hysteresis.The aim of present paper is to find the solution of generalized fractional order kinetic equation, using a new special function. The results obtained here is moderately universal in nature. Special cases, relating to the Mittag-Leffler function is also considered.

Keywords Fractional kinetic equation · Generalized M function · Riemann-Liouville operator · Laplace transform · Modified riemann-liouville fractional derivative operator · Differential equation · Probability density function

Mathematics Subject Classification: 33C60 · 33E12 · 82C31 · 26A33

1 Section I

1.1 Introduction

The talk divided into two sections: First we obtain computable extensions of advanced fractional kinetic equation and then in second section applied Modified Riemann-Liouville fractional Integral i.e. ms-operator and differential equation for a Class of Levy-type probabilities. The aim of this talk is to explore the behavior of physical and biological systems from the point of view of fractional calculus. Fractional

M. Sharma (✉)
Department of Mathematics, RJIT, BSF Academy, Tekanpur, Gwalior, Madhya Pradesh, India
e-mail: manoj240674@yahoo.co.in

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_6

calculus, integration and differentiation of an arbitrary or fractional order, provides new tools that expand the descriptive power of calculus beyond the familiar integer-order concepts of rates of change and area under a curve. Fractional calculus adds new functional relationships and new functions to the familiar family of exponentials and sinusoids that arise in the area of ordinary linear differential equations. Among such functions that play an important role, we have the Euler Gamma function, the Euler Beta function, the Mittag-Leffler functions, the Wright and Fox functions, M-Function, K-Function and so forth.The Fractional Calculus applied in The distributions of a extensive variety of physical, biological, and man-made phenomena approximately follow a power law over a wide range of magnitudes. More than a hundred power-law distributions have been identified in biology (e.g. species extinction and body mass), in physics (e.g. sand pile avalanches and earthquakes), and the social sciences (e.g. city sizes and income).When the probability of measuring a particular value of some quantity varies inversely as a power of that value, the quantity is said to follow a power law. A power law distribution is a special kind of probability distribution. Fractional moments are very useful in dealing with random variable with power law distributions, F(x) $\sim|x|-\alpha$, $\alpha>0$ where F(x) is the distribution function. In such cases, moments E(|x|p) exist only if p$<\alpha$ and integer order moments greater then α diverge. This type of problem arises in the distributions where power law statistics appear in many fields of applied science.

We give the new special function, called Generalized M function, which is the most generalization of M function [21]. Here, we give first the notation and the definition of the New Special Generalized M function, introduced by the author as follows:

$$
\begin{aligned}
&{}^{\alpha,\beta,\gamma,\delta,\rho}{}_{p}M_{q}^{k_1,\ldots k_p,l_1,\ldots l_q;c}(t) \\
&= \sum_{n=o}^{\infty} \frac{(a_1)_n\cdots(a_p)_n(\gamma)_n}{(b_1)_n\cdots(b_q)_n}\frac{(\delta)_n}{(\rho)_n}\frac{k_1^n\cdots k_p^n}{l_1^n\cdots l_q^n}\frac{(t-c)^{(n+\gamma)\alpha-\beta-1}}{\prod_{i=1}^{p}(n_i)!\Gamma((n+\gamma)\alpha-\beta)}
\end{aligned}
\tag{1}
$$

There are p upper parameters $a_1, a_2, \ldots a_p$ and q lower parameters $b_1, b_2, \ldots b_q$, $\alpha, \beta, \gamma, \delta, \rho \epsilon C$, $Re(\alpha) > 0$, $Re(\beta) > 0$, $Re(\gamma) > 0$, $Re(\delta) > 0$, $Re(\rho) > 0$, $Re(\alpha\gamma - \beta) > 0$ and $(a_j)_k (b_j)_k$ are pochhammer symbols and $k_1, \ldots k_p, l_1, \ldots l_q$ are constants. The function (1) is defined when none of the denominator parameters $b_j s$, $j = 1, 2, \ldots q$ is a negative integer or zero. If any parameter a_j is negative then the function (1) terminates into a polynomial in $(t - c)$.

1.2 Relationship of the ${}^{\alpha,\beta,\gamma,\delta,\rho}{}_{p}M_{q}^{k_1,\ldots k_p,l_1,\ldots l_q;c}(t)$ Function and Other Special Functions

In this section, we defined relationship of Generalized M function and various special functions.

(i). For $\prod_{i=1}^{p}(n_i)! = 1$, Then Eq. (1) converts in M function [21]

$$\begin{aligned}&{}^{\alpha,\beta,\gamma,\delta,\rho}{}_pM_q^{k_1,\ldots k_p,l_1,\ldots l_q;c}(t)\\&=\sum_{n=o}^{\infty}\frac{(a_1)_n\cdots(a_p)_n(\gamma)_n}{(b_1)_n\cdots(b_q)_n}\frac{(\delta)_n}{(\rho)_n}\frac{k_1^n\cdots k_p^n}{l_1^n\cdots l_q^n}\frac{(t-c)^{(n+\gamma)\alpha-\beta-1}}{n!\Gamma((n+\gamma)\alpha-\beta)}\end{aligned}\tag{2}$$

(ii). For $k_1 = a, k_2 \ldots k_p = 1, l_1, \ldots l_q = 1, \delta = 1 \wedge \rho = 1$, $\prod_{i=1}^{p}(n_i)!$ $\prod_{j=1}^{q}(n_j)! = n!$ K_4—function is given by Sharma [20] (2012),

$${}^{\alpha,\beta,\gamma,1,1}{}_pM_q^{a,1;c}(t)=\sum_{n=o}^{\infty}\frac{(a_1)_n\cdots(a_p)_n(\gamma)_n}{(b_1)_n\cdots(b_q)_n}\frac{a^n}{n!}\frac{(t-c)^{(n+\gamma)\alpha-\beta-1}}{\Gamma((n+\gamma)\alpha-\beta)}\tag{3}$$

(iii). If we take no upper and lower parameter ($p = q = 0$) in Eq. (3) then the function reduces to the G-Function, which was introduced by Lorenzo and Hartley [6] (1999).

$${}^{\alpha,\beta,\gamma,1,1}{}_1M_1^{a,1;c}(t)=\sum_{n=o}^{\infty}\frac{(\gamma)_n(a)^n(t-c)^{(n+\gamma)\alpha-\beta-1}}{n!\Gamma((n+\gamma)\alpha-\beta)}=G_{\alpha,\beta,\gamma}(a,c,t)\tag{4}$$

(iv). Taking $\gamma = 1$, in Eq. (4), we get the R—function given by introduced by Lorenzo and Hartley [6] (1999).

$$\begin{aligned}&{}^{\alpha,\beta,1,1,1}{}_1M_1^{a,1;c}(t)=\sum_{n=o}^{\infty}\frac{(\gamma)_n(a)^n(t-c)^{(n+\gamma)\alpha-\beta-1}}{n!\Gamma((n+\gamma)\alpha-\beta)}\\&=R_{\alpha,\beta}[a,t]\alpha>0,\beta>0,(\alpha-\beta)>0\end{aligned}\tag{5}$$

Now, we take $c = 0$, in various standard function.

(v). For $c = 0$, in Eq. (4), the Generalized M function reduces to New Generalized Mittag-Leffler Function [17]

$${}^{\alpha,\beta,\gamma,1,1}{}_1M_1^{a,1}(t)=t^{\alpha\gamma-\beta-1}\sum_{n=o}^{\infty}\frac{(\gamma)_n(a)^n(t)^{an}}{n!\Gamma((n+\gamma)\alpha-\beta)}=t^{\alpha\gamma-\beta-1}E^{\gamma}_{\alpha,\alpha\gamma-\beta}[at^{\alpha}]\tag{6}$$

(vi). We take $\gamma = 1$, in (6) obtained Generalized Mittag-Leffler Function [17], we get

$${}^{\alpha,\beta,1,1,1}{}_1M_1^{a,1;c}(t)=\sum_{n=o}^{\infty}\frac{(a)_n(t)^{(n+1)\alpha-\beta-1}}{\Gamma((n+1)\alpha-\beta)}=t^{\alpha-\beta-1}E^{\gamma}_{\alpha,\alpha-\beta}[at^{\alpha}]\tag{7}$$

(vii). Further $\beta = \alpha - 1$ in (6), this Generalized M function converts in to Mittag-Leffler Function [8, 9], we have

$${}^{\alpha,\alpha-1,1,1,1}{}_1M_1^{a,1}(t)=\sum_{n=o}^{\infty}\frac{(a)^n(t)^{na}}{\Gamma(na+1)}=E_{\alpha,\beta}[t^{\alpha}]\tag{8}$$

(viii). When $a = 1, c = 0 \wedge \beta = \alpha - \beta$ in (4) then the Generalized M function treats as Agarwal's Function [1]

$$ {}_{\alpha,\alpha-\beta,1,1,1}{}_1M_1^{1,1}(t) = \sum_{n=o}^{\infty} \frac{(t)^{na+\beta-1}}{\Gamma(na+\beta)} = E_{\alpha,\beta}[t^{\alpha}] \tag{9} $$

(ix). Robotnov and Hartley Function [6] is obtained from M function by putting $\beta = 0, a = -a, c = 0$ in (9), we have

$$ {}_{\alpha,0,1,1,1}{}_1M_1^{-a,1}(t) = \sum_{n=o}^{\infty} \frac{(-a)^n (t)^{(n+1)\alpha-1}}{\Gamma((n+1)\alpha)} = F_{\alpha}[-a,t] \tag{10} $$

(x). On substituting $\alpha = 1, \beta = -\beta$ in (5), we get Miller and Ross Function [7].

$$ {}_{1,-\beta,1,1,1}{}_1M_1^{a,1}(t) = \sum_{n=o}^{\infty} \frac{(a)^n (t)^{n+\beta}}{\Gamma(n+\beta+1)} = E_t[\beta,\alpha] \tag{11} $$

(xi). Let us consider $c = 0$ in Eq. (4), this function converts into Wright Function [16]. We have,

$$ {}_{\alpha,\beta,\gamma,1,1}{}_1M_1^{a,1}(t) = \frac{t^{\alpha\gamma-\beta-1}}{\Gamma\gamma} {}_1^0\Psi_1\left[\begin{matrix}(\gamma,1)\\(\alpha,\gamma-\beta),\alpha;at^{\alpha}\end{matrix}\right] \tag{12} $$

where ${}_1^0\Psi_1(t)$ is special case of the wright's generalized Hypergeometric function ${}_p^0\Psi_q(t)$.
Or

(xii). Thus we get H-Function [16] from last case.

$$ {}_{\alpha,\beta,\gamma,1,1}{}_1M_1^{a,1}(t) = \frac{t^{\alpha\delta-\beta-1}}{\Gamma\gamma} H_{1,2}^{1,1}\left[-at^{\alpha}\left|\begin{matrix}(1-\gamma,1)\\(0,1)\,(1-\alpha\gamma+\beta),\alpha\end{matrix}\right.\right] \tag{13} $$

The Laplace transform of (1), from Lorenzo and Hartley [6] (1999) with shifting theorem (Wylie, p.281) we have

$$ \begin{aligned} &L\left\{{}_{\alpha,\beta,\gamma,\delta,\rho}{}_pM_q^{k_1,\ldots k_p,l_1,\ldots l_q;c}(t)\right\} \\ &= \frac{(a_1)_n \ldots (a_p)_n}{(b_1)_n \ldots (b_q)_n \prod_{i=1}^{p}(n_i)!} \frac{1}{l_1^n \ldots l_q^n} \frac{s^{\beta}e^{-cs}}{\left\{s^{\alpha}+\left(k_1^n \ldots k_p^n\right)\right\}^{\gamma}} \end{aligned} \tag{14} $$

1.3 Governing Fractional Kinetic Equation

Let us define an arbitrary reaction which is dependent on time $N = N(t)$. It is possible to calculate rate of change dN/dt to a balance between the destruction rate d and the production rate p of N, then

$$\frac{dN}{dt} = -d + p.$$

The production or destruction at time t depends not only on N(t) but also on the previous history $N(t_1), t_1 < t$, of the variable N.

This was represented by Haubold and Mathai [5] as follows:

$$dN/dt = -d(Nt) + p(Nt) \tag{15}$$

where $N(t)$ denotes the function defined by

$$Nt(t_1) = N(t - t_1), t_1 > 0. \tag{16}$$

Haubold and Mathai [5] considered a special case of this equation, when spatial fluctuations inhomogeneities in quantity $N(t)$ are neglected. this is given by the equation

$$\frac{dN_i}{dt} = -c_i N_i\,(t) \tag{17}$$

where the initial conditions are $N_i(t = 0) = N_0$, the number density of species i at time $t = 0$; constant $c_i > 0$, is called standard kinetic equation and $c_i > 0$ is a constant.

The solution of the Eq. (15) is as follows:

$$N_i\,(t) = N_0 e^{-c_i t} \tag{18}$$

Or

$$N(t) - N_0 = c_0 D_t^{-1} N(t) \tag{19}$$

As D_t^{-1} is the integral operator, Haubold and Mathai [5] described the fractional generalization of the standard kinetic Eq. (15) as

$N\,(t) - N_0 = c_0^v D_t^{-v} N\,(t)$

Where D_t^{-v} is the Riemann-Liouville fractional integral operator; Miller and Ross [7]) defined by

$${}_0D_t^{-v} N\,(t) = \frac{1}{\Gamma(\nu)} \int_0^t (t-u)^{v-1} f\,(u)\,du,\, R\,(v) > 0 \tag{20}$$

The solution of the fractional kinetic equation (18) is given by (see Haubold and Mathai [5])

$$N(t) = N_0 \sum_{k=0}^{\infty} \frac{(-1)^{vk}}{\Gamma(vk+1)} (ct)^{vk} \tag{21}$$

Also, Saxena, Mathai and Haubold [17] studied the generalizations of the fractional kinetic equation in terms of the Mittag-Leffler functions which is the extension of the work of Haubold and Mathai [5].

In the present work, we studied of the generalized fractional kinetic equation. The advanced generalized fractional kinetic equation and its solution, obtained in terms of the M—function.

1.4 Advanced Generalized Fractional Kinetic Equations

In this section, we investigate the solution of advanced generalized fractional kinetic equation. The results are obtained in a compact form in terms of generalized M—function. The result is presented in the form of a theorem as follows:

Theorem 1 *If* $b \geq 0, c > 0, \alpha > 0, \beta > 0, \gamma > 0, \delta > 0, \rho > 0$ *and* $(\gamma\alpha - \beta) > 0$ *then for the solution of the Advanced generalized fractional kinetic equation*

$$N(t) - N_0{}^{\alpha,\beta,\gamma,\delta,\rho}{}_p M_q^{k_1,\ldots k_p, l_1,\ldots l_q;c}(t)(t) = -\sum_{r=1}^{n} \binom{n}{r} c^{r\alpha} D_t^{-r\alpha} N(t) \tag{22}$$

Then

$$N(t) = N_0{}^{\alpha,\beta,\gamma,\delta,\rho}{}_p M_q^{k_1,\ldots k_p, l_1,\ldots l_q;c}(t)(t) \tag{23}$$

Proof We have,

$$\begin{aligned} &N(t) - N_0 \sum_{n=o}^{\infty} \frac{(a_1)_n \cdots (a_p)_n (\gamma)_n}{(b_1)_n \cdots (b_q)_n} \frac{(\delta)_n}{(\rho)_n} \frac{(-c^a)^n}{b_1^n \cdots b_q^n} \frac{(t-b)^{(n+\gamma)\alpha-\beta-1}}{(n_i)!n!\Gamma((n+\gamma)\alpha-\beta)} \\ &= -\sum_{r=1}^{n} \binom{n}{r} c_0^{r\alpha} D_t^{-r\alpha} N(t) \end{aligned} \tag{24}$$

Taking the Laplace transforms of both the sides of Eq. (24), we get

$L\{N(t)\} - L.$

$$\begin{aligned} &L\{N(t)\} - L\left\{N_0 \sum_{n=o}^{\infty} \frac{(a_1)_n \cdots (a_p)_n (\gamma)_n}{(b_1)_n \cdots (b_q)_n} \frac{(\delta)_n}{(\rho)_n} \frac{(-c^a)^n}{b_1^n \cdots b_q^n} \frac{(t-b)^{(n+\gamma)\alpha-\beta-1}}{(n_i)!n!\Gamma((n+\gamma)\alpha-\beta)}\right\} \\ &= -L\left\{\sum_{r=1}^{n} \binom{n}{r} c_0^{r\alpha} D_t^{-r\alpha} N(t)\right\} \end{aligned} \tag{25}$$

From Lorenzo and Hartley [6] (1999) using shifting theorem for Laplace transform, we have

$$\begin{aligned} N(s) - N_0 \frac{(a_1)_n \ldots (a_p)_n}{(b_1)_n \ldots (b_q)_n} \frac{(\delta)_n}{(\rho)_n} \frac{1}{\prod_{i=1}^{p} (n_i)! n! b_1^n \ldots b_q^n} \frac{s^{\beta} e^{-bs}}{(s^{\alpha}+c^{\alpha})^{\gamma}} \\ = -\left\{ \sum_{r=1}^{n} \binom{n}{r} c^{r\alpha} s^{-r\alpha} N(t) \right\} \end{aligned} \tag{26}$$

Or,

$$\begin{aligned} N(s) - N_0 \frac{(a_1)_n \ldots (a_p)_n}{(b_1)_n \ldots (b_q)_n} \frac{(\delta)_n}{(\rho)_n} \frac{1}{\prod_{i=1}^{p} (n_i)! n! b_1^n \ldots b_q^n} \frac{s^{\beta} e^{-bs}}{(s^{\alpha}+c^{\alpha})^{\gamma}} \\ = -\left[{}_0^n c_1 c^{\alpha} s^{-\alpha} +_0^n c_2 c^{2\alpha} s^{-2\alpha} \ldots {}_0^n c_n c^{n\alpha} s^{-n\alpha} \right] N(s) \end{aligned} \tag{27}$$

$$N(s)(1 + c^{\alpha} s^{-\alpha})^n = N_0 \frac{(a_1)_n \ldots (a_p)_n}{(b_1)_n \ldots (b_q)_n} \frac{(\delta)_n}{(\rho)_n b_1^n \ldots b_q^n \prod_{i=1}^{p} (n_i)! n!} \frac{s^{\beta-\alpha\gamma} e^{-bs}}{(1 + c^{\alpha} s^{-\alpha})^{\gamma}} \tag{28}$$

$$N(s) = N_0 \frac{(a_1)_n \ldots (a_p)_n}{(b_1)_n \ldots (b_q)_n} \frac{(\delta)_n}{(\rho)_n b_1^n \ldots b_q^n \prod_{i=1}^{p} (n_i)! n!} \frac{s^{\beta-\alpha\gamma} e^{-bs}}{(1 + c^{\alpha} s^{-\alpha})^{\gamma+n}} \tag{29}$$

$$N(s) = N_0 \frac{(a_1)_n \ldots (a_p)_n}{(b_1)_n \ldots (b_q)_n} \frac{(\delta)_n}{(\rho)_n b_1^n \ldots b_q^n \prod_{i=1}^{p} (n_i)!} \frac{s^{\beta-\alpha(\gamma+n)+n\alpha} e^{-bs}}{(1 + c^{\alpha} s^{-\alpha})^{\gamma+n}} \tag{30}$$

$$\begin{aligned} N(t) = N_0 \frac{(a_1)_n \ldots (a_p)_n}{(b_1)_n \ldots (b_q)_n} \frac{(\delta)_n}{(\rho)_n \prod_{i=1}^{p} (n_i)!} \frac{1}{b_1^n \ldots b_q^n} \\ \sum_{n=o}^{\infty} \left(\frac{-c^{\alpha}}{s^{\alpha}} \right)^n \frac{(\gamma+n)_n s^{\beta-\alpha\gamma} e^{-bs}}{n!} \end{aligned} \tag{31}$$

Now, taking inverse Laplace transform, we get

$$\begin{aligned} N(t) = N_0 \frac{(a_1)_n \ldots (a_p)_n}{(b_1)_n \ldots (b_q)_n} \frac{(\delta)_n}{(\rho)_n \prod_{i=1}^{p} (n_i)!} \frac{1}{b_1^n \ldots b_q^n} \\ \sum_{n=o}^{\infty} (-c^{\alpha})^n \frac{(\gamma+n)_n}{n!} L^{-1} \left\{ s^{\beta-\alpha\gamma-\alpha n} e^{-bs} \right\} \end{aligned} \tag{32}$$

$$\begin{aligned} N(t) = N_0 \sum_{n=o}^{\infty} \frac{(a_1)_n \ldots (a_p)_n}{(b_1)_n \ldots (b_q)_n} \frac{(\delta)_n}{(\rho)_n \prod_{i=1}^{p} (n_i)!} \frac{(-c^{\alpha})^n}{b_1^n \ldots b_q^n} \frac{(y+n)_n}{n!} \\ \frac{(t-b)^{\alpha\gamma+\alpha n-\beta-1}}{\Gamma((\gamma+n)\alpha-\beta)} \end{aligned} \tag{33}$$

$$N(t) = N_0^{\alpha,\beta,(\gamma+n),\delta,\rho}{}_p M_q^{-c^{\alpha},b_1,\ldots b_n;b}(t) \tag{34}$$

This is the complete proof of the theorem.

1.5 Special Cases

Corollary 1 *If we take* $(a_1)_n \ldots (a_p)_n = 1 = (b_1)_n \ldots (b_q)_n$, $\delta = 1, \rho = 1$ *and* $b_1^n \ldots b_q^n = 1 \prod_{i=1}^{p} (n_i)! = 1$ *then for the solution of the Advanced generalized fractional kinetic equation*

$$N(t) - N_0^{\alpha,\beta,\gamma,1,1}{}_1M_1^{-c^\alpha,1;b}(t) = -\sum_{r=1}^{n} \binom{n}{r} c^{r\alpha} D_t^{-r\alpha} N(t) \tag{35}$$

There holds the result

$$N(t) = N_0^{\alpha,\beta,(\gamma+n),1,1}{}_1M_1^{-c^\alpha,1;b}(t) \tag{36}$$

In view of the relation (33), this result coincides with the main result of Chaurasia and Pandey [3].

Corollary 2 *If we put* $b = 0$ *in Corollary (1) then the solution of the Advanced generalized fractional kinetic equation reduces to the special case of Theorem (1) in Chaurasia and Pandey [3] (2010), given as follows:*

For the solution of

$$N(t) - N_0^{\alpha,\beta,\gamma,1,1}{}_1M_1^{-c^\alpha,1;0}(t) = -\sum_{r=1}^{n} \binom{n}{r} c^{r\alpha} D_t^{-r\alpha} N(t) \tag{37}$$

There holds the result

$$N(t) = N_0^{\alpha,\beta,(\gamma+n),1,1}{}_1M_1^{-c^\alpha,1;0}(t) \tag{38}$$

Corollary 3 *If we put* $\beta = \gamma\alpha - \beta$ *in Corollary (1) then the solution of the Advanced generalized fractional kinetic equation reduces to the special case of Theorem (1) in Chaurasia and Pandey [3] (2010), which is given as follows:*

For the solution of

$$N(t) - N_0^{\alpha,\gamma\alpha-\beta,\gamma,1,1}{}_1M_1^{-c^\alpha,1;b}(t) = -\sum_{r=1}^{n} \binom{n}{r} c^{r\alpha} D_t^{-r\alpha} N(t) \tag{39}$$

There holds the formula

$$N(t) = N_0^{\alpha,\gamma\alpha-\beta,(\gamma+n),1,1}{}_1M_1^{-c^\alpha,1;b}(t) \tag{40}$$

Corollary 4 *If we put* $b = 0$ *in Corollary (3) then the solution of the Advanced generalized fractional kinetic equation reduces to another special case of Theorem (1) in Chaurasia and Pandey [3] which is given as follows:*

For the solution of

$$N(t) - N_0^{\alpha,\gamma\alpha-\beta,\gamma,1,1}{}_1M_1^{-c^\alpha,1;0}(t) = -\sum_{r=1}^{n}\binom{n}{r}c^{r\alpha}D_t^{-r\alpha}N(t) \tag{41}$$

There holds the formula

$$N(t) = N_0^{\alpha,\gamma\alpha-\beta,(\gamma+n),1,1}{}_1M_1^{-c^\alpha,1;0}(t) \tag{42}$$

This completes the analysis.

1.6 Result and Discussion

In this present part of talk, we have introduced a fractional generalization of the standard kinetic equation and a new special function given by author and also established the solution for the computational extension of Advanced fractional kinetic equation. The results of the computational extension Advanced generalized fractional kinetic equation and its special cease are same as the results of Chaurasia and Panday [3] (2010). And also, the relations function to the various standard functions is discussed in this present section.

2 Section II

The aim of the second part of talk to obtain the solution of differential equations involving Modified Riemann-Liouville fractional derivative i.e. ms-operator in addition to it Levy-type one sided probability density function. Some scientist studied a small number of interesting problems regarding diffusing processes in media with fractal geometry [15], and formulated fractional diffusion [11].

The theory of stochastic processes may be regarded as the 'dynamic' part of statistical theory with a multiplicity of applications.By a stochastic process we shall in the first place mean some possible actual, e.g. physical process in the real world that has some random or stochastic element involved in its structure. The most elementary examples of stochastic processes are classical enough to be discussed by statisticians are random sequences in which the variable X_r at time t_r is independent of the entire previous set of X's. The statistical interest of such sequences lies in the properties of derived variables or sequences, such as the cumulative sums

$$S_r = X_1 + X_2 + \cdots + X_r \tag{43}$$

The process S_r is called a 'random walk', as it represents the position at time t_r of a person talking a random step X_r independently of his previous ones.

The Random variables are bases of starting the theory of probability. Non standard random walk, which is depends on a variable step size x. Generally, one-sided probability distribution function of asymptotic type (large x values) [13]

$$f(x) \sim x^{1-\mu}, \mu > 0, x > 0 \tag{44}$$

And are called Levy flights since Eq. (1) represents a Levy distribution [4], and the trace of the site visited by the walker forms a set of fractal dimension μ [10]. Here, we will show that for a certain class of one sided probability densities $f(x)$, defined on R_+, Levy index μ can be related to the order of a fractional integral operator. We apply the fractional calculus [15] based on the Riemann-Liouville operator ${}_0D_x^{-q}$ given by

$${}_0D_x^{-q} f(x) = \frac{1}{\Gamma(q)} \int_0^x (x-y)^{q-1} f(y) dy, 0 < q \tag{45}$$

The author introduced a new fractional integral operator by generalization of classical c ${}_0^{ms}D_x^{-v}$ which deal with differentiable functions is denoted by ${}_0^{ms}D_x^{-v}$ and called M-S operator, which is a modification in Riemann-Liouville operator, and Ali [2] operator which is defined as follows:

$${}^{ms}{}_0D_x^{-v} f(x) = \frac{1}{\Gamma(\nu)} \int_0^x (x-t)^{v-1} f(at)\, dt, R(v) > 0, a > 0, n > 0 \tag{46}$$

When $a = 1$ it converts in the Classical definition of Reimann—Liouville fractional calculus operator.

The fractional differential operator ${}_0^0D_x^q$ for $q > 0$ is given by the definition

$${}_0D_x^q f(x) = \frac{d^n}{dx^n}({}_0D_x^{q-n} f(x)), q - n < 0 \tag{47}$$

where ${}_0D_x^{q-n}$ for $q - n < 0$ is defined in (48) indicating that for differentegrable (i.s. differentiable and Integrable) functions f(x). The operation fractional differentiation can be decomposed in to a fractional integration followed by an ordinary differentiation $\frac{d^n}{dx^n}$. Here n is the least positive integer greater than q. If 0 < q < 1 then we choose n 1 and if 1 < q < 2 we take n 2 and so on.

Just about a decade ago it had already been suggested [19] that Levy-type probability function are not just solutions of standard type differential equations but moderately are to be represented by integral equations with an integral kernel $K(x-y)\ (x-y)^{-\alpha}$ of some fractional order α. Newly, a class probability densities $f(x)$ such as Fox function representation [18] given by the integral equation

$$x^m f(x) = \int_0^x (x-y)^{-\alpha} f(y)\, dy, x > 0 \tag{48}$$

where $\alpha > 0$ is non integer number, and m is positive integer ($m = 1, 2, 3\ldots$). The special class of normalized one-sided Levy-type probability densities is developing by deeply study and analysis. Therefore,

$$f(x) = \frac{a^{\mu}}{\Gamma\mu} x^{-\mu-1} e^{\frac{-a}{x}}, a > 0, x > 0 \tag{49}$$

is a solution of fractional differential equation.

$$a^{vms}{}_0D_x^{-v} f(x) = \frac{a^{5\mu-2}}{\Gamma\mu} x^{\mu-1} e^{-1/x} \tag{50}$$

If we recover the Levy-index μ as the fractional order q of the differential operator ${}_0^A D_x^{-v}$. We note that $f(x)$, given in (49), tends to zero for $x \to 0$ and has a for large x-values the desired asymptotic behavior Eq. (44). To show the $f(x)$ for several values of μ. It is obvious that the asymptotic power-law tail satisfies the scaling property $f(\lambda x) = \lambda^{-\mu-1} f(x)$, where λ is the scaling factor and μ has been identified as a fractal (similarity) dimension [10]. In the study of random walks, of ion-channel getting kinetics [12] and in the understanding of regular structures and pattern formation in biophysical systems [12, 22] self-similar processes play a dominant part based on Levy dynamics. The class of probability densities (49) is non-negative, and has the moments ($k = 0, 1, 2, \ldots$)

$$\left\langle x^k \right\rangle = \int_0^x x^k f(x) dx = a^k \frac{\Gamma(\mu - k)}{\Gamma\mu} \tag{51}$$

Including normalization $\left\langle x^0 \right\rangle = 1$

To show that (49) is a solution of (50) we insert $f(x)$, given in (49), into the integral (48) and substitute

$$t = \frac{ax}{(xz + a)} \tag{52}$$

We have,

$$a^{vms}{}_0D_x^{-v} f(x) = \frac{a^v}{\Gamma(v)} \frac{a^{\mu}}{\Gamma\mu} \left\{ \int_0^x (x-t)^{v-1} (at)^{-\mu-1} e^{-a/at} dt \right\} \tag{53}$$

$$a^{vms}{}_0D_x^{-v} f(x) = \frac{a^v}{\Gamma(v)} \frac{a^{4\mu-2}}{\Gamma\mu} \left\{ \int_0^x \frac{\left(\frac{z}{a}\right)^{v-1}}{(xz + a)^{v-1}} e^{-z/a} dt \right\} \tag{54}$$

On putting $v = \mu$ the remaining integral is just Euler's definition of the Γ-function $\Gamma\mu$ for $\mu > 0$. Thus we have,

$$a^{vms}{}_0D_x^{-v} f(x) = \frac{a^{5\mu-2}}{\Gamma\mu} x^{\mu-1} e^{-1/x} \tag{55}$$

Which is more correct and feasible due to new applied ms-operator. This operator can applied in physics and bio-engineering, bio-informatics, Statistical mechanics, Control engineering, PSO Theory etc.

3 Conclusions

The fractional calculus is little studied and it is very useful in diffusion process in media with fractal geometry [11, 15] and also have formulated fractional diffusion [23] and fractional Boltzmann equations [14]. In this paper we have studied a new generalized special function i.e. generalized M–function and a new fractional calculus operator i.e. ms-operator both will be applied in applications of fractional calculus in signal processing, Artificial Intelligence, Bio-Medical engineering, Mechanical engineering, Automobile engineering, Control engineering and automation . In designing of control systems and new machines, advantages and disadvantages of human being operator have to be taken into account. Better work organization should ensure high inspiration, increase competencies, personnel firmness and human work efficiency with new approach of Mathematical Modeling using above New Mathematical functions and Operators. It seems to be obvious that the new approach to the process of man-machine interaction should be developed. It has to be based on the latest achievements of the software engineering, new unconventional possibilities of hardware systems and deep knowledge about human performance. Better employment of human and modern manufacturing systems' advantages by proper man-machine interaction design should be the aim of the multidisciplinary research.

References

1. Agarwal, R.P.: A propos d'une note de m. pierre humbert. C. R. Acad. Sci. Paris **236**(21), 2031–2032 (1953)
2. Ali, M.: Fractional calculus operators and their applications in science and engineering. Ph.D. thesis, Jiwaji University Gwalior (2014)
3. Chaurasia, V.B.L., Pandey, S.C.: Computable extensions of generalized fractional kinetic equations in astrophysics. Res. Astron. Astrophys. **10**(1), 22–32 (2010)
4. Feller, W.: An Introduction to Probability Theory and its Applications. Wiley, New York (1971)
5. Haubold, H., Mathai, A.: The fractional kinetic equation and thermonuclear functions. Astrophys. Space Sci. **273**(1–4), 53–63 (2000)
6. Lorenzo, C.F., Hartley, T.T.: Generalized functions for the functional calculus. Technical Report NASA/TP-1999-209424/REV1, NASA (1999)
7. Miller, K.S., Ross, B.: An Introduction to the Fractional Calculus and Fractional Differential Equations. Wiley, New York (1993)
8. Mittag-Leffler, G.: Sur la nouvelle fonction Eα (x). C. R. Acad. Sci. Paris **137**, 554–558 (1903)
9. Mittag-Leffler, G.M.: Sur la representation analytique d'une branche uniforme d'une fonction monogene. Acta Mathematica **29**, 101–181 (1905)
10. Montroll, E.W., Shlesinger, M.F.: On 1/f noise and other distributions with long tails. Proc. Natl. Acad. Sci. **79**(10), 3380–3383 (1982)

11. Nigmatullin, R.: The realization of the generalized transfer equation in a medium with fractal geometry. Physica Status Solidi (b) **133**(1), 425–430 (1986)
12. Nonnenmacher, T.: Fractal scaling mechanisms in biomembranes. Eur. Biophys. J. **16**(6), 375–379 (1989)
13. Nonnenmacher, T.: Fractional integral and differential equations for a class of levy-type probability densities. J. Phys. A: Math. Gen. **23**(14), L697S (1990)
14. Nonnenmacher, T., Nonnenmacher, D.: Towards the formulation of a nonlinear fractional extended irreversible thermodynamics. Acta Physica Hungarica **66**(1–4), 145–154 (1989)
15. Oldhman, K., Spanier, J.: The Fractional Calculus: Theory and Applications of Differentiation and Integration to Arbitrary Order. Academic Press, New York (1974)
16. Perdang, J.: Lecture Notes in Stellar Stability. Part I and II. Instito di Astronomia, Padov (1976)
17. Saxena, R., Mathai, A., Haubold, H.: On generalized fractional kinetic equations. Phys. A: Stat. Mech. Appl. **344**(3), 657–664 (2004)
18. Schneider, W.: Stable distributions: fox function representation and generalization. In: Albeverio, S., Casati, G., Merlini, D. (eds.) Stochastic Processes in Classical and Quantum Systems. LNP, vol. 262, pp. 497–511. Springer, Berlin (1986)
19. Seshadri, V., West, B.J.: Fractal dimensionality of lévy processes. Proc. Natl. Acad. Sci. **79**(14), 4501 (1982)
20. Sharma, K.: On application of fractional differential operator to the K4 function. Boletim da Sociedade Paranaense de MatemĞtica **1**(30), 91–97 (2012)
21. Sharma, M., Ali, M.F., Jain, R.: Advanced generalized fractional kinetic equation in astrophysics. Prog. Fract. Differ. Appl. **1**(1), 65–71 (2015)
22. West, B.J., Bhargava, V., Goldberger, A.: Beyond the principle of similitude: renormalization in the bronchial tree. J. Appl. Physiol. **60**(3), 1089–1097 (1986)
23. Wyss, W.: The fractional diffusion equation. J. Math. Phys. **27**(11), 2782–2785 (1986)

Part II
Human–Computer Interfaces

A Model-Driven Engineering Approach to the Evaluation of a Remote Controller of a Movement Assistant System

Anna Derezińska and Karol Redosz

Abstract In Model-Driven Engineering, structural and behavioral models can be applied in code generation. State machine models are an important mean of describing behavioral features of a system. This paper presents development of a controller of an exoskeleton system and its interface. Exoskeleton is a system to assist movement abilities of a person that is carrying on a mechanical suit and controlling the system operations. The controller development is based on UML class models and their state machines. The models are transformed into code, extended to an executable application and run with a test scenario. Different transformation approaches are compared in experiments and adequacy of the derived implementation is observed. Moreover, the model can be used as a benchmark in experiments with other code generation tools.

Keywords UML · MDE · Code generation · State machine · Statechart · Exoskeleton system · Remote controller

1 Introduction

An exoskeleten system is a kind of an orthotic robot that supports basic movement of persons with various motoric disorders, including spinal cord injury [3]. A system has a form of a mechanical skeleton that is attached to a user body, to its shoulders and legs [9, 13]. There are two main goals of the system. (i) A person has an ability to move around in a vertical position, without a wheelchair. (ii) A person can make exercises, mimicking the typical pattern of leg movements, which is very beneficial for a certain range of wheelchair users.

A user operates a system directing commands to system components. Therefore, a system is equipped in an electronic device to control moves of particular system parts. A user can use an interface of a remote controller to cooperate with the skeleton. An important issue is specification of the system behavior in the terms corresponding to

A. Derezińska (✉) · K. Redosz
Institute of Computer Science, Warsaw University of Technology, Warsaw, Poland
e-mail: a.derezinska@ii.pw.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_7

a real system. Furthermore, this specification should be transformed to executable code in an automatic way. Therefore, the system interface, behavior, and their modifications can be observed and interpreted.

In this paper we showed how a problem of development and evaluation of a control system of an exoskeleton can be dealt with a Model-Driven Engineering Approach (MDE) [10, 14]. The main idea of MDE is to create appropriate structural and behavioral models and transform them into an executable code. Consequently a system can be effectively evaluated by testing of the corresponding application. Considering approaches that use UML models as a source of a model to code transformation (m2c in short) the most of them concentrate on structural models, i.e. class diagrams. However, behavioral models are also an important source of code generation services. The mostly used in this context are UML state machines and other automata-based relatives, like Finite State Machines, and Harel statecharts [7].

In the presented solution, a system development is based on m2c covering not only classes but also comprehensive UML state machine models. First, we present a model consisting of UML class and state machine submodels. The model was developed to serve as a mock system in the development of an exoskeleton [18] and has an intuitive interpretation in the real system. This paper is not dealing with design problems of mechanical structure of the system and the system power supply [1]. We focus on an interface and a system remote controller. Other parts of the exoskeleton system are modeled with so many details that allow for effective testing of this controller. The model specifies behavior of system classes using various notions of UML state machine. Hence, different scenarios of a user-system cooperation can be examined.

Having the model, its corresponding application was developed and evaluated. The main challenge was a complete and accurate automatic transformation of complex state machine models. Therefore, different MDE approaches were verified. The created model was used in experiments with three code generation tools, two commercial [11, 12], and FXU—an m2c tool developed in the Institute of Computer Science WUT [16, 17]. The adequacy of the final applications was verified. Moreover the presented system can be used as a benchmark model that could be served as a transformation source in evaluation of different model to code transformation tools.

The paper is structured as follows. The next section reviews a related work. The basic ideas of the exoskeleton system and its models are described in Sect. 3. The code generation tools and experiment results are discussed in Sect. 4. Finally, Sect. 5 concludes the paper.

2 Related Work

Transformation of models is a central concept of Model-Driven Engineering (MDE) [10, 14]. Transformations can be focused on the domain-specific models [8], but transformation of UML models into a code of general purpose programming languages is of the most practical impact.

A taxonomy of code generators is summarized in [2]. The authors compare three tools using a set of UML models and point out at various errors in the transformation results, such as compilation errors, execution errors, information loss and missing notation. However the research was focused on the class models, and state machines were not taken into account in model transformations.

Transformations of class models are commonly supported by CASE tools. Structural models are sometimes enhanced with data from profiles, OCL constraints, etc. [4, 5].

Different approaches to transformations of behavioral models are surveyed in [7]. Various solutions are used for the implementation of state concepts, from simple state attributes [15], to the run-time libraries supporting all notions of state machine diagrams [16]. Many code generation approaches, and the most attempts to formalize state machine semantics, are covering a small subset of UML state machines, ignoring complex states and several pseudostates like fork and join, history concepts etc. However, these advanced state machine concepts are especially important for modelling of control systems and embedded systems.

Another variant of dealing with models is their direct execution using a dedicated virtual machine instead of building an application from a model [20].

Different robots are developed in order to assist people movements [3]. Most of the problems discussed in this context refer to mechanical solutions of the equipment. One of the systems available to the patients is the ReWalk exoskeleton [19]. Its general functionality is similar to the system considered in this paper, but the details about its control design are not known [9]. Another system of this kind has been developed in WUT [13]. Programing of its control system inspired the work presented here. The modeling and simulation performed for this system [1] refers to problems of the kinematic research and power supply that are not discussed in this paper.

3 Modelling of Exoskeleton System

An orthotic robot supports basic movements of a handicapped person. A robot of this kind is worn by a person in a form of a medical exoskeleton. An exoskeleton enables a person to stand up, sit down and walk in a vertical position over ground. Using some robots it could be possible to walk down and up the stairs. It also helps patients to re-learn walking and perform medical training during rehabilitation therapy. A system is equipped with a control unit, so a patient controls the movement with a set of commands.

3.1 Exoskeleton System

One of the authors was involved in the research on a robot of this kind within the UE project *Eco-mobility* [13]. It included the development of the robot control system and its software. Based on this experience, various conceptual models were created

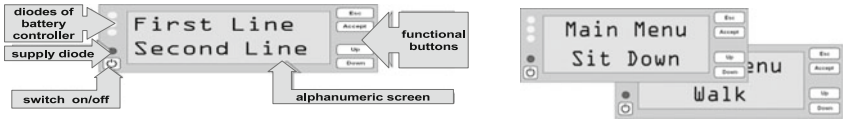


Fig. 1 Remote controller of the orthotic model and its screens

for the control system of an orthotic robot [18]. The models covered mainly structural UML models based on classes and description of behavior using state machines.

The modeled system assists in walking, standing up and sitting down by a person carrying an exoskeleton. A system of each leg is equipped with two active joints: a knee joint and a hip join. There is a set of sensors in the upper part of the skeleton. The exoskeleton is battery powered and facilitated with a remote controller. The controller is mounted to a hand wrist.

The remote controller has several buttons, LED diodes, and an alphanumerical screen (Fig. 1). On the left side of the controller, there are three diodes showing a battery status, a diode indicating power supply, and a power switch. Four control buttons are placed on the right side, namely: *Esc*, *Accept*, *Up* and *Down*. The alphanumeric screen allow to display two lines of information.

In the remote controller, a hierarchical menu is presented on the screen. It can be navigated by buttons. Examples of two menu screens are given in Fig. 1. Main menu corresponds to selection of movement options. After expanding a *Walk* submenu, a user can adjust a number of steps and launch a walking function.

3.2 UML Structural Models

The UML model focuses on a basic exoskeleton system and its system remote control. The main parts of the structural model are shown in Fig. 2.

The *RemoteController* class stores a current status of the controller, a currently selected function, and a number of steps to walk. It also keeps information about a battery supplying the controller. Available operations allow to activate the controller buttons and to switch the controller lightning indicators.

Control signals from the remote controller can be passed to the *ExoSkeleton* class. This class aggregates *Joint* classes, depicting joints in hips and knees of two legs. *ExoSkeleton* also includes *Sensor* objects and the *MemoryCard* class.

3.3 UML Behavioral Models

Behavior of each class of the model was specified with a state machine. Any state machine was intended to completely describe all events influencing an object state. In Figs. 3 and 4, a part of the behavioral model of the system is presented.

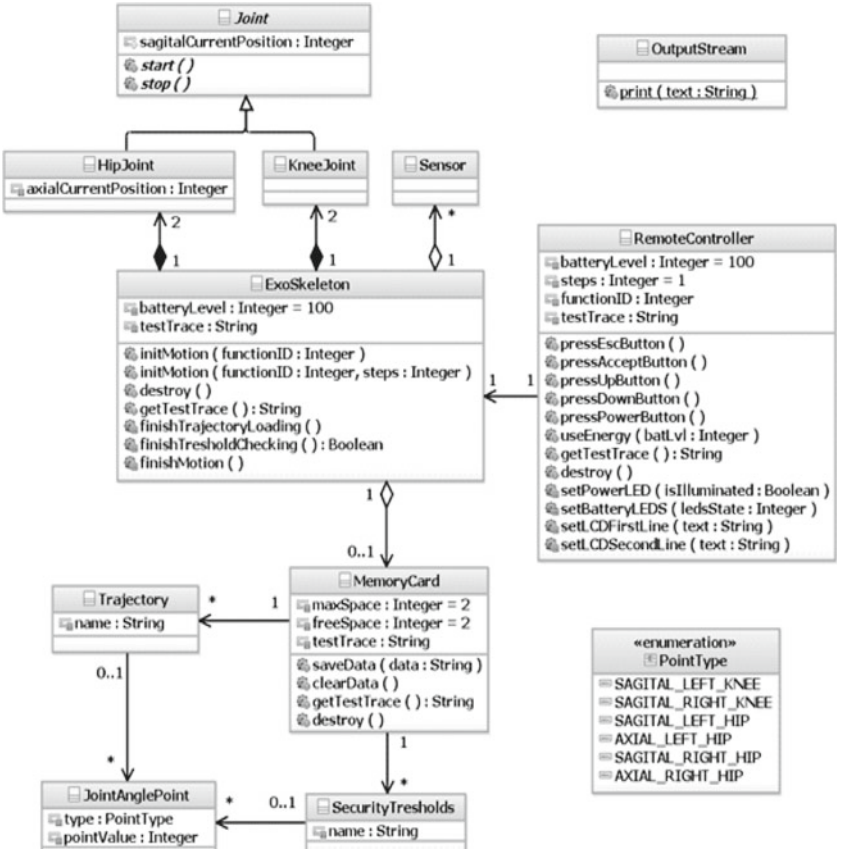


Fig. 2 ExoSkeleton—a part of the class model

An object of the *ExoSkeleton* class can stay in one of three major states: *Idle*, *MotionInitializing* and *Motion* (Fig. 3). If the system is switched on and not operating, the skeleton is idle. The movement can be launched and initialized. During initialization two operations should be completed: loading of a trajectory and checking of a threshold. Both operations can be performed in any order. This situation was represented by orthogonal regions in the complex state *MotionInitializing*. The operations are started in parallel using a fork pseudostate. After the operations have been finished, a transition to the *Motion* state can be done. This fact is shown by the join pseudostate.

Apart from triggers of operation calls that are associated with transitions, we have used some internal activities performed on entry to a state, on exit from a state, or while being in a state, e.g., motion in the *Motion* state. An exoskeleton movement performed in the *Motion* state can be abandon, and the object returns to its idle state

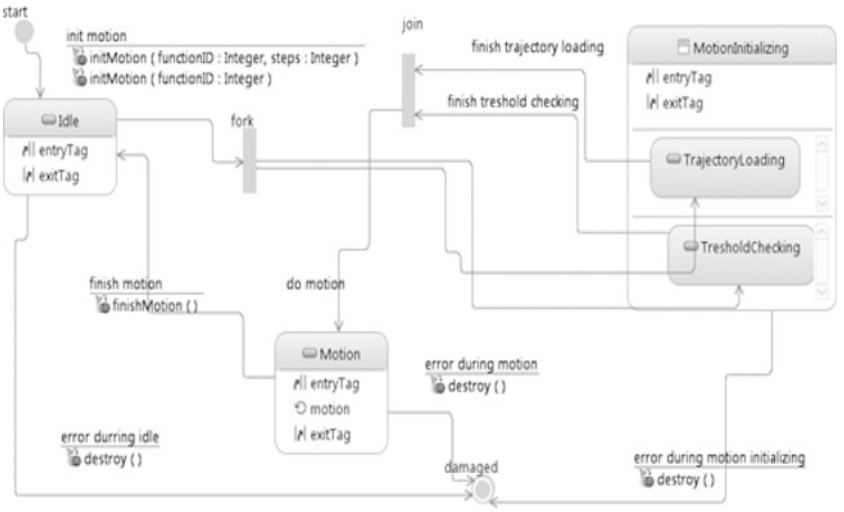


Fig. 3 ExoSkeleton—behavioral state machine

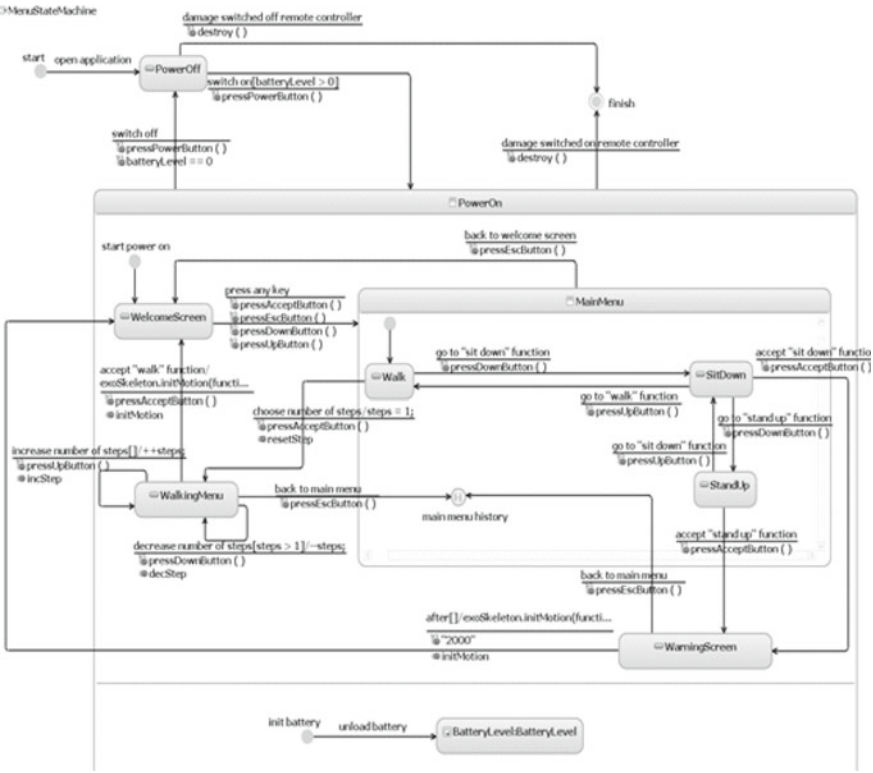


Fig. 4 RemoteController—behavioral state machine

waiting for next events. In each state an error can occur and afterward the object will be transformed into its final state.

The most interesting part of the system was the behavioral specification of the remote controller. The main parts of the model describing the *RemoteControler* class are shown in Fig. 4. The controller can remain in two general modes: *PowerOff* and *PowerOn*. The state corresponding to the *PowerOn* mode consists of two orthogonal regions. The upper region represents actions performed by the device. The lower region is devoted to monitoring of the battery level. The *BatteryLevel* state is specified by an external statemachine. The upper region of the *PowerOn* state reflects the control facilities of the device. From the welcome screen we can move on to the main menu. In the main menu three substates can be handled: *Walk*, *Sit down*, and *Stand up*. Detailed control of walking is applied in the walking menu.

Coming back from the walking menu or from the warning screen we enter the same substate of the main menu. This requirement is assured by the usage of a history substate.

In state machines of the remote controller and the battery level, different kinds of triggers, guard conditions, choice pseudostate and other state machine concepts are used to model their behavior.

4 Experiments

4.1 Code Generation

Many CASE tools support transformation of UML classes into code. A result of a transformation constitutes draft of classes in a given programming language. These classes include fields transformed from attributes, signatures of methods originated from operations, and corresponding implementation of class relations. The classes can be accompanied with implementation details according to stereotypes used in a model, for example reflecting design patterns, or language-specific features [2, 4, 5]. More complicated is transformation from the behavioral models. States and other concepts of state machines have not straightforward representation in general purpose programming languages [7, 15].

In experiments three different tools were applied that support transformation from UML state machines:

- Framework for eXecutable UML v.5.0 (in short FXU),
- IBM Rational Software Architect v.9.0 (in short RSA),
- IBM Rational Rhapsody Developer v.8.0.5 (in short Rhapsody).

Framework for eXecutable UML (FXU) is a tool for code generation from UML class models and state machines [17]. Since its first version, C# code was the target of the transformation supported by the tool [16]. Further versions were enhanced with new facilities, such as off-line tracing of state machines [6], incorporation of a

part of OMG MARTE profile, different UML semantic variants of state machines, and code reuse in the multiple model transformation.

The tool (versions 0.1–0.5) is a stand-alone environment. It inputs an external file with an UML model created with a CASE tool. The current, 5.0 version of FXU accepts as an input UML files in the XMI/OMG UML v.2.2 format An output of the transformation process is a C# project with generated code fragments. The project together with the FXU Run-time Library can be used for creation of a C# application.

The IBM Rational Software Architect [12] is a commercial CASE tool. It is used to design, modelling and development of software with the UML notation. It supports many kinds of model transformations. The tool is built on the top of the Eclipse environment, therefore it is integrated well with Java development tools. Java code can be generated from a class model and state machines. However, only simple state machines are represented, as a transformation of state machines is based on a simple usage of an additional attribute in a class. Dealing with the C# language requires a special extension of the tool to support the .NET environment. Though, a model transformation to C# code takes into account only class models.

The IBM Rational Rhapsody tool [11] originates from the tool developed in I-Logix in 1996, and further advanced by Telelogic. It was one of the first tools that implemented code generation from state machines with complex states. Currently the tool is a member of the IBM portfolio. Its functionality covers preparation of UML models, as well as transformation to an executable code. It supports code generation of UML class and state machine models in C, C++, Java and Ada. Code generation for C# is limited to a class model only, no state machines are assisted. A special library OXF makes possible to use predefined state machine concepts, such as pseudostates or events.

4.2 Model Testing

In approaches based on automatic code generation, a final outcome is an executable application. Such an application should reflect the ideas expressed in source behavioral models. An open issue is how we can verify that the target application conforms to expectations articulated by the model developers. In particular, does an m2c transformation if correctly performed.

The problem is correspondence of the generated code to the expectation of a developer reflected in UML models. A developer expresses several system ideas in the model designs. Therefore a verity of state machine concepts provided by the UML specification can be used in models. The meaning of the model elements, their syntax and semantics should be consistent with UML specification.

A verification procedure of the presented model was based on a benchmark scenario. Therefore, a sequence of events was prepared. It reflected procedures of an exoskeleton usage, which were expressed in notion of events accepted by the model. The events triggered different transitions of the state machines. This kind of behavior can be run and observed at the model level. However, the aim of the experiment was

examination of the corresponding applications at the program level. Therefore, the sequence of events was transformed into a list of appropriate function calls. Finally, the list was prepared in two programming languages, C# and Java, as these languages were the target of the model to code transformations.

The test scenario covered all states and all transitions of the state machines. All sequences of the typical usage of the remote controller were taken into account. The exceptional and erroneous situations were also applied in the testing scenarios. Moreover, some tools can generate traces that includes series of events. In an FXU trace, the events are originated from an application but are interpreted in the terms of source models [6].

The target programs were developed in C# (for FXU), and Java (RSA, and Rhapsody). The application built with RSA has the biggest number of discrepancy to the original models. It was caused by the insufficient support for the complex states in the m2c transformations. The better results were obtained for Rhapsody and FXU. In general, the application behavior follow the expectation of the modeled system. The one exception was the usage of timing events that were not correctly served (in FXU) or provided compilation errors (in Rhapsody). Only FXU correctly supported all types of an internal behavior in a state, an operation performed on *Entry*, *Do* and *Exit*. The final recommendation of m2c depends on the target language, for C# it is FXU. The effects for Java, although with some limitations, were the best for Rhapsody.

5 Conclusions

In the paper we presented a model-driven approach for evaluation and testing of a control system and its interface. The system structure and its behavior were modeled with UML class and state machine diagrams. Different tools were used in experiments for the model to code transformation. The experiments showed that the tools supporting state machine transformations covering the most of UML concepts, including complex states, can give the valuable solutions. The simplified transformation approaches, as in RSA, were not suitable to evaluation of the modelled system. The results of applications built with use of Rhapsody (Java) and FXU (C#) were helpful in the verification of the expected system behavior.

References

1. Bagiński, K., Jasińska-Choromańska, D., Wierciak, J.: Modelling and simulation of a system for verticalization and aiding the motion of individuals suffering from paresis of the lower limbs. Bull. Pol. Acad. Sci. Tec. Sci. **61**(4), 919–928 (2013)
2. Bajovs, A., Nikiforowa, O., Sejans, J.: Code generation from UML models: state of art and practical implications. Appl. Comput. Syst. Sci. J. Riga Tec. Univ. **14**(1), 9–18 (2013)

3. Bogue, R.: Exoskeletons and robotic prosthetics: a review of recent developments. Ind. Robot Int. J. **36**, 421–427 (2009)
4. Dang, F., Gogolla, M.: Precise model-driven transformations based on graphs and metamodels. In: SEFM 2009, pp. 307–316. Hanoi, Vietnam (2009)
5. Derezińska, A., Oltarzewski, P.: Code generation of contracts using OCL tools. In: Borzemski, L., et al. (eds.) Information Systems Architecture and Technology, Web Information Systems Engineering, Knowledge Discovery and Hybrid Computing, pp. 235–244. Publishing House of Wroclaw University of Technology, Wroclaw (2011)
6. Derezińska, A., Szczykulski, M.: Tracing of state machine execution in model-driven development framework. In: ICIT 2010, pp. 109–112. Gdansk, Poland (2010)
7. Dominguez, E., Perez, B., Rubio, A.L., Zapata, M.A.: A systematic review of code generation proposals from state machine specifications. Inf. Softw. Technol. **54**(10), 1045–1066 (2012)
8. Edwards, G., Brun, Y., Medvidovic, N.: Automated analysis and code generation for domain-specific models. In: WICSA/ECSA 2012, pp. 161–170. Helsinki, Finland (2012)
9. Esquenazi, A.: New bipedal locomotion options for individuals with thoracic level motor complete spinal cord injury. J. Spinal Res. Found. **8**(1), 26–28 (2013)
10. France, R., Rumpe, B.: Model-driven development of complex software: a research roadmap. In: FOSE 2007, pp. 37–54. Minneapolis, MN, USA (2007)
11. IBM: IBM Rational Rhapsody Developer. http://www-03.ibm.com/software/products/en/ratirhap
12. IBM: IBM Rational Software Architect. http://www-03.ibm.com/software/products/en/ratisoftarch
13. Jasińska-Choromańska, D., Szykiedans, K., Wierciak, J., Kolodziej, D., Zaczyk, M., Bagiński, K., Bojarski, M., Kabziński, B.: Mechatronics system for verticalization and the motion of the disabled. Bull. Pol. Acad. Sci. Tec. Sci. **61**(2), 419–431 (2013)
14. Liddle, S.: Model-driven software development. In: Embley, D., Thalheim, B. (eds.) Handbook of Conceptual Modeling, pp. 17–54. Springer, Berlin (2011)
15. Niaz, I.A., Tanaka, J.: An object-oriented approach to generate Java code from UML statecharts. Int. J. Comput. Inf. Sci. **6**(2), 83–98 (2005)
16. Pilitowski, R., Derezińska, A.: Code generation and execution framework for UML 2.0 classes and state machines. In: Sobh, T. (ed.) Innovations and Advanced Techniques in Computer and Information Sciences and Engineering, pp. 421–427. Springer, Netherlands (2007)
17. Pilitowski, R., Szczykulski, M., Zaremba, L., Redosz, K., Derezińska, A.: FXU framework for eXecutable UML V5.0. http://galera.ii.pw.edu.pl/~adr/FXU/
18. Redosz, K.: Automatic code generation from UML models—development of the FXU tool. Master's thesis, Institute of Computer Science, Warsaw University of Technology (2009)
19. ReWalk Robotics: ReWalk—more than walking. http://www.rewalk.com/
20. Schattkowsky, T., Müller, W.: Transformation of UML state machines for direct execution. In: VL/HCC 2005, pp. 117–124. Dallas, USA (2005)

Neural Network and Kalman Filter Use for Improvement of Inertial Distance Determination

Piotr Kopniak and Marek Kaminski

Abstract Appropriate distance estimation is very important in different applications, e.g. in navigation or developing natural interfaces for man-machine interaction. Article refers to this problem and presents two approaches in improving estimation of the distance. The distance is computed on the base of linear acceleration. The acceleration data is captured by an inertial sensor mounted on moving object. The first approach uses Kalman filter and appropriate preprocessing steps to denoise measured acceleration. This method improves the distance estimation in noticeable manner but is not optimal because of time growing errors. These errors results come from the imperfection of the accelerometer and double integration of acceleration data during computational step. The second approach improves the estimation accuracy by using a neural network. The neural network estimates position of moving object on the base of statistical properties of the acceleration signal. Both of mentioned approaches were compared and the results are described in this article. Theoretical contemplation was confirmed by practical verification which results are also presented. Conducted research show that these two approaches can be combined for an optimal problem solution.

Keywords Motion capture · Distance measurement · Accelerometer · Inertial measurement unit · Kalman filter · Neural network · Inertial navigation · Man-machine interface · MEMS · IMU

1 Introduction

Inertial measurement units (IMU) are very popular sensors for many different applications. They are commonly used in various branches of research and industry. For example, smartphones use accelerometers to position determination [14], medicals

P. Kopniak (✉) · M. Kaminski
Institute of Computer Science, Lublin University of Technology, Lublin, Poland
e-mail: p.kopniak@pollub.pl

M. Kaminski
e-mail: m.kaminski@pollub.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_8

may use accelerometers to measure patient activity [17, 29], sportsmen may use them for improvement of their training [21] and animators for creating natural animation or movies [13]. An another application of inertial sensors is also measurement of vehicle work conditions and dynamics of unmanned aircrafts or cars [9, 25].

One of the most popular use of IMUs is inertial navigation. Standard navigation solutions are based on GPS system, but when the GPS signal is weak, additional position determination system is needed. Many developed solutions are based on IMUs [1, 4, 22, 27]. It means that to determine position of vehicle or human acceleration measurement is used. The acceleration data is double integrated to determine changes of subject position in time. As it was mentioned above, the most advanced navigation solutions use GPS and inertial data fusion based on extended Kalman filtering [4, 22, 23, 28]. Some of them apply neural networks or genetic algorithms to improve the inertial data processing and better position prediction [4, 20, 22, 23].

Previously conducted research on remote controlling of the robotic arm showed that inertial sensors are good solution for development of man-machine natural interfaces [12]. Research concerned remote controlling of robotic arm with Xsens MTx motion capture trackers [15] and developing Java software for this sole purpose [16]. Results of the research showed that acceleration data is full of noise and highly error prone during integration. The distance computation and linear positioning of the robotic arm was prone to drift. It was necessary to reset the robot position frequently. The noise influence for acceleration measurement is known and was widely examined [3, 4, 19, 22, 24, 27]. Kalman filtering may be used for the problem minimization, but its effect is limited. It is giving rise to additional research with use of artificial intelligence to better distance determination. Many researchers use neural network to INS/GPS navigation and accuracy improvement [4, 9, 20, 23, 28], but it is hard to find research which in straight manner compares effects of Kalman filtering and use of neural networks for improvement of the distance computing. This article replies to this absence and describes comparison of the two approaches.

2 Kalman Filtering

Estimation of the velocity and distance covered by subject of measurement is achieved by integration and double integration of acceleration data recorded by IMU (Xsens MTx sensor in this case). It results in unbounded estimation error growth over time [7, 19]. This problem may be overcome by correction feedback. Such feedback may be realized by Kalman filter.

Kalman filtering is a computational algorithm for combining noisy sensor output to estimate a state vector of the system with dynamics that needs to be modeled in the most precisely manner. The state vector includes system variables and inner variables representing time correlated noise sources [6].

Equations of process state and measurement model may be written as follows [6, 26]:

$$\boldsymbol{x}_n = \boldsymbol{A}\boldsymbol{x}_{n-1} + \boldsymbol{B}\boldsymbol{u}_n + \boldsymbol{w}_n \tag{1}$$

and

$$\boldsymbol{z}_n = \boldsymbol{H}\boldsymbol{x}_n + \boldsymbol{v}_n \ , \tag{2}$$

where: $\boldsymbol{A}$—state transition matrix, $\boldsymbol{x}$—state vector, $\boldsymbol{B}$—control input matrix, $\boldsymbol{u}$—control input vector, $\boldsymbol{w}$—process noise vector, $\boldsymbol{z}$—measurement vector, $\boldsymbol{H}$—transformation matrix, $\boldsymbol{v}$—measurement noise vector, and $\boldsymbol{n}$—number of current processing step.

If there is no control input u function component equals zero. Process noise and measurement noise are statistically independent. Matrices A, B and H can be assumed constant and equal to 1 in most cases. It means that it is necessary to estimate standard deviation of the noise functions w and v to fit Kalman filter to particular problem. The signal may be assumed as pure Gaussian because Kalman algorithm tries to coverage into correct estimation, even for poorly estimated noise parameters.

Made assumptions simplify standard filter equations. The equations in one dimensional form for a computation algorithm may be written as:

$$x_n = x_{n-1} \tag{3}$$

$$p_n = p_{n-1} + q \tag{4}$$

$$k_n = \frac{p_n}{p_n + r} \tag{5}$$

$$p_n = (1 - k_n)\, p_n \tag{6}$$

$$x_n = x_n + k_n\,(z_n - x_n) \ , \tag{7}$$

where: x—predicted state, p—error covariance, q—processing noise covariance, r—measurement noise covariance, k—Kalman gain, z—measured value.

As it was mentioned above the most important is to define appropriate noise values. They are processing noise q and measurement noise r in this case. These parameter's values for described application were defined on the base of publications [19, 27]. The measurement noise covariance r was estimated by computing variance of measured acceleration by MTx tracker in no movement conditions. The process noise was estimated experimentally on the base of initial noise value equal to $0.008\ \frac{\mathrm{m}}{\mathrm{s}^2}$ taken from individual Motion Tracker Test and Calibration Certificate attached to used MTx sensor by its producer. The chosen values for Kalman filter parameters were: process noise $q = 0.01$ and measurement noise $r = 0.00088$.

To start the filtering process loop, we need to know the estimate of x_0, and p_0. If the initial state of the system is no movement conditions the values may be set $x_0 = 0$ and $p_0 = 1$, respectively. The filtering process is done in a loop where the first step is prediction equations computing and the second one is computing corrections. Output values are input for next loop iteration.

The described filter was used to denoise the accelerometer data and improve distance computing of moving MTx tracker. The tracker was a part of man-machine natural controlling interface. The tracker was attached to a hand of the operator who was controlling a robotic arm remotely.

3 Distance Determination with Kalman Filter

The research of distance determination was done with use of MTx motion tracker from Xsens, as it was mentioned before, and obtained results and parameters of computation process presented below are valid for this type of IMU.

Distance measurement on the base of acceleration data requires double integration. A result of the first integration is a velocity. A result of the second integration is a distance. To increase accuracy of integration linear approximation between successive samples was done. The process requires additional data processing to avoid huge computation errors. The computation steps should look like that:

1. Removing Earth acceleration influence to other linear acceleration measurements. The IMU does not move in a horizontal plane so a part of Earth acceleration may be added or subtracted from accelerations measured along other axis. This influence is removed by multiplication of rotation matrix achieved from MTx and Earth acceleration vector and subtracting from measured acceleration.
2. Kalman filtering to remove signal noise. White noise may strongly change the integration results. Computed initial values of noise covariances for Kalman filter was: $q = 0.01$ and $r = 0.00088$.
3. Acceleration threshold to remove low values of acceleration. The acceleration value is not equal to zero even after Kalman filtering. Its value fluctuates below $0.1 \frac{\mathrm{m}}{\mathrm{s}^2}$. It should be zeroed because it causes of increasing velocity in no movement conditions.
4. Force the velocity down to zero if acceleration is equal to zero. This is done because the computed velocity is not equal to zero even the acceleration is equal to zero and the sensor does not move. This action was performed after 20 samples of acceleration equal to zero.

An example of a positive influence of Kalman filtering is shown at Fig. 1. The measurements presented at Fig. 1 are achieved after two pairs of moves. Each pair included one move forward and one move backward at distance 0.5 m. As we can see, the filter caused better velocity cut off when acceleration is 0 and distance determination with small drift, but more accurate than in no filtering case where the error is huge and growing fast with time. To approve the distance computation a neural network was used in the next step of the research.

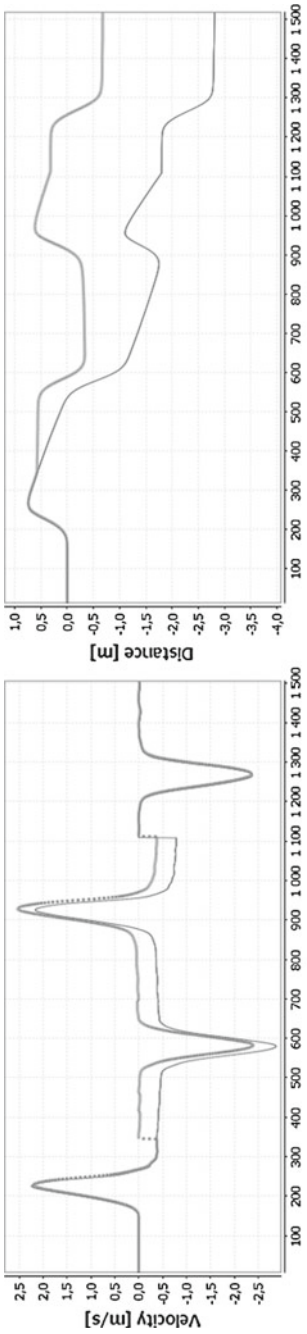


Fig. 1 Influence of Kalman filtering to improvement of velocity and distance determination. Bold data series are computed values after Kalman filtering of acceleration data

4 Neural Networks

A neural networks can be used to determine unknown processing function. It can be done by supplying network with a sufficient data to its input and usually telling what output is expected and that technique is used in current study. The name of this technique is described as training with supervisor, as the input and output data is known from beginning, which is described in following part of paper. Method is used, where finding a connection between input and output is somehow difficult and it needs complex analytic approach [5, 8, 10]. Training without supervisor is also possible, but in the current study, supervised training is much more efficient due to type of data.

A working process of artificial neural networks can be simply described as a operation that convert input space into output space by certain function, so logical representation is created from three logical layers: input, processing and output, but usually network is described as a three physical layers: input layer, hidden layer and output layer (where hidden layer can be created by many more sublayers) [8, 10, 18].

Data used to supply input layer during training process is called pattern, so doing a training process implies collection of sufficient amount of patterns [8, 10]. One of the most popular method of training technique is back propagation algorithm [2, 11, 18] and in short, the idea of this algorithm is about supplying network with pattern and then, reading response of trained network. Response value or values are compared with desired output value and overall error is calculated, which is propagated back to all neurons in network. In order to do that, activation function must have a derivative. The end of training is usually chosen by programmer, when overall error is sufficient [11]. One of the main problem of this approach is the way to determine, when error is sufficient enough and when chosen structure matches designer problem. Large error means, that due to topology, network can not find solution that can solve stated problem or more training time is still needed—overall, network can not be used in that state as response seems like random value [11]. Second problem is present, when calculated error is too small and system does not give any way of finding out, if network learned everything exactly in one to one pattern, due to large amount of memory. Good design of network use as low as can neurons to find a sufficient solution to stated problem [11].

The specific algorithms and methods that helped to find a solution to problem stated in the title of this paper are described in the following sections.

5 Distance Determination with Neural Network

In order to create working network that can solve stated problem, many different criteria were studied and one specific model was chosen. In chosen model of network there are 21 inputs, 3 hidden layers and only 1 output, which calculates distance value in response to input data.

Firstly, explanation of chosen input data is needed. During conducted research, many individual samples has been taken, and each of them describes one specific move or many different moves in one complete sample saved to file. One of the first thing to think is to supply neural network with each of saved files and wait for calculated response of the network. There are many problems in that solution and many problems about choosing right topology. One of them is presence of many samples in one file—each of file contains samples of move from few to dozen of seconds and due to registering data in 150 Hz, even short move is described by one thousand samples, which is not a good thing for neural network in terms of requirements of calculating power and therefore time. Much more simplified solution is needed. Second problem is a way of telling the network, how the response should look like, and without proper way of telling that to the network, a process of learning is even more difficult. Going further, structure of network is also going to be chosen, and without proper input and output values, created network will have large problems in matching inner topology to the class of the problem.

To ensure that created network can work with any data coming from inertial sensors, many actions have been taken. First thing to do is about reducing number of inputs to the minimum as long, as there is possibility for network to tell from them, what kind of move user has done and what is approximate distance of that move. In order to do that, 21 inputs have been chosen. First one is average of all samples values of accelerations read from file, so it can tell network in which way the move was done. Parameter also tells about approximate distance of movement, but as Kalman filter section shows, it is not usually precise representation. Second parameter is average of positives accelerations and third parameter is average of negatives. There are used to distinct movement further. Forth parameter is time of the movement in minutes, so network can tell if movement is rapid or slow-paced (which is crucial, when it comes to drift which is generated from accelerometer). Fifth one is about how many times there is move from positive acceleration to negative. It is also can help to distinct shaky movements, where data received from accelerometer contains more bias about it real acceleration. Next sixteen parameters contain data about percentage of distribution of acceleration values during study. One of the purpose of that fragmentation is to show trained network hints to determine, where the drift of the data is going to be bigger and when it is going to be smaller. Much bigger purpose is about creating system which is unvulnerable to existing data noise. By using described above parameters, impact of noise is highly reduced, mostly due to use of average value of received samples. As being said, first three parameters using approach which depends on average. Fourth parameter does not need a further filtering. When gathering information from fifth parameter, necessary cutoff level has been set to ensure, that noise is not counted as significant value. Last sixteen parameters of percentage of acceleration distribution are by the nature average values as first parameters, so they are not affected by bias occurrence. In short, this section describes chosen model for the input of the network, but there is still need to choose appropriate model for hidden layer.

Before going further, there is also need for explaining format of registered data, that for conducted study can be limited to values of one of the axis of used

accelerometer. As mentioned before, used network model required either input and output to start training process. The way of receiving missing output value is possible due to additional information that is stored in file and tells, how far the device moved from point of origin during all registered moves. That informational is crucial to create a training patterns. After setting the input layer, there is a need to choose number of neurons in hidden layer, which is not usually easy task to do, due to many circumstances. In order to find out topology that can solve the stated problem, the dedicated algorithm is needed. Implemented algorithm get all registered patterns and divide them into two groups: one of them is group of patterns for input of the network, second one is group of patterns used to verify network outcome during training process. Ratio between two groups depends on number of registered patterns, but usually 80–90 % for training data and 10–20 % for verification is enough to train network. Algorithm starts without hidden layers and gradually increases amount of layers and neurons within. In each iteration, overall error of network is calculated (which is based on input patterns) and second error calculated upon verification data. That kind of approach can help with choosing topology, where calculated error and number of neurons are as small as possible.

Created algorithm was used to choose inner structure of network, that was set as two hidden layers and each of them contains ten neurons. Final model can be described as 21-20-20-20-1 model, where are 82 neurons and 1240 connections between them and this model is used to calculate distance of new registered patterns.

6 Comparison of Kalaman Filtering and Neural Network Approaches

As conducted study shows, using neural network to solve distance calculation problem give acceptable results only in certain circumstances, which can lead to the conclusion, that network still needs more various data about moving. Figure 2a shows calculated distance by double integration with and without use of Kalman filter. Measured distance in this case is about 0.0 m, so distance line should eventually reset to starting position however, there is a drift value that finally leads to positive value at the end of move and it is about 0.18 m for Kalman filter method and 0.34 m for no filtering approach. One of the primary use of neural network is its generalization process, which is used to calculate distance value over time. In current study, network recognizes oscillating movement and assumes that it will end in certain value, which in this case is about 1 mm above real value, which is precise result.

In another case presented in Fig. 2b, network once again predicted outcome in a higher resolution that standard approach. Real distance was −0.40 m, but computed value with use of Kalman filter was equal to −0.35 m and without filter it was about −0.31 m. Due to previous training method, the network could also determine this case and returned value of distance which was −0.395 m, which differed from expected value in measurement error margin.

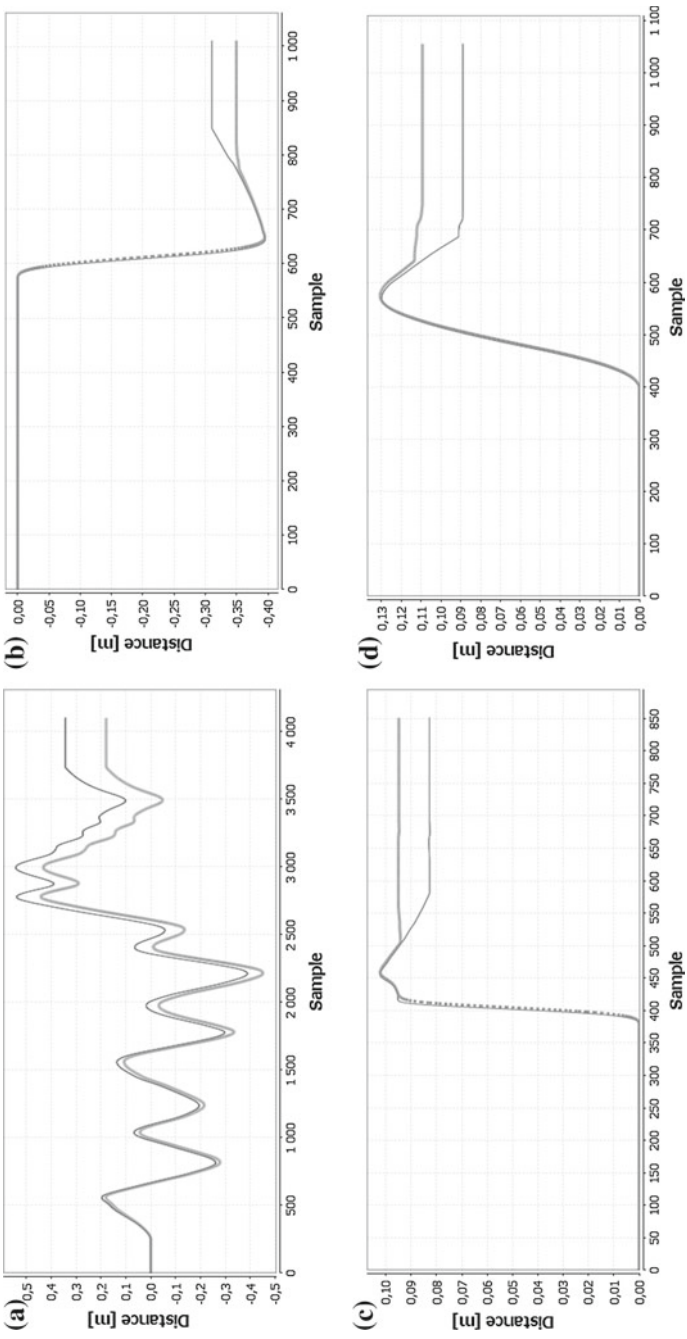


Fig. 2 Distance computed on the base of motion tracker acceleration data by double integration with Kalman filtering (*bold lines*) and without filtering (*thin lines*); **a** for real final distance equal to 0.0 m, **b** final distance −0.40 m, **c** final distance 0.10 m, **d** final distance 0.15 m

In previous cases, the movement that was made, was relatively dynamic in terms of acceleration gain, which can lead to conclusion, that growing bias is in pair with higher acceleration values. The next case covered relatively low dynamic of movement and it is shown in Fig. 2c. With lower acceleration gain, the outcome bias was also much lower than in previous measurements. Expected value in this case was about 0.10 m and once again, Kalman filter helped with correcting measure bias over time. Value gained from output of neural network was also close to expected value and the difference value was about 0.002 m.

Last case covered movement, where neural network had problems with calculating relatively simple, one directional move. Answer of the network was about 0.2 m, where expected value was about 0.15 m. In that case, also standard approaches had a problem with correct computation of the value (which suggests low precision of conducted measurement), but by using Kalman filter, error was still smaller, than network error. That movement is presented in Fig. 2d.

For many other cases, there are times, where neural network returns closer values to expected value and times where filtering data with Kalman filter gives better results. Overall, using neural network approach still needs additional development, but for data recorded during conducted research it gave sufficient results.

7 Conclusions and Future Works

The article presents two approaches of distance determination with use of inertial measurement unit. The acceleration data is not precise because of accelerometer bias, white noise and other error types. It causes incorrect determination of a distance covered by subject and the computing process should be additionally supported. There are at least two methods to resolve this problem. They are: Kalman filtering and use of artificial intelligence. Kalman filtering is a method sufficient to improve measurements of single or double moves. When the number of moves grows the measurement error grows, too. In these cases better results are achieved from a neural network.

As study shows, the preferable way of using these two approaches can be done by combining result of each other, which can be case in further study. Trained network still can be used as a tool, which will help in certain cases. It is not sufficient enough to replace analytical approach to calculate distance in few example cases and it will require additional study about which topology suits problem better and is it possible to completely replace standard way of calculating distance.

The distance covered by MTx sensor was measured with use of a metal measure tape. It is not so precise method. The measurement accuracy may be raised in future research. It is planned to use the distance measurement with game controller Razer Hydra which returns more accurate data of 3D localization.

References

1. Bebek, O., Suster, M., Rajgopal, S., Fu, M.J., Huang, X., Cavusoglu, M.C., Young, D.J., Mehregany, M., Den Bogert, V., Ton, A.J., et al.: Personal navigation via shoe mounted inertial measurement units. In: IROS 2010. pp. 1052–1058. Taipei, Taiwan (2010)
2. Bourg, D.M., Seemann, G.: AI for Game Developers. O'Reilly, Sebastopol (2004)
3. Chang, H., Xue, L., Jiang, C., Kraft, M., Yuan, W.: Combining numerous uncorrelated MEMS gyroscopes for accuracy improvement based on an optimal Kalman filter. IEEE Trans. Instrum. Meas. **61**(11), 3084–3093 (2012)
4. Chiang, K.W., Chang, H.W., Li, C.Y., Huang, Y.W.: An artificial neural network embedded position and orientation determination algorithm for low cost MEMS INS/GPS integrated sensors. Sensors **9**(4), 2586–2610 (2009)
5. Coppin, B.: Artificial Intelligence Illuminated. Jones and Bartlett Publishers, London (2004)
6. Grewal, M.S., Andrews, A.P.: Kalman Filtering: Theory and Practice Using Matlab. John Wiley and Sons, New York, Toronto (2001)
7. Guerrier, S.: Integration of skew-redundant MEMS-IMU with GPS for improved navigation performance. Master's thesis, Ecole Polytechnique Federale de Lausanne (2008)
8. Gurney, K.: An Introduction to Neural Networks. UCL Press, London, New York (1997)
9. Gwak, M., Jo, K., Sunwoo, M.: Neural-network multiple models filter (NMM)-based position estimation system for autonomous vehicle. Int. J. Automot. Technol. Manage. **14**(2), 265–274 (2013)
10. Haykin, S.: Neural Networks–A Comprehensive Foundation. Pearson Education, New Delhi (1999)
11. Heaton, J.: Programming Neural Networks with Encog 2 in Java. Heaton Research, Chesterfield (2010)
12. Kaminski, M., Kopniak, P., Zyla, K.: Zdalne sterowanie ramieniem robota z wykorzystaniem inercyjnych czujnikow rejestracji ruchu. Logistyka **6**, 5168–5177 (2014)
13. Kitagawa, M., Windsor, B.: MoCap for Artists. Workflow and Techniques for Motion Capture. Elsevier, Burlington, USA (2008)
14. Kopniak, P.: Interfejsy programistyczne akcelerometrow dla urzadzen mobilnych typu Smartphone. Pomiary Automatyka Kontrola **12**, 1477–1479 (2011)
15. Kopniak, P.: Budowa, zasada, dzialania i zastosowania systemu rejestracji ruchu firmy xsens. Logistyka **3**, 3049–3058 (2014)
16. Kopniak, P.: Java wrapper for xsens motion capture system sdk. In: HSI 2014, pp. 106–111. Costa da Caparica, Portugal (2014)
17. Lebel, K., Boissy, P., Hamel, M., Duval, C.: Inertial measures of motion for clinical biomechanics: comparative assessment of accuracy under controlled conditions-effect of velocity. PloS one **8**(11), e79945 (2013)
18. Mehrotra, K., Mohan, K.C., Ranka, S.: Elements of Artificial Neural Networks. A Bradford Book, Cambridge (1996)
19. Munoz Diaz, E., Heirich, O., Khider, M., Robertson, P.: Optimal sampling frequency and bias error modeling for foot-mounted imus. In: IPIN 2013, pp. 1–9. Montbeliard-Belfort, France (2013)
20. Nguyen, H.M., Zhou, C.: Improving GPS/INS integration through neural networks. J. Telecommun. **2**(2), 1–6 (2010)
21. Pellegrini, A., Tonino, P., Paladini, P., Cutti, A., Ceccarelli, F., Porcellini, G.: Motion analysis assessment of alterations in the scapulohumeral rhythm after throwing in baseball pitchers. Musculoskelet. Surg. **97**(1), 9–13 (2013)
22. Reinstein, M.: Use of adaptive filtering methods in inertial navigation systems. Ph.D. thesis, Czech Technical University in Prague (2010)
23. Saadeddin, K., Abdel-Hafez, M.F., Jaradat, M.A., Jarrah, M.A.: Optimization of intelligent approach for low-cost INS/GPS navigation system. J. Intell. Rob. Syst. **73**(1–4), 325–348 (2014)

24. Shi, G.X., Yang, S.X., Su, Z.: Random drift suppression method of MEMS gyro using federated Kalman filter. In: ICACC 2011, pp. 274–277. Harbin, China (2011)
25. Vorsmann, P., Kaschwich, C., Kruger, T., Schnetter, P., Wilkens, C.S.: MEMS based inte-grated navigation systems for adaptive flight control of unmanned aircraft–state of the art and future developments. Gyroscopy Navig. **3**(4), 235–244 (2012)
26. Welch, G., Bishop, G.: An Introduction to the Kalman Filter. University of North Carolina at Chapel Hill, Department of Computer Science, Los Angeles (2001)
27. Woodman, J.O.: An Introduction to inertial navigation. Tech. Rep. UCAM-CL-TR-696, University of Cambridge Computer Laboratory (2007)
28. Xia, L., Wang, J., Yan, G.: RBFNN aided extended Kalman filter for MEMS AHRS/GPS. In: ICESS 2009, pp. 559–564. Zhejiang, China (2009)
29. Zhou, H., Hu, H.: Human motion tracking for rehabilitation-a survey. Biomed. Signal Process. Control **3**(1), 1–18 (2008)

Mobile Activity Plan Applications for Behavioral Therapy of Autistic Children

Agnieszka Landowska and Michal Smiatacz

Abstract This paper concerns technological support for behavioral therapy of autistic children. A family of tools dedicated for mobile devices is presented, that allows to design and perform an activity schedule in behavioral therapy. The paper describes the main challenges, that were encountered, especially in meeting the non-functional requirements of the application: simplicity, repeatability, robustness, personalization, re-usability and learning support.

Keywords Autism · Behavioural therapy · Supportive technologies · Mobile technologies · Activity schedule method

1 Introduction

Autism is a developmental disorder, that influences the ability to socialize, communicate as well as learning skills [1, 9]. Autistic children have diverse level of deficits in language understanding, speaking and other areas, that make the therapy and education very difficult [8]. Autism is not only a personal or family matter, but a social problem, as the number of children diagnosed with autism rises all over the world. Our motivations to develop e-technologies supporting autistic children lies in a belief, that an appropriate therapy adjusted to the deficits of an individual, may result in independent life as an adult, including not only daily routines, but also finding a job. Moreover, late diagnosis as well as insufficient or inappropriate therapy may result in becoming an adult dependent on the help of the others with medical and social care financed from public money. Therefore a reliable diagnosis and effective therapy are crucial for solving the social problem of autistics' exclusion from society.

A. Landowska (✉) · M. Smiatacz
Gdansk University of Technology, Faculty of Electronics, Telecommunications and Informatics, Gdansk, Poland
e-mail: nailie@eti.pg.gda.pl

M. Smiatacz
e-mail: msmiatacz@gmail.com

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_9

There are several premises for supporting autistics' therapy with e-technologies. Most of the autistic children are eagerly using computers and tablets once they are taught how to use them [8]. Autistics require repetitive environment for functioning and learning. Technologies are able to perform the same activities in exactly the same way and with indefinite patience. Moreover, systems and devices might be customized in order to adjust to a unique set of deficits of an individual.

Some studies report, that tablets and dedicated applications are used in therapy in half of the centers for autism therapy [5, 6]. Mobile devices have several advantages over traditional personal computers: it' easier to learn to operate a touch screen than a mouse and a tablet can be taken home or to school to support an autistic child in the real environment. E-technologies, and especially mobile solutions, can be used outside specialized schools, are able to support parents in education of their autistic children as well as enable autistic individuals to have an independent life in the society.

In this paper, we describe a design of a set of tools, that support behavioural therapy, based on the activity plan method. The applications are dedicated for mobile devices (tablets). The paper describes, how the basic structure of an activity plan must be multiplied and extended in order to fulfil the following non-functional requirements: simplicity, repeatability, robustness, personalization, re-usability and learning support.

2 Background

Educating children with autism spectrum disorder could be a challenge in the best of circumstances [10]. Autism is a developmental disorder that affects mainly social and communication skills, and those deficits influence educational processes. Autistics usually aren't able to respond appropriately to their environment and learn from other people by imitation. Children with autism often have delays in language skills, including not only reading, but also speech. Some develop the ability to read simultaneously or even earlier, than the ability to speak.

Autistic children often do not perform acceptable forms of spontaneous play [8]. Instead, they use toys for knocking, putting into mouth or even throwing. Therapists as well as parents are faced with the challenge of making a child busy every minute all day long, as an autistic child often requires external support or at least initiative in performing acceptable activities [10].

Behavioural therapy proved to be successful in education of autistic children, especially when introduced in the first years of life [8]. Individualized early therapy and treatment may result in (almost) normal functioning of some people with autism in adulthood.

The activity plan method of behavioural therapy was proposed by Lynn McClannahan and Patricia J. Krantz in 1998 [7]. It is a method of teaching as well as a way of organizing self-service activities and is widely used across the world in therapy and education of autistic children. An activity plan is a detailed step-by-step instruction of performing a specific task. The instruction might be visual (photos, pictures) as well as verbal (words, sentences) and in practical settings those two types of information are usually combined to additionally develop reading skills. The tasks might include: self-service activities such as hand washing or dressing up, development of basic skills like drawing, speaking, up to social interaction and scholar challenges. The types of tasks and the mode of instructions in a specific plan depend on the development stage of a child and an individual set of deficits, as the activity plans must be highly personalized to be effective [7].

In 2014 a voluntary project has started at Gdansk University of Technology (GUT), that aimed at implementation of an application, that supports a therapy with activity plan methods [3]. The project is held with the support of Institute for Child Development (ICD) from Gdansk, Poland, which uses the activity plan method on a daily basis in children therapy. The institute runs a kindergarten for autistic children, that accepts children aged 3–8, diagnosed with autism spectrum disorders. Therapists prepare paper and binder versions of activity plans, that are designed for an individual child. Each child has tens of activity plans (and binders to guide them). Paper and binder versions of plans are effective, but inefficient due to the following reasons:

- their preparation is labor-intensive and therapists spend hours modifying them for the next use,
- their re-usability is limited,
- their storage is space consuming (large shelves of binders for each child),
- it's hard to take them out (a child can carry one binder, however more would not be handy).

For a couple of years therapists in the Institute use computers as well as tablets for therapy support. Each child educated in the institute has a tablet assigned, however only some of the activities are performed with the use of e-technologies. The Institute had tried to adapt some applications for activity plans support (a shopping list program as well as photo gallery), however, they had faced the following challenges:

- modification of the plan or activity was almost as labor-consuming as with the paper version,
- visualization of child's plan execution was often modified by mistake,
- there was no tracking of the progress with the plan nor activity,
- the program was sometimes unintentionally escaped by a child and some information was lost,
- the programs were hard to configure and contained irremovable distractors.

Due to the reasons mentioned above, a dedicated application would be a much better solution and the joint initiative of GUT and ICD aimed at developing one. The project goal is a dedicated personal activity plan, designed by a therapist on child's tablet to be used in the class as well as taken-away home and for holiday. Students of GUT

develop the activity plan application within a group project under the supervision of the paper author. The intention is not only to provide an executable version to ICD and other parties, but also to share the developed application under an open source license in order to enable extending and adjusting it by any willing institution or individual.

3 Requirements and Evaluation Criteria

The main requirement for the application is to prepare, store and display an activity schedule prepared by a therapist for a child. The basic model of an activity schedule contains the following objects: activity plan, activity and step.

An **activity plan** is a sequence of activities that a child performs during therapy, designed for one or a couple of days. The plan is always prepared by a therapist and modified with the progress of a child, sometimes day-by-day.

An **activity** is a task to be performed by a child (a task can be a simple action or more complex task, that requires additional instructions). During the learning stage the activity is followed with a script (step-by-step instruction of performing the task). The skills, that are addressed by activities might include: self-service, receptive speech, active speech, reading, writing, imitation and recognition, matching, artistic skills, mathematics, physical activities (gymnastics), group activities and peer interaction as well as reduction of inappropriate behavior.

An **activity step** is a single and simple unit action performed in order to accomplish a task (an activity). In the learning stage every activity step is supported with hints (one step might have: visual, textual and/or audio hint). With child progress the hints are gradually reduced and in the final stage a child is expected to perform a specific task without a hint or a therapist intervention. Autistic people are usually good at visual perception, therefore the visual hint is the most frequently used and then gradually replaced with others.

The functional requirements seemed simple at the beginning of the project—to edit and play slide show with sound. However, non-functional requirements and specificity of the user (i.e. autistic child) made the project challenging. The additional criteria included: simplicity, repeatability, robustness, personalization, re-usability and learning support. The criteria were explicitly or inexplicitly expressed by therapists from ICD and are a result of years of practice in behavioral therapy of autistic children.

Simplicity of application interface and interaction patterns is a basic requirement for autistic children. When provided with an application containing distractors (animations, buttons etc.), children might fix on triggering them and be unable to perform, what they are supposed to do.

Repeatability feature has at lest two meanings in this project: (1) it should be possible to repeat the same activity in exactly the same way again and again in order to support learning, but also to create habits like hand washing; (2) all activities in the plan should be handled in exactly the same way—the simpler the interface interaction

pattern is, the quicker the child would learn to use it independently. The repeatability of interaction pattern is critical due to two deficits, that autistics typically have: limited ability (or even inability) to make choices and lack of spontaneous exploration of something they do not know.

Personalization is one of the most important features, as it influences directly the effect of the therapy. With respect to the application it requires the ability to design plan, activities and steps individually for every child. On the other side, personalization of paper versions of activity plans is the reason for low re-usability and as a result, work-consuming preparation. Another expected feature, that is partially related to personalization is **learning support**, that represents ability to re-design and adjust the plans with child's progress.

Re-usability of activity plans was claimed to be a very important requirement for the designed application. Personalization and re-usability might seem in contradiction, but versioning and multiplication is easy with the use of technology. Both criteria significantly influenced the database design.

Another required feature of the designed application is **robustness**, and that requirement has significantly influenced the design decisions. Robustness in this case means, that the application is tolerant for human errors and the ones are reversible. An autistic child, especially in the early stage of therapy, does not follow typical paths he/she is expected to. Moreover, some children have associated manual disorders that make them very demanding tablet application users, as their tap and swipe gestures could be imprecise. They are expected to click anywhere, perform multi-touch and mark the activity as completed even if it is not. The specific condition of autistic children requires, that the child's plan might be changed only by therapists and any action done by mistake should be reversible by the supervising therapist.

There were priorities assigned to the non-functional requirements according to the interviews with therapists from ICD and the following order was accepted (from the highest to the lowest priority): robustness, personalization, simplicity, repeatability, re-usability and learning support. The priorities significantly influenced the architectural and design decisions for implementation as well as the project development process.

4 Application Design

The basic structure of the activity plan can be represented as a simple triad of entities: *Activity Plan*, *Activity* and *Activity Step*, as presented in Fig. 1.

Each *Activity plan* is described with a plan name, plan type and plan level. Each *Activity* is described with a skill, that is addressed, and a level of the skill, as well as with a name and type. An *Activity Step* is described with a sequence number indicating the order of steps, step name, set of hints (picture, text, sound) as well as success and failure actions (that might be a picture or a sound).

The presented basic triad of entities is not enough to implement non-functional requirements for a dedicated application supporting the activity plan method.

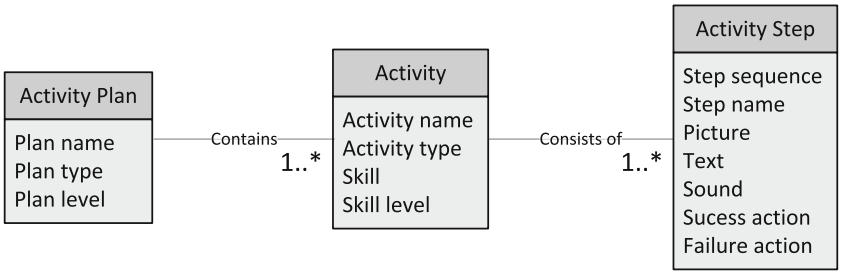


Fig. 1 Basic triad of entities for modeling activity plans

As the first priority was to provide robustness, the decision was made to divide the activity plan support to three applications: Activity Plan Player (dedicated for a child using it and robust for unintentional changes), Activity Plan Designer (dedicated for therapists) and Activity Plan Share (for therapist). There were other technical solutions, that were considered, including client-server architecture, however, due to the specific environment an application might work in (e.g. no Internet access or no server access while taken outside the institution), the decision on functionality split into the three applications have been made. The family of Activity Plan tools is presented in Fig. 2.

The aforementioned requirements raised also the database design complexity and as a result, the basic triad of entities presented in Fig. 1. was not enough. The final database model contained twelve entities, that were arranged into three layers: template layer, plan layer and monitoring layer.

The template layer was introduced in order to provide reusability of activity plans. The idea is to share templates between therapists, and even between supporting organizations in order to make activity plan preparations less work-consuming. The most important is sharing the Activity template together with prepared multimedia for each Activity Step. For some activities also Activity Versions with information on the Selection of steps might be shared.

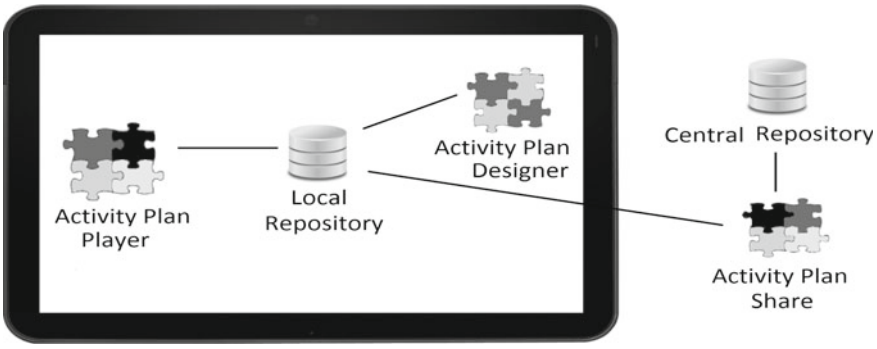


Fig. 2 Family of activity plan tools

The middle layer called Plan layer is the heart of the activity plan system, as it stores daily or weekly Activity plans prepared for a certain child therapy and education. Versatile activity plans of good quality might be saved as templates. The plans for a child are prepared by the therapists, however, they might be used outside the educational or therapeutic institution (they may be taken home and supported by parents). The direct user of the plan layer is the child, who performs the assigned activities step by step. The plan is prepared for the specific Child and the activities assigned might use templates from the template layer. Any Activity Assigned is combined with information on necessary Step adjustments, that are specific for deficits, the child has.

The monitoring layer contains data structures representing execution of the assigned plans, activities and tasks. Please note, that the basic triad of entities Activity Plan-Activity-Activity Step is multiplied in each layer in order to provide both reusability and personalization. This multiplication of the basic data structures might be typical for template-plan-do applications for any human activity. The final database

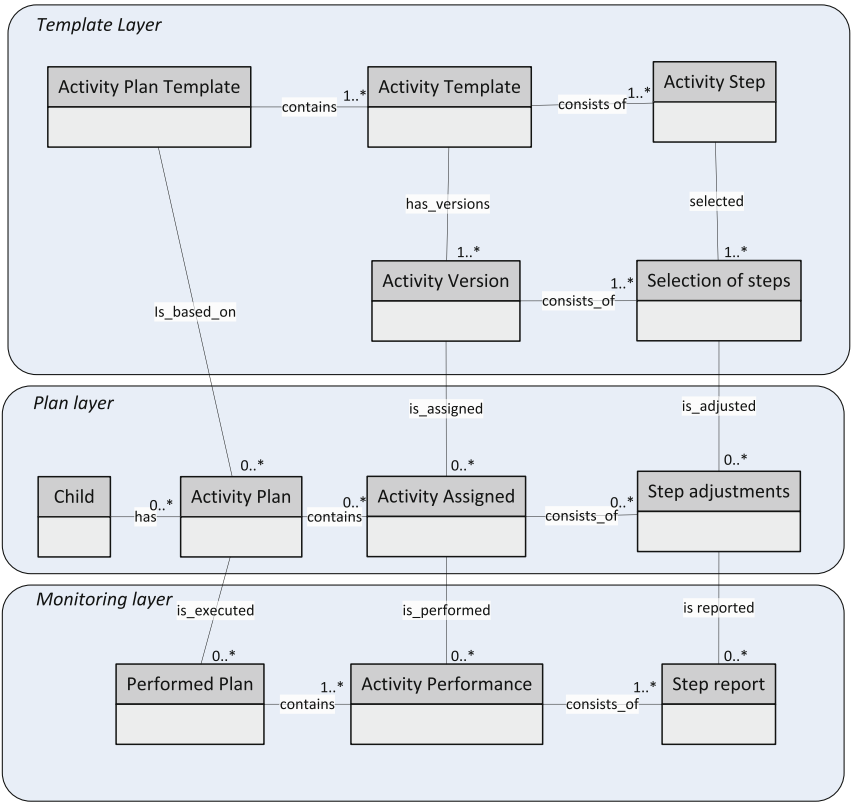


Fig. 3 Layered model of database for Activity Plan tools

design is visualized with entity-relationship diagram in Fig. 3. Please note, that for readability the attributes were removed (they have been already represented in Fig. 1).

The decision on application split influenced database implementation, as some data structures must have been shared between applications. As a result template layer and plan layer are assigned to Activity Plan Designer, plan layer and monitoring layer are assigned to Activity Plan Player, while Activity Plan Share makes copies of template layer between local and remote central repository. Please note, that layers are shared among the three applications. Synchronization of the data between applications is based on files and that decision was a result of the specific environment an application might work in (i.e. no Internet access).

5 Implementation of the Solution

The application is dedicated for Android operating system. The database is implemented with SQLite 3.7.2 and application in Java with Android SDK (version 7+). The state of the development is provided as follows: Activity Plan Player is implemented, and validated in the ICD, while Activity Plan Designer and Share applications are fully implemented, but not yet validated with therapists. Additionally in order to reduce the development time and improve efficiency, a decision was made to intentionally denormalize the template layer and now each version of an activity is stored as a separate Activity. This decision was accepted by therapists, however in future versions of applications the distinction of Activity Template and Activity Version will be reconsidered.

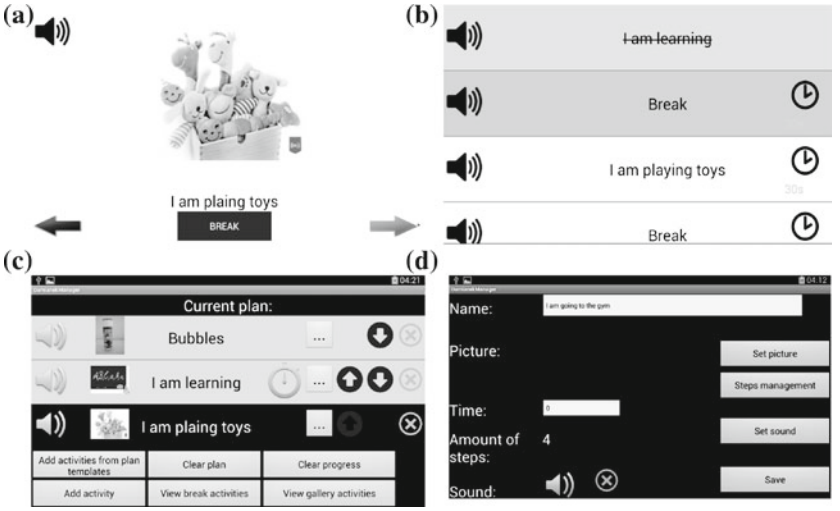


Fig. 4 Selected screenshots of Activity Plan applications: **a** Player—single activity step with sound. **b** Player—child plan view. **c** Designer—plan editor. **d** Designer—activity editor

Selected screenshots of Activity Plan applications are provided in Fig. 4. Two different views of an Activity Plan Player were shown: single activity step view (Fig. 4a) and child plan view (Fig. 4b). The list view is the basic one and activities already performed are crossed. An activity or a step can be assigned with audio instruction (denoted with speaker icon). The other two screenshots present Activity Plan Designer—edition of a plan (Fig. 4c) and configuration of an activity (Fig. 4d).

Visual information is crucial in educational support of the younger children, who do not read and is gradually replaced with audio and textual information with therapy progress.

6 Evaluation of the Solution

The developed applications were validated in ICD with real activity schedules. The evaluation of the solution is qualitative—observational study of the interactions with applications was performed. Two children took part in the validation phase: 3-year old, non-reading, with speaking difficulties and 5-year old, with high speaking and moderate reading skills. Two therapists prepared the plan and supported the child during validation process. There were three studies performed (two with the elder child). Based on the real-time observation and video analysis the applications were evaluated against the criteria defined in Sect. 3: robustness, personalization, simplicity, repeatability, re-usability and learning support.

Robustness of the solution is high—the decision on functionality split into three dedicated applications, and especially the separation of Activity Plan Player for a child only, was the ultimate solution to address the robustness challenge. During validation cases, there were a number of times when a child performed something unwittingly, and each time the change was reversed by a therapist within seconds. There was no irreversible change occurrence.

Personalization and **Repeatability** of the solution is also high—the plans for a non-reading and reading child were very different and changes of the plans were easy with some quick modifications just before validation started. Moreover, the activity plan may contain any number of repetitions of an activity and each occurrence of the activity can be adjusted to the child's progress. Each step of the activity is fully modifiable within any occurrence of the activity in the plan.

Simplicity of interaction pattern is on acceptable level. At the beginning a child required manual support from a therapist in all three cases, however after about 15 min a child interacted properly with application on his own. Learning time of 15 min is well acceptable, however one restriction must be made—both children taking part in the validation were high-functioning and learning time might increase for low-functioning autistics. Two important observations were also made: although most children with autism are reluctant to changes, both did not opposed against changing paper and binder plan to tablet schedule. Moreover, after they learned how to use the tool, they found pleasure in it (the younger one was laughing out loud, when tapping worked).

Re-usability criterion is supported by Activity Plan Share application. This application was not validated yet. Re-usability, although partially provided, is the one that suffered most from the architectural decision of the database split into three applications and denormalization of the template layer. Although partial, the progress from the paper and binder version is significant and allows to gradually reduce the work-load of the therapy preparation phase.

Learning support is possible within the proposed solution, however it is neither controlled nor automated in any way and it mainly depends on the therapist work in Activity Plan Designer application. Moreover, the decision on denormalization of the template layer caused some reduction to possible learning support, leaving more flexibility to the therapists, however reducing direct learning support by the solution.

The priorities assigned to the solution by the therapists, especially robustness as the most important one, resulted in partial implementation of the less important criteria, especially re-usability and learning support. Implementation is always some compromise between the ideal model of the database and the constraints of the target environment as well as constraints of the development process. Next versions of the applications as well as development of the Progress Monitor is expected to significantly improve both re-usability and learning support.

7 Conclusion and Future Work

Although three observational studies is unsatisfactory evidence, the preliminary evaluation results are promising. Most importantly, the children liked the application and were willing to use it. The elder one preferred looking at tablet timer than at TV screen during "film watching" activity. More precise (quantitative) evidence should use behavioral criteria and averaged therapists opinion, however that requires a period of daily use of the applications. The applications are ready to be applied in ICD kindergarten in 2015 as well as will be shared on open license (see project web page: autyzm.eti.pg.gda.pl for more information).

Current design and development processes concentrate on adding another application dedicated for monitoring child's progress based on the activity plan performance as well as behavioral metrics of tablet use. Child's progress in the ICD and many institutions is now reported manually on large sheets of millimeter paper and any support from the application would be welcomed. The behavioral patterns, that might be useful in monitoring, include: tablet movements (from accelerometer), tablet usage patterns (tap and swipe events, including multipoint tapping), application usage patterns (navigation, decision latency, reaction to distractors) [4]. Patterns could be recorded during learning tasks on a daily basis and confronted with therapist evaluation [2].

The case study presented in this paper is a good exemplification of how the non-functional requirements might influence the design as well as of trade-offs between the requirements. Moreover, it shows, that making software applications really sup-

portive for disabled people is a challenge, as certain disabilities have special deficits, that impose significant non-functional requirements.

Acknowledgments This work is a part of "e-Technologies for autistic children" project, that is supported in part by the National Centre for Research and Development, Poland under grant no IS-2/6, as well as DS Programs of the Faculty of Electronics, Telecommunications and Informatics, Gdansk University of Technology. Authors thank Activity Plan co-developers, MSc students of ETI Faculty, Gdansk University of Technology: Krzysztof Balcerowski, Damian Tykalowski and Mateusz Truszczynski. Authors thank Institute for Child Development in Gdansk, Poland for sharing their knowledge on activity plans and providing requirements for the development of the applications.

References

1. American Psychiatric Association: Diagnostic and statistical manual of mental disorders (1980)
2. Emotions in Human-Computer Interaction Research Group: Automated therapy monitoring for children with developmental disorders of autism spectrum. http://autmon.eti.pg.gda.pl
3. Emotions in Human-Computer Interaction Research Group: E-technologies for autistic children. http://emorg.eu/autism
4. Kolakowska, A.: A review of emotion recognition methods based on keystroke dynamics and mouse movements. In: HSI 2013, pp. 548–555. Sopot, Poland (2013)
5. Landowska, A., Kołakowska, A., Anzulewicz, A., Jarmołkowicz, P., Rewera, J.: E-technologie w diagnozie i pomiarach postępów terapii dzieci z autyzmem w polsce. e-mentor **56**(4), 26–30 (2014)
6. Landowska, A., Kołakowska, A., Anzulewicz, A., Jarmołkowicz, P., Rewera, J.: E-technologie w edukacji i terapii dzieci z autyzmem w polsce. EduAkcja. Magazyn edukacji elektronicznej **8**(2), 42–48 (2014)
7. McClannahan, L.E., Krantz, P.J.: Activity Schedules for Children With Autism. Woodbine House, Bethesda, MD, USA (1999)
8. Volkmar, F.R., Paul, R., Klin, A., Cohen, D.J.: Handbook of Autism and Pervasive Developmental Disorders, Diagnosis, Development, Neurobiology, and Behavior, vol. 1. Wiley, New York (2005)
9. WHO: The icd-10 classification of mental and behavioural disorders: clinical descriptions and diagnostic guidelines. http://www.who.int/classifications/icd/en/bluebook.pdf
10. Winerman, L.: Effective education for autism: psychologists are working to help struggling schools-faced with limited budgets and increasing enrollment-educate children with autism. Monit. Psychol. **35**(11) (2004)

Usability Tests Supporting the Design of Ergonomic Graphical User Interface in Decision-Support Systems for Criminal Analysis

Aleksandra Sadowska and Kamil Piętak

Abstract Usability, defined as a measure of the ease with which a system can be learned or used, is an important aspect of developing modern applications. It is crucial not only in common-use solutions such as mobile or web applications, but also in sophisticated software dedicated to specialists. Tools supporting criminal analysis are a good example of such software. The aim of the paper is to show how usability tests can influence the process of development of a specialized software. To achieve the goal the paper presents tests of a sample graph editor for link analysis provided by LINK platform—an environment supporting criminal analysis developed at AGH-UST in Kraków. The methodology of usability tests together with results and summary about modifications in user interface are presented in corresponding chapters. Than, conclusions about influence of usability tests on development process are drawn.

Keywords User experience · Design · Graphical user interface · Software supporting criminal analysis

1 Introduction

Designing sophisticated and specialized software in comparison to regular desktop, mobile or web application may seem harder and easier in the same time. It is easier, because specialized tools are usually used by people prepared to learn a lot, before they are actually able to perform their work. People with analytical minds more willingly accept, that some goals need sacrifice in order for goal to be achieved. On the other hand, such users have higher demands and expectations about features and solutions being deployed in the software. So, usually there is a correlation between length of a learning curve and software capabilities. If application is meant to be

A. Sadowska (✉) · K. Pietak
AGH University of Science and Technology, Kraków, Poland
e-mail: asadowska@iisg.agh.edu.pl

K. Piętak
e-mail: kpietak@agh.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_10

sophisticated, learning of all the functions needs to take time. The question to be asked is do specialized applications need to obey design constraints defined by usability guidelines? The answer is obvious—if it is possible then why not? Well, usually it is possible, but it is hard to be realized in the process of continuous implementation of new features. This perspective can change by pointing how many advantages can come from conducting simple usability study. Our hypothesis is that the ergonomics of usage may be increased through well designed usability and user experience studies. Moreover such studies give us not only measurement of how ergonomic the graphical interface is, but also allow for identification of crucial problems, give clues how to fix them and help to set priorities of improvements. The possible advantages of such attitude are going to be shown in context of designing the software supporting data visualization for criminal analysis. Such solutions are built by AGH-UST for polish national security institutions such as Polish National Police, Border Guide or Government Protection Bureau. Usability tests have been performed on one of such solutions, which is LINK—an extensible, desktop application for data visualization and analysis.

2 The Basis of User-Experience Studies

Usability is a subjectively new field in the domain of men-machine interactions [14]. It is usually defined as "a measure of the ease with which a system can be learned or used, its safety, effectiveness and efficiency, and attitude of its users towards it" [4]. Besides being a measure, usability can be understood as an quality attribute of a system (accordingly to ISO/IEC 25010:2011 [10]). Usability is usually being determined through usability testing. There are many approaches to usability testing, although some of them are less and some are more informative. For example, widely used The System Usability Scale [1] lets only differentiate between usable and unusable systems. Though, besides determining how well or bad a system is, usability tests should be employed to point drawbacks and suggest improvements [11]. Such effect can be better achieved through interactive user-experience tests. The most basic procedure in the case of computer application is to invite at least one participant (but three to five is optimal), ask him to deal with the application and report his major difficulties. This process should be performed iteratively [18]. Usability tests of any system are based on observation of user performance while dealing with the system. For the purpose of this article we will focus on computer applications, but it should be remembered that principles stands for any other system.

Although, it is obvious that anyone could benefit from usability testing, it is usually somewhere far on the list of priorities. Especially in Poland, the culture of user-centered design is only starting to be visible. Many companies do not separate responsibilities for design from responsibilities for development [7]. Moreover, there are not too many places where usability tester can get complete education on this matter. And last but not least, there is a lot of myths about skills that usability tester should have in order to perform his job [11].

The major domain that realizes how much benefit can come from usability studies is web design [12]. In our opinion it is good tendency that should spread. By describing the core of usability tests' methodology and example of implementing it, we would like to show, that even as sophisticated domains as criminal intelligence can benefit from it.

2.1 Structure of Usability Studies

In details, designing usability test consists of are four major steps:

(1) priorities in system's features are addressed and tasks are designed,
(2) an archetype user is defined,
(3) the actual study is performed,
(4) results are analyzed and conclusions are drawn.

Usually, usability test need to focus on some particular feature of a software. Regarding specialized software it is especially important not to bite more than one can chew [11]. Focus of each study should be narrowed to one major functionality or even some of features of such functionality. It is reasonable to choose part of software that seems to be most troublesome to actual users. Asking few of them should work perfectly.

Having determined part of software that is going to be tested, the particular task should be designed. The task is a list of assignments that participant of study should perform. Tasks can acquire different forms, but the form is not what really matters. It can be list of things to find in application, a scenario to perform or goal to achieve. The most important thing is to design a task that is entertaining and a bit challenging (but not too much, because participants may stuck somewhere in the process).

The point of the second step is to invite for study a right person. The participant should have sufficient domain knowledge in order to perform the task. Although, it does not mean that it has to be actual user of our software. In most cases, person who does not know the software can be much more informative about intuitions that novice users have and about problems they face.

The next thing is to invite participant to quiet room supplied with computer with installed version of software that is going to be tested. Then let him to perform the prepared task, accordingly with printed instruction. The computer should be supplied with screen and voice capturing software, for what participant should agree. Participant should be encouraged to speak his thoughts aloud. After completing task, what should not take longer than an hour, he should be asked about his impressions of software and things that bothered him most. The procedure described above can be enriched with more specialized equipment such as eye-tracker.

Analyzing results of the study need some expert knowledge of usability principals, although some conclusions are so obvious that anyone can draw them. Results can be described with some quantitative measures, such as time spent on each step of task, number of clicks needed to success or number of errors committed. Nevertheless,

qualitative evaluation is the one that really matters. Each study should result with the list of things that need to be improved in order to make a software more intuitive and user friendly and to shorten time needed to learn its functionalities.

2.2 Usability Guidelines

In the literature there are mentioned many dimensions on which usability can be judged (for review see [14]). Don Norman in his book [16] focuses on six fundamental psychological concepts, such as affordances, signifiers, constraints, mappings, feedback and the conceptual model. Affordances and signifiers suggest to users what actions can he perform and how and where he can do it. Constraints prevent user from making actions that are either unwanted or that can crash the system. Mapping refers to correspondence of features and controls with its analogues in the real world. Feedback stands for information that user gets after performing any action. Norman stresses that the most important is to understand conceptual model of system that user maintains while interacting with it. As a mental model he understands set of rules that govern the system. This set of principles is often supplemented with visibility, consistency, performance effectiveness, learnability and user satisfaction [7, 14]. Visibility determines how accessible are functions of the system through looking it its interface. Consistency refers to designing interfaces to have similar operations and to use similar elements for achieving similar tasks. Effectiveness says that products should be designed to achieve a high level of productivity measured in terms of speed and error and refers to levels of user performance [15]. Learnability and related to it memorability consider ability of non-trained user to accomplish certain tasks and capacity to perform tasks after period of break. If all this goals are deployed, work with the system ought to be ergonomic.

3 Criminal Analysis Tools as an Example of Specialist Decision-Support Software

Criminal analysis is a complex process involving information gathered from various sources, such as phone billings, bank account transactions or eyewitnesses testimonies. Because of massive character of this information, it is hardly possible for it to be processed without the help of sophisticated computer systems. On the other hand, because of crucial role of a human expert in the process, the main role of such a system is to support dealing with so complex data. It means that all the information has to be transformed into a coherent and relatively simple visual form, in which key objects (suspects, events, etc.) and their interrelations can be easily spotted. The analysis process is comprised of set of operations related to importing external data, data analysis and visualization.

On the market, there is a few leading technologies such as Palantir [17], a set of IBM i2 tools (i2 Analyst's Notebook [8], iBase [9], etc.) or Sentinel Visualizer [5] most of them are complex commercial solutions, designed to be general-purpose products so that they could be easily sold to many clients in different countries. Most of them provide similar set of tools which can be categorized in the following groups[1]:

- data acquisition—data can be loaded from various files (tables, documents) or data bases,
- advanced search mechanisms,
- data visualization:
  - graph visualization—showing relations between objects usually used in link analysis together with advanced analytical methods such as SNA,
  - geospatial visualization and analysis—ability to show data on a geographical map as well as executing many various analytical operations that utilize GIS methods (eg. finding possible routes, meetings spots, searching important points nearby objects locations),
  - timeline visualization and analysis—which is showing data (usually events related to analyzed objects) on a timeline due to discovering some relations between them in time context (eg. finding a frequent sequence of events),
- complex analysis tools to explore data and discover non-obvious relations among them, eg. frequent pattern mining, social network analysis.

Among all functionalities listed above, graph visualisation seem to be the most natural way of analysing complex, relational data. It allows to combine many records into one object and to spatially organize all relations that it has with other objects. This way of visualization is suitable for all sorts of data. For example, having phone calls billing of a suspect, a criminal analyst can visualise on graph his phone number as main node and all other numbers that he contacted as related nodes. The relation between them can bear information about frequency of their contact or about combined time that they have spent on talking to each other. Moreover, visualisation on graphs can employ many other modes of perceptual cues that make complex data more transparent. Manipulation of size, color and weight of nodes and relations are just few to mention.

Another example of a software which is aimed to aid criminal analysis is LINK platform [6], developed at AGH-UST in Krakow by Forensic Software Laboratory, which authors of this article are members of. LINK is a comprehensive data analysis platform [3] that provides a set of desktop tools for integrating, processing and visualizing data, which may originate from various sources (e.g. phone billings, bank account statements, address books, etc.) [2]. It covers most of the functionalities mentioned above and is designed for Polish terms and conditions.

[1]One can notice that the above software provides much more functionalities, but the intention of authors is not to do deep overview of such tools but only to outline criminal analysis and software that is usually used in this domain

Authors of this article have been developing LINK platform for years—they took part in designing the current version and have influence on further shape of graphical interface and ergonomy of LINK. All this constitutes LINK as a good playground to show how usability tests can be performed and how they can help in the process of development specialized but ergonomic software.

Link analysis with graphs visualization is also a part of decision-support systems realized for Government Protection Bureau by AGH-UST. In this case graph editor is implemented as web component dedicated for small amounts of objects.

4 Usability Tests of Sample Criminal Analysis Software—the Assumptions, Results and Conclusions

This section presents a simple usability tests methodology and presents a case study for testing a small part of LINK environment.

4.1 The Usablity Tests Methodology

Based on the knowledge that was shortly presented in Sect. 2 a simple test methodology is proposed. It adds a phase for analyzing cruciability of founded drawbacks which serves as input to estimate priorities of improvements.

In the first stage simple tasks are prepared in shape of printed instructions. Than a few participants chosen from a target group get a running application and follow the instruction to achieve given goals. Tasks execution isn't time limited. The participants are free to explore functionalities of tested program along with their intuition, but they are asked to report their thoughts aloud. During the section a test leader observes behavior of participants and writes down remarks. Due to characteristics of study design, the descriptive type of results reporting is going to be employed. However, some auxiliary quantitative (such as time of task performance as success rate) and qualitative (as participants personal feelings and experience) variables will be invoked as well. To maximize accuracy of the variables, they are set independently by two test leaders based on the live observations and video records made during the tests.

The next stage is a debriefing session where participants give feedback, share their feelings about way for using the application, point crucial difficulties and propose some improvements. It is desired to record the session to have a huge material for further analysis.

In the third stage all remarks are analyzed to produce a list of crucial areas of tested application which require improvements. Based on the test results which allow to estimate how difficult particular functional areas are, priorities for further development are suggested. An output of the stage is a prioritized list of founded bugs,

improvement or even new features. Such output is than passed to development team for further evaluation.

4.2 Study Design for LINK Usability Tests

We designed two simple tasks, with partly overlapping instruction that covers data visualization on graph together with simple graph editing facilities. LINK was preloaded with data and ready to perform data analysis on graph diagrams. In the study participated four young, frequent computer users, novice in LINK software, without domain knowledge. Study was conducted on regular notebook, 13.3' screen size, with screen and voice capturing program (Screenpresso [13]).

4.2.1 Task 1

The task explored subject's ability to build a model of functions possible to execute, based on given interface. Participants were meant to:

(1) find how to visualize preloaded dataset (a phone call billing) on a diagram,
(2) explore tools for graph editing,
(3) determine what aspects of graph means what (nodes, connections, labels).

At figures below screenshots from LINK during execution of Task 1 are shown. Figure 1 shows interface of program before executing the task. Figure 2 shows example of desired results.

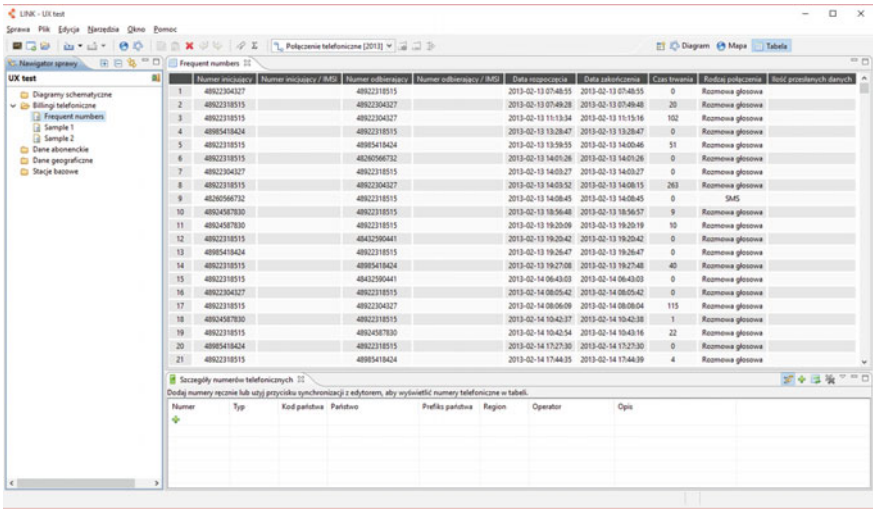


Fig. 1 A screenshot from the LINK environment which is a starting point for Task 1

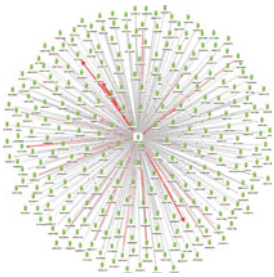


Fig. 2 An exemplary result of Task 1b

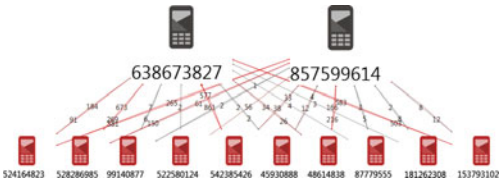


Fig. 3 An exemplary result of Task 2

4.2.2 Task 2

This task was focused on specific graph editing functionalities and had scenario-like formula. The goal was to prepare a readable diagram that may be accepted as a proof in the courtroom.

The instruction that was more specific included:

1. Visualization of two datasets on one diagram.
2. Inference from the diagram a subset of nodes common for both datasets.
3. Visual preparation of the diagram by:
   (a) Removing unimportant nodes
   (b) Adjusting the layout of nodes (by choosing transparent display)
   (c) Resize and color icons of phone numbers
   (d) Change width and color of connections accordingly to their weights
   (e) Edit labels, so that they contain crucial information
   (f) Export the results of analysis to an image

The desired result is shown in Fig. 3.

5 Test Results

The Task 1. was completed by more or less accurately all four participants, although one of them resigned after it. The Task 2. was completed by three participants. In the Table 1 there is time of executing particular parts of each task and subjective rate of success (estimated in percentage), being resultant of completion of each goal, number of errors and given hints. Average time of executing the Task 1. is around

Table 1 Results of particular goals in both tasks

Participants ID number:		1		2		3		4	
Goal to achieve:		min	%	min	%	min	%	min	%
Task 1	Finding how to visualize billing on diagram	<1	100	2	85	3	70	5	65
	Finding the most frequent contacted phone number by adjusting size and color of icons and arrows	4	85	10	60	5	70	7	10
Total time/avarage percentage		5	92.5	12	72.5	8	70	12	37.5
Task 2	Visualizing two diagrams on graph and finding common numbers	6	50	<1	100	1	100	–	0
	Erasing not important connections	2	80	7	60	6	85	–	0
	Adjusting size and color of icons and arrows	12	75	4	90	6	70	–	0
	Editing labels	3	45	4	65	4	30	–	0
	Adjusting display	1	90	<1	100	2	85	–	0
	Exporting results	<1	100	<1	100	<1	100	–	0
Total time/average percentage		25	73.3	18	85.8	22	78.3	–	0

9 min which is about 3 time more than performing the same task by authors. In the second case, average time is around 21 min and can be compared to 5–6 min that takes to do the same task by the authors. The Table 1 shows that users had especially problems with:

- finding appropriate analytical functionalities (eg. removing not important nodes in Task 2. took a lot of time and LINK provides a function which performs such operation automatically in a few seconds)
- adjusting visibility attributes (eg. editing label),
- reading important information from diagram (eg. finding the most frequent contacted phone numbers took a few minutes what mostly was caused by not-readable labels on a diagram)

Observing users allowed us not only to identify difficulties, but also gave us a lot of information how to change graphical interface—observing where people look for particular functions helps to design improvements.

In the last part of usability tests, new priorities of improvement have been set—the results summarized in Table 1 identified most crucial areas that need to be changed. This constitutes a basic set of new development tasks that can be passed to the development team.

6 Conclusions and Further Work

The main goal of the study was to present how usability tests can be utilized in design ergonomic graphical interface of specialized software such as LINK environment for criminal analysis. The research was focused on describing usability test methodology which begins from planning test areas, through preparing and performing test scenarios, interviews, results analysis and creating new issues with priorities based on identified crucial difficulties.

The sample test case performed on LINK environment proved that this methodology gives promising results—a few major improvement were proposed and many minor changes were identified.

The next step in our research is to implement proposed changes in LINK and then verify how they contribute to usability improvements. The similar tests will be performed on a new version of the environment and result (especially time of dealing with particular tasks) will be compared. Another direction of research will be comparing various users groups: experienced analysts that worked on LINK environment (to identify how they deal with common tasks and what they complain about), experienced analysts that haven't used LINK before but normally use similar tools and inexperienced users. This will be used to measure learnability of chosen functionality areas. Another direction of further research is to check how results gathered from LINK tests could be applied in other similar software such as decision-support system for Government Protection Bureau.

Acknowledgments The research reported in the paper was partially supported by grants "Advanced IT techniques supporting data processing in criminal analysis" (No. 0008/R/ID1/2011/01) and "Information management and decision support system for the Government Protection Bureau" (No. DOBR-BIO4/060/13423/2013) from the Polish National Centre for Research and Development.

References

1. Brooke, J.: SUS–a quick and dirty usability scale. Usability Eval. Indus. **189**(194), 4–7 (1996)
2. Dajda, J., Debski, R., Kisiel-Dorohinicki, M., Piętak, K.: Multi-domain data integration for criminal intelligence. In: Gruca, A., Czachórski, T., Kozielski, S. (eds.) Man-Machine Interactions 3, AISC, vol. 242, pp. 345–352. Springer, Switzerland (2013)
3. Dębski, R., Kisiel-Dorohinicki, M., Miłoś, T., Piętak, K.: Link: a decision-support system for criminal analysis. In: MCSS 2010, pp. 110–116. Kraków, Polska (2010)
4. Dix, A., Finlay, J., Abowd, G., Beale, R.: Human-computer Interaction. Prentice-Hall Inc, Upper Saddle River (1997)
5. FMS Advanced Systems Group:http://www.fmsasg.com/Products/SentinelVisualizer
6. FSLAB: https://www.fslab.agh.edu.pl/#!product/link2
7. Henderson, A.: Interaction Design: Beyond Human-Computer Interaction. Ubiquity 2002 (March) (2002)
8. IBM: http://www-03.ibm.com/software/products/en/analysts-notebook
9. IBM: http://www-03.ibm.com/software/products/en/ibase

10. ISO: http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=35733
11. Krug, S.: Don't Make Me Think: A Common Sense Approach to the Web Usability, 2nd edn. New Riders, Thousand Oaks (2005)
12. Krug, S.: Rocket Surgery Made Easy: The Do-It-Yourself Guide to Finding and Fixing Usability Problems, 1st edn. New Riders Publishing, Thousand Oaks (2009)
13. Learnpulse SAS: http://screenpresso.com
14. Lee, S.H.: Usability testing for developing effective interactive multimedia software: concepts, dimensions, and procedures. Educ. Technol. Soc **2**(2) (1999)
15. Lindgaard, G.: Usability Testing and System Evaluation–A Guide for Designing Useful Computer Systems. Chapman and Hall, Chapman and Hall computing series (1994)
16. Norman, D.A.: The Design of Everyday Things. Basic Books Inc, New York (2002)
17. Payne, J., Solomon, J., Sankar, R., McGrew, B.: Palantir: The future of analysis. In: IEEE VAST 2008, pp. 201–202. Columbus (2008)
18. Rubin, J.: Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests. Wiley, New York (1994)

Evaluating Informational Characteristics of Oculomotor System and Human–Computer Interfaces

Raimondas Zemblys

Abstract Interest in adapting eye movements to everyday interaction is apparent in recent consumer technology: more and more interfaces with gaze as input will be developed in years to come. Usually *throughput* and other Fitts' index of difficulty based methods are used to evaluate human's ability to perform a coordinated movement. However, it is questionable if such methods apply to gaze based interaction. This paper explores measures of *information transfer rate* and *channel capacity*—a better alternative to throughput. Based on difference of initial and final entropy of the extent of the movement, these measures can be used to evaluate and compare any pointing devices (mouse, joystick, eye tracker, etc.) because they evaluate only information processing capacity of oculomotor channel, without including informational characteristics of the methods to perform object selection.

Keywords Eye movements · Eye tracking · Human–computer interaction · Information theory · Fitts's law

1 Introduction

Eye tracking and eye movement based measures provide a wealth of cognitive and behavioral data suitable for real time human–computer interaction. Because of the long history of academic research using eye movements to understand human cognition, perception and attention (see Kowler [14], Martinez-Conde [11], Rayner [5] for excellent reviews), and because of the highly successful adaptation of this technology for eye control of interfaces for the physically disabled (e.g. Majaranta and Räihä [10], Purwanto et al. [13]), there is now a lot of interest in using eye movements as an input when interacting with computers, intelligent environments and mobile devices. For example, recently Denmark's The EyeTribe unveiled a $99 USB 3.0 hardware accessory which provides eye tracking capabilities to a laptop or PC running Windows 7/8 or OS X. This external tracker is mainly focused on developers,

R. Zemblys (✉)
Siauliai University, Siauliai, Lithuania
e-mail: r.zemblys@tf.su.lt

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_11

but the company is in negotiations to come out with their own tablet with built in eye-tracking software and hardware [16]. Large companies are also very interested in consumer-level eye tracking: almost 2 years old Samsung's flagman Galaxy S4 already has a very basic eye tracking features: "Smart Scroll" (which allows users to scroll while looking at the screen and tilting the phone), and "Smart Pause" (which pauses videos when the user looks away). Tobii, the global leader in eye tracking, and SteelSeries, a leading manufacturer of top quality gaming gear, launched the mass-market consumer eye tracking device for gamers.[1] Sony's Magic Lab is currently looking into gaze control technology using SMI's RED-oem eye tracker that could allow players to use eye movements to steer a game's camera, employ eye gaze as an aiming or selection mechanism, and also let the gamers have more intelligence as they'll know what the other players are looking at.[2] The companies believe that low cost eye tracking has the potential to enrich games and everyday computer interaction experience. Game developers who want to lead the innovation and be part of the first wave of eye tracking games are already using consumer-level eye trackers and already started creating new ways of interacting with computers.

1.1 Evaluation of Human–Computer Interaction

The key focus in human–computer interaction (HCI) research is prediction, simulation and evaluation of user's ability to perform actions with interactive objects in graphical user interfaces or to execute control commands for interaction with physical world objects. The purpose of this type of research is to develop more efficient and more intuitive devices and interfaces, that match natural capabilities of humans with interaction techniques on computing systems. Fitts's law is a well established method for assessment of individual's ability to perform a coordinated movement and is often used to predict time, required to hit a target of a certain size at a certain distance [17]. Based on Fitts's law, ISO 9241—part 9 standard establishes uniform guidelines and testing procedures for evaluating computer pointing devices. The metric, used to assess performance of target selection tasks is called throughput C (measured in bits/s, Eq. (1)), and includes both, the speed and accuracy of a pointing task.

$$C = \frac{ID_e}{MT}, ID_e = \log_2\left(\frac{A}{W_e} + 1\right) \tag{1}$$

Here A is target distance, MT—movement time, ID_e—effective index of difficulty, W_e—effective target size (target width, normalized to reflect what a subject actually did, rather than what was expected [9]).

[1] Tobii Press release. Tobii Declares 2015 "Year of Consumer Eye Tracking" at International CES, Dec 2014. http://goo.gl/YeYLbJ.

[2] SMI news. Eye tracking for PS4—Sony Magic Lab and SMI eye tracking, Nov 2013. http://goo.gl/5BZgBB.

However, HCI community does not agree whether Fitts's law applies to eye movements. A lot of researchers express themselves skeptical on the validity of Fitts's law for the eyes, nevertheless they use it when comparing different computer input devices or methods (see Ware and Mikaelian [18], Ashmore et al. [1]), others claim that Fitts's law can be used to evaluate gaze based interfaces (see Miniotas [12], Zhang and MacKenzie [20]).

1.2 Evaluation of Gaze Interaction

If we consider a gaze-aware interface, basics of Fitts's law suggest that given enough time subject would be able to direct his gaze within a presented stimulus of any size. However, accuracy of eye pointing is limited to the size of fovea and accuracy of gaze tracker (see Holmqvist et al. [4], p. 42). Assuming Fitts's law for eye movements also is a contradiction to a well established psychological eye speed models [8]. Carpenter's formula (Eq. 2, [2]), the most prevalent model, does not involve target size and states, that saccade (i.e. movement) duration D (or movement time) has a direct relationship with saccade amplitude A:

$$D = 2.2 * A + 21 \tag{2}$$

A very simple example experiment was conducted to demonstrate that Fitts's law does not directly apply to eye movements: participant was asked to direct her eyes from the center of the screen to the targets of 0.25, 0.5 and 1° width, displayed 5, 10 and 20° away. Velocities of the saccades, performed to the targets with equal Fitts's indexes of difficulty (equal A/W ratios), are presented in the left of Fig. 1 (traces a, b and c). According to the Fitts's law, movement times to the targets with equal A/W ratios should be the same. However, resulting durations of the saccades, as well as peak velocities, are quite different. Right part of Fig. 1 (traces c, d and e) shows

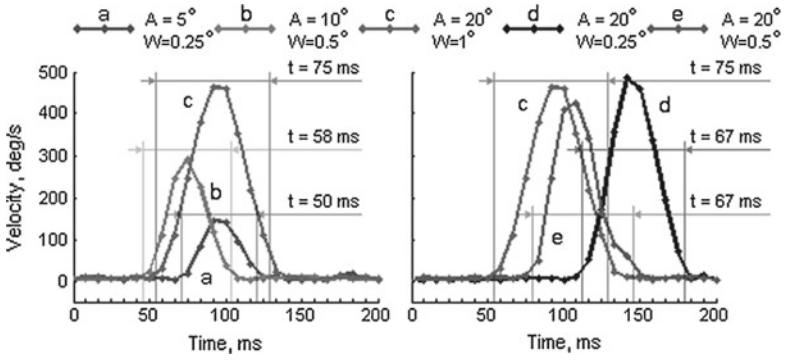


Fig. 1 Example experiment: *a*, *b*, *c*, *d* and *e*—velocity profiles of saccades when shifting gaze to the targets with different Fitts's indexes of difficulty. *A*—distance to the target, *W*—width of the target in degrees, *t*—duration of the saccade in milliseconds

velocities of the saccades, performed to the targets at the same distances but different sizes, and therefore resulting to unequal A/W ratios. Durations of the saccades are more or less the same and confirm psychological eye speed models.

Interest in adapting eye movements to everyday interaction is apparent in recent consumer technology: more and more interfaces with gaze as input will be developed in years to come. In [3] Drewes argues that recipe for a successful submission of a paper on HCI is to provide Fitts's law evaluation, but a simple example described above demonstrates that Fitts's law cannot be used to evaluate performance of gaze pointing and therefore other methods are needed to assess and compare gaze-aware interfaces and other common computer input methods. This paper addresses this problem and provides a basis for informational evaluation of gaze-aware interfaces.

2 Method

The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point.

C.E. Shannon [15]

From the very beginning Fitts attributed his law to information theory with the idea that positioning time is limited by information processing capacity of subject's central nervous system. Using information theory concepts, target tracking (or pointing) task can be described as follows: trajectory of the target in physical world or computer screen would be source (input) information of the oculomotor system, gaze trajectory while tracking this target—output information, and the difference between two trajectories could be regarded as information, lost during the transmission process [7]. If the target velocity (frequency) would increase or trajectory of the target would become more complicated, the source information rate would increase and, because of the noise in *information transfer channel* (inaccurate perception of target position, limited computational resources of the brain, performance of eye muscles), at some point, oculomotor system won't be able to accurately track target. This allows to formulate conclusion that oculomotor system has a limited *channel information capacity* and this property can be used as an important characteristic to determinate control system of the eye movements. Illustration of human's oculomotor system, modeled as Shannon's information transfer channel is presented in Fig. 2. It should be noted, that in real world, *information transfer channel* refers not only to oculomotor system, but also includes equipment to display the target and measure the movement, e.g. eye tracker (components within black thick line in Fig. 2), therefore the same concept can also be used to evaluate HCI systems.

In this study target acquisition task is modeled as information transfer process, i.e. when subject tracks a target, information of stimulus position is "transferred" to the gaze position and therefore informational characteristics of oculomotor system can be described by *R—information transfer rate* over an oculomotor channel (Eq. 3), which, in case of discrete target trajectory, is calculated as a difference of initial and

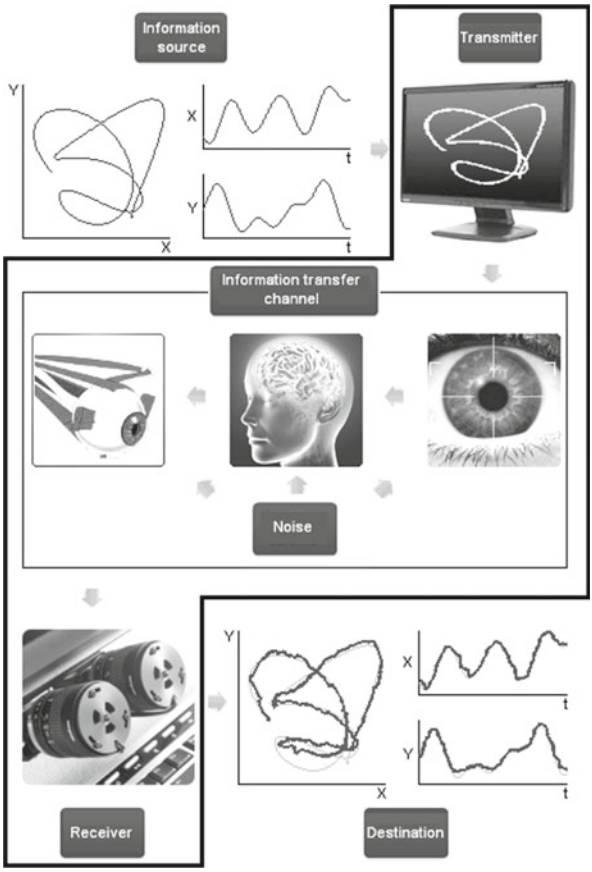


Fig. 2 Human oculomotor system as Shannon's information transfer channel. *Black thick line*—information transfer channel in practice, i.e. "real world" applications

final entropy of the extent of the movement [7]. Initial entropy would be described by N—number of uniformly distributed stimulus locations. Final entropy would be described by remaining uncertainty after gaze shift and, if end-points of the movement have normal distribution with standard deviation of σ, it would be calculated as $\log\left(\sqrt{2\pi e}\sigma\right)$. It is necessary to account for the information rate presented to the subject, which in this case is B—frequency stimulus presentation.

$$R = B \log \frac{N}{4.132\sigma} \tag{3}$$

Common practice to evaluate informational characteristics of movement control system is to apply Fitts's law based methods, therefore in this study information transfer rate over human oculomotor system is compared to its informational characteristics, based on Fitts's index of difficulty. Moreover, standardized target acquisition task is used to ensure comparability between this and other studies, where Fitts's law and methods described in ISO 9241—part 9 standard are used.

2.1 Procedures

Ten subjects—8 male and 2 female with average age of 24.5 years ($SD = 3.2$) volunteered to participate in the experiment without any compensation. All had normal or corrected to normal vision and had prior experience with eye tracking. LC Technologies Eye Gaze System was used to record eye movements of both eyes at a frequency of 60 Hz each. Subjects used a chinrest and after 9 point calibration procedure were asked to track a stimulus presented on a computer screen (17 in, 1280×1024 px) positioned at a distance of 70 cm from the eyes. Recorded data was storied in a computer memory for offline analysis using custom MATLAB scripts. System latency of approximately 42 ms was removed and, in order to improve accuracy, gaze position was recalculated using interpolation based method for approximation of gaze data from separate sensors (see [19], in Lithuanian). Fixations were detected using a dispersion based algorithm with minimum fixation duration threshold of 100 ms and dispersion threshold of 0.5°.

In this experiment discrete scheme of multidirectional target acquisition task [17] was selected, and involved the subject to repeatedly shift her gaze from the homing center to the presented stimulus of $W = [0.25; 0.5; 1]°$ width at a distance of $D = [5; 10]°$. Total of 16 stimuli for each combination (except for $W = 0.5$ and $D = 5$) were presented in time intervals of $T = [0.8; 0.6; 0.5; 0.4; 0.3; 0.2]$ s. To avoid prediction and anticipation, random order of stimuli was used and data while shifting gaze to the homing center was not analyzed. Movement *end-points* were obtained by simulating *eyeclicks* with a dwell time of 300 ms. If no such fixation appeared during the period of stimulus presentation, fixation that started after the new stimulus presentation and was closest to the stimulus was considered as end-point. This ensured that in the case of double-step saccade mode [6] only fixations after corrective saccades were marked as end-points and that no stimuli was marked as missed if subject was able to fixate on it, but because of short stimulus presentation duration, fixation duration was under the dwell time threshold. Movement times (MT) were calculated as durations between the end of last fixations that belong to the previous stimuli and a start of fixations that mark movement end-points.

3 Results

3.1 Throughput Based on Fitts's Index of Difficulty

Selected values of D and W result to Fitts's index of difficulties ID, ranging from 2.58 to 5.36 bit (see Table 1). Throughput TP was calculated as ID/MT, where MT is an average movement time. Following the procedure described in [17], effective distance D_e, effective width W_e, resulting effective index of difficulty ID_e, and effective throughput TP_e were calculated (Table 1). According to Soukoreff and MacKenzie [17] *movement time refers to the time subjects spent moving the pointing device,*

Table 1 Informational characteristics of human oculomotor system, based on Fitts's index of difficulty

D	W	ID	T	D_e	W_e	ID_e	MT	TP	TP_e	TP'_e
5	0.25	4.39	0.8	5.00	2.61	1.72	0.11	39.91	15.64	4.20
5	0.25	4.39	0.3	4.41	3.55	1.48	0.07	62.71	21.14	4.00
5	1	2.58	0.8	4.94	2.36	1.76	0.10	25.80	17.60	4.40
5	1	2.58	0.3	4.81	3.26	1.51	0.08	32.25	18.88	3.97
10	0.25	5.36	0.8	9.71	3.55	2.29	0.15	35.73	15.27	5.09
10	0.25	5.36	0.3	9.09	5.05	1.75	0.08	67.00	21.88	4.61
10	0.5	4.39	0.8	9.68	4.57	2.09	0.18	24.39	11.61	4.35
10	0.5	4.39	0.3	9.02	5.09	1.77	0.08	54.88	22.13	4.66
10	1	3.46	0.8	9.64	4.34	2.03	0.14	24.71	14.50	4.61
10	1	3.46	0.3	9.02	4.55	1.62	0.07	49.43	23.14	4.38

D, W, D_e and W_e—distance, width, effective distance and effective width of the target in degrees of visual angle; ID and ID_e—resulting Fitts's index of difficulties in bits; T—stimulus presentation time in seconds, MT—average movement time in seconds; TP, TP_e and TP'_e—throughput, effective throughput and throughput of intentional stimulus selection in bits/s

and specifically should not include homing time, dwell time, or reaction time if a discrete task is used, therefore one can notice very high TP (24.39–67 bit/s) and TP_e (11.61–22.13 bit/s) values, resulting from a very short movement times (70–180 ms).

Obtained values of TP_e and TP are very high and no way comparable to the throughput of manual pointing (3.7–4.9 bit/s, [17]), unless one considers intentional stimulus selection and adds time, required to perform an eyeclick (e.g. dwell time) to overall movement time. Dwell time of 300 ms is used to calculate *throughput of intentional stimulus selection* TP'_e (Table 1), which ranges from 3.97 to 5.09 bit/s. However, this "modification" is a contradiction to a well established guidelines of throughput calculation [17]. Also, because saccades are very fast (i.e. movement times are very short), TP'_e is mostly dependent on dwell time selected, therefore it is more likely to evaluate informational characteristics of the method, used to perform eyeclicks, but not characteristics of gaze pointing or information processing capacity of the oculomotor system. Just by selecting different dwell times, one can conclude whatever she is up to: gaze pointing has better performance than manual pointing or vice versa.

Throughput values seem to depend on stimulus presentation duration. This raises a question: which fixation describes information processing capacity of the oculomotor system better—one after initial saccade (inaccurate but resulting to the lower movement time), or one after complete movement (accurate but with higher movement time)? Throughput values in Table 1 suggest that, despite of increased effective width, higher throughput is obtained when frequency of stimulus presentation $1/T$ increase. This can be explained by a fact that MT values are lower when T is shorter, and most likely it is because there is no time to execute corrective saccades, i.e. movement time is in the left part of bimodal distribution of movement times (see Fig. 3, right).

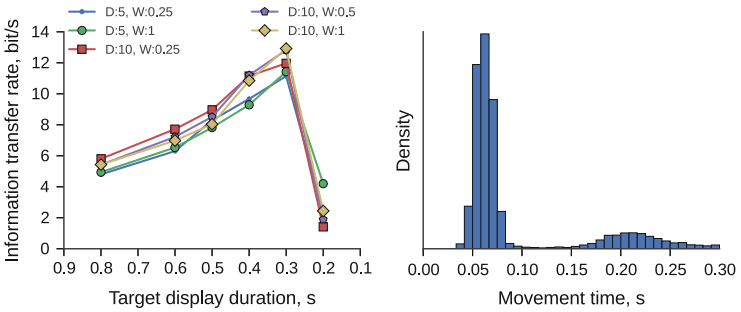


Fig. 3 *Left* Information transfer rate (in bit/s) over an oculomotor channel for different targets. *Right* bimodal distribution of movement times

3.2 Information Transfer Rate and Channel Capacity

A measure of *information transfer rate* is a better alternative to throughput, based on Fitts's index of difficulty, because it's calculation does not directly involve movement time. Results show that with the increase of input information rate (decrease of target display duration T), information transfer rate over an oculomotor channel increases until reaches its maximum of 13 bit/s at $T = 0.3$ s when $D = 10$ and $W = 1°$ of visual angle (Fig. 3, left, diamond marker). This maximum can be described as an *average channel capacity* of the oculomotor system. Maximum information transfer rates for other D and W combinations are lower and range from 11.14 to 12.82 bit/s, meaning that target size affects accuracy of gaze pointing. This is not surprising, because as the target gets smaller it is more difficult to accurately determine its spatial location using peripheral vision, especially having very limited amount of time to perform a gaze shift. This results to increased errors in information transfer channel, and, as a consequence, increased amount of lost information. This result partially confirms claims of other researchers, that gaze pointing follow Fitts's law but the reason is not linearly increased movement time to more distant targets. Because of inaccurate perception of target location, saccade might undershoot or overshoot the target and corrective saccade is needed. This sums-up to a step-wise longer movement times and indeed, distribution of movement times is bimodal (Fig. 3, right). Two separate distributions can be identified: one with the mean, corresponding to the saccade duration (around 60 ms) and therefore depending on D_e, and another with the mean of approximately 210 ms, resulting from gaze shifts performed in double-step mode [6].

Information transfer rates when performing gaze shifts to the targets at a distance of $D = 5°$ are lower comparing to $D = 10$, meaning that 1) source information rate is lower—number of possible target locations N and therefore initial entropy is lower, and 2) target appears close to homing position and therefore its spatial location is perceived more precise. However, at maximum frequency of stimulus presentation used ($T = 0.2$ s), information transfer rate is noticeably higher (4.2 bit/s, Fig. 3,

circle) when $D = 5$ and $W = 1°$, meaning that given a very limited amount of time, it is easier to direct gaze to the nearby and clearly visible targets. On the other extreme, when subjects had enough time for gaze shift ($T = 0.8$ s), obtained information transfer rates are 4.8–5.8 bit/s, which is very in line with the throughput of manual pointing (3.7–4.9 bit/s, [17]) and *throughput of intentional stimulus selection*(3.97–5.09 bit/s, Table 1).

It should be noticed that information transfer rate explicitly evaluates only information processing capacity of oculomotor channel, without including informational characteristics of the method, used to perform object selection.

4 Discussion and Conclusions

Evaluation of information processing capability of the oculomotor system, based on difference of initial and final entropy of the extent of the movement can be used to directly evaluate any pointing device. This is not possible using current approach—calculation of throughput based on Fitts's index of difficulty, unless one considers intentional stimulus selection and add time, required to perform an eyeclick (e.g. dwell time) to overall movement time. The results presented show that, in the case of eye movements, information transfer rate is a more stable measure comparing to throughput, however practical use of the described method is somewhat limited, because it does not directly predicts the time to acquire a target. Measures of *information transfer rate* and *channel capacity* can only be used when comparing performance of different input devices, and time to acquire a target with gaze pointing needs to be predicted using eye speed models [2, 8]. If one considers measuring human performance in absolute terms, she needs to account for the latency in oculomotor system and eye tracker, because in most of the cases at a highest frequency of stimulus presentation used, subjects were able to direct their gaze towards stimulus position, despite it was already gone. However, eye trackers are not perfect and introduce measurement errors, therefore in real world, *oculomotor channel* would refer to human oculomotor system together with eye tracker device. Assuming that subject performs tasks with same the performance, described method can be used to evaluate and compare different eye trackers. In this case, one should not adjust any latencies and calculate information transfer rates and channel capacities as is.

Results imply that information transfer rate over an oculomotor channel depends on the distance and width of the stimulus, however deeper analysis is needed in order to find combination of width and distance of the stimulus, where information transfer rate reaches absolute maximum, i.e. channel capacity. The experimental database was relatively small and was intended for initial assessment of the proposed measure, therefore no statistical analysis could be performed. However, in the future author plans to scale up the experimental database in order to further support the advantages of proposed measure and analyze additional parameters implicated in HCI.

References

1. Ashmore, M., Duchowski, A.T., Shoemaker, G.: Efficient eye pointing with a fisheye lens. In: GI 2005. pp. 203–210. Victoria, Canada (2005)
2. Carpenter, R.H.S.: Movements of the Eyes. Pion, London (1988)
3. Drewes, H.: Only one Fitts' law formula please! In: CHI EA 2010. pp. 2813–2822. Atlanta, USA (2010)
4. Holmqvist, K., Nystrom, M., Andersson, R., Dewhurst, R., Jarodzka, H., van de Weijer, J.: Eye Tracking: A Comprehensive Guide to Methods and Measures. Oxford University Press, Oxford (2011)
5. Kowler, E.: Eye movements: the past 25 years. Vis. Res. **51**(13), 1457–1483 (2011)
6. Laurutis, V., Zemblys, R.: Bayesian decision theory application for double-step saccades. Elektronika ir Elektrotechnika **4**(4), 99–102 (2009)
7. Laurutis, V., Zemblys, R.: Informational characteristics of the double-step saccadic eye movements. Inf. Technol. Control **39**(1), 55–60 (2010)
8. Lebedev, S., VanGelder, P., Tsui, W.H.: Square-root relations between main saccadic parameters. Investig. Ophthalmol. Vis. Sci. **37**(13), 2750–2758 (1996)
9. MacKenzie, I.S.: Fitts' law as a research and design tool in human–computer interaction. Human–Comput. Inter. **7**(1), 91–139 (1992)
10. Majaranta, P., Räihä, K.J.: Twenty years of eye typing: systems and design issues. In: ETRA 2002, pp. 15–22. New Orleans, USA (2002)
11. Martinez-Conde, S., Macknik, S.L., Hubel, D.H.: The role of fixational eye movements in visual perception. Nat. Rev. Neurosci. **5**(3), 229–240 (2004)
12. Miniotas, D.: Application of Fitts' law to eye gaze interaction. In: CHI EA 2000, pp. 339–340. Hague, Netherlands (2000)
13. Purwanto, D., Mardiyanto, R., Arai, K.: Electric wheelchair control with gaze direction and eye blinking. Artif. Life Robot. **14**(3), 397–400 (2009)
14. Rayner, K.: Eye movements in reading and information processing: 20 years of research. Psychol. Bull. **124**(3), 372–422 (1998)
15. Shannon, C.E.: A mathematical theory of communication. Bell Syst Tech J **27**(379–423), 623–656 (1948)
16. Skovsgaard, H., Hansen, J.P., Møllenbach, E.: Gaze tracking through smartphones. In: Gaze Interaction in the Post-WIMP World: CHI 2013 Workshop, Paris, France (2013)
17. Soukoreff, R.W., MacKenzie, I.S.: Towards a standard for pointing device evaluation, perspectives on 27 years of Fitts' law research in HCI. Int. J. Human–Comput. Stud. **61**(6), 751–789 (2004)
18. Ware, C., Mikaelian, H.H.: An evaluation of an eye tracker as a device for computer input. In: CHI/GI 1987, pp. 183–188. Toronto, Canada (1987)
19. Zemblys, R.: Interpolation based method for approximation of eyesight data from separate sensors. Prof. Stud.: Theory Pract. **10**, 159–166 (2012)
20. Zhang, X., MacKenzie, I.S.: Evaluating Eye Tracking with ISO 9241—Part 9. In: Jacko, J.A. (ed.) Human–Computer Interaction. HCI Intelligent Multimodal Interaction Environments, LNCS, vol. 4552, pp. 779–788. Springer, Berlin Heidelberg (2007)

Part III
Robot Control, Embedded and Navigation Systems

On Control of Human Arm Switched Dynamics

Artur Babiarz

Abstract In this paper, the analysis of switched human dynamics is shown. The analysis concerns the use of fractional-order $PI^{\mu}D^{\lambda}$ controller and integer-order *PID* controller. The above-mentioned controllers are applied to control the non-linear plant, which is the human arm. The control object is described as a non-linear continuous-time switched system. The switching rule is state-dependent. At the end of the article, illustrative examples are presented. The examples show the influence of fractional order controller parameters on the quality of the responses to a given input signal.

Keywords Fractional order PID controller · Switched system · Human arm · Switching rule

1 Background

In the last decade, we can observe a growing interest in fractional order systems and the application of fractional order PID controller. PID controller design for fractional-order systems with time delays is described in [17]. In [26], application of fractional order PID controller to an automatic voltage regulator is presented and studied. The authors of [21] present the design of Fractional Order Proportional Integral Derivative Controller (FOPID) for liquid level control of a spherical tank. The tank is modeled as a First Order Plus Dead Time (FOPDT) system about an operating point. The article [6] involves the design of fractional order PID controller for the plant which is described by the fractional order transfer function. The base control methods originating in kinematics, dynamics and on-line imitation of human motion are looked at authors of [10] describe an attempt to create a lower limb exoskeleton control system with the PID regulators and conduct an analysis of its stability by means of Nyquist

A. Babiarz (✉)
Institute of Automatic Control, Silesian University of Technology, Gliwice, Poland
e-mail: artur.babiarz@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_12

criterion. The article contains a full description of conducted experiments for one degree of freedom only. In [7] the research concerning construction and control of exoskeletons is presented. Its control system was examined from the perspective of moment control with the use of PI regulator as well as trajectory generation with use of PID regulator. The authors of [8] described in great detail mechanical aspects of construction of artificial human arm, and presented a simple control algorithm based on PID controller. In [11], the information can be found, where authors put special attention to precise mapping of anthropomorphic parameters of human body for the purpose of creation of exoskeleton representing whole musculoskeletal system. Similar research was published in [12], where work is concerning aspects of anthropomorphic arm's elasticity influence on accuracy of motions performed and the control quality. The control system of such object was designed, with the use of PID regulator.

In [20] research on upper limb prosthesis construction were described together with results of PD regulation of the prosthesis following a sinusoidal trajectory.

The application of standard PI and PD regulators can be found in [18, 22]. The translation of human arm control into the robotic domain is given in [18]. The authors of [18] use fundamental models in the form of harmonic oscillator's dumped motion for the purpose of trajectory planning. The control system is composed of such model and a PD regulator. In [22] the controlled object is represented by a standard equation of motion for rigid body with friction. Simulation results are obtained with the use of PI regulator and compensation for friction.

The remaining part of this paper is organized as follows. In Sect. 2, the mathematical description of fractional and integer order PID controller is presented. Additionally, the switched model of two-link human arm is shown. The simulation results are presented in Sect. 3. Finally, conclusions are drawn in Sect. 4.

2 Preliminaries

2.1 Fractional and Integer Order PID Controller

PID-Integer Controller Generally speaking, the continuous form of a PID controller is given by following formula [9]:

$$u(t) = K_P e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{de(t)}{dt}, \tag{1}$$

where: K_P is the proportional gain, K_I is the integral gain, K_D is the derivative gain, $e(t)$ is error signal, and $u(t)$ is output signal.

PID-Fractional Controller The equation for the $PI^{\mu}D^{\lambda}$-controller's output in the time domain has a form [19, 24]:

$$u(t) = K_P e(t) + K_I \mathfrak{D}_t^{-\lambda} e(t) + K_D \mathfrak{D}_t^{\mu} e(t). \tag{2}$$

where: $\mathfrak{D}_t^{\alpha}$ is the fractional integro-differential operator.

Definition 1 The continuous integro-differential operator is defined by the following form [24]:

$${}_a\mathfrak{D}_t^{\alpha} = \begin{cases} \frac{d^{\alpha}}{dt^{\alpha}} & \text{if } \alpha > 0 \\ 1 & \text{if } \alpha = 1 \\ \int_a^t (d\tau)^{-\alpha} & \text{if } \alpha < 0 \end{cases} \tag{3}$$

where: α is the operator order, a and t denote the limits of the operation.

Definition 2 Grünwald-Letnikov definition of the fractional-order differ-integral operator is as follows [24]:

$${}_a\mathfrak{D}_t^{\alpha} f(t) = \lim_{h \to 0} \frac{1}{h^{\alpha}} \sum_{j=0}^{k} (-1)^j \binom{\alpha}{j} f(t - jh). \tag{4}$$

where $a = 0$, $t = kh$, k is the number of steps, and h is the step size.

2.2 Non-linear Mathematical Model of Human Arm

Figure 1 presents a kinematic scheme of two-link arm. In this case, the motion equation is presented in the following non-linear differential equation [2, 5]:

$$\frac{d}{dt} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ M^{-1}(q) \left[u - C(q, \dot{q})\dot{q} - G(q) - W\dot{q} \right] \end{bmatrix} \tag{5}$$

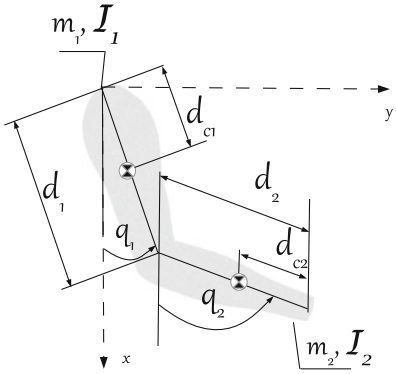


Fig. 1 Kinematic scheme of two-link arm

where:

$$M(q) = \begin{bmatrix} m_1 d_{c1}^2 + m_2 d_1^2 + I_1 & m_2 d_1 d_{c2} \cos(q_1 - q_2) \\ m_2 d_1 d_{c2} \cos(q_1 - q_2) & m_2 d_{c2}^2 + I_2 \end{bmatrix},$$

$$C(q, \dot{q}) = \begin{bmatrix} 0 & m_2 d_1 d_{c2} \sin(q_1 - q_2)\dot{q}_2 \\ -m_2 d_1 d_{c2} \sin(q_1 - q_2)\dot{q}_1 & 0 \end{bmatrix},$$

$$G(q) = \begin{bmatrix} -(m_1 d_{c1} + m_2 d_1) g \sin q_1 \\ -m_2 d_{c2} g \sin q_2 \end{bmatrix}, \quad B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$$

and $M(q) \in \mathbb{R}^{2\times 2}$—is a positive definite symmetric inertia matrix, $C(q, \dot{q}) \in \mathbb{R}^{2\times 2}$—is a vector centripetal and Coriolis forces, $G(q) \in \mathbb{R}^2$—is gravity forces vector, $B \in \mathbb{R}^{2\times 2}$—is the joint friction matrix, $u = [u_1\, u_2]^T \in \mathbb{R}^2$—is the joint torque, $q = [q_1\, q_2]^T \in \mathbb{R}^2$—is the angular displacement, m_i—is the mass, d_i—is the link length, d_{ci}—is the distance from the joint to the center of mass, I_i—is the moment of inertia, i—is the number of human link, $i = 1, 2$.

In order to obtain an equivalent set of first-order state equations, the state variables of equations (5) are given by the following form:

$$x_1 = q_1, \quad x_2 = q_2, \quad x_3 = \dot{x}_1 = \dot{q}_1, \quad x_4 = \dot{x}_2 = \dot{q}_2,$$

$$x = \left[q_1,\, q_2,\, \dot{q}_1,\, \dot{q}_2 \right]^T.$$

Then, the two-link human arm system into a state space form can be expressed as a vector first-order non-linear differential equations:

$$\dot{x} = F(x) + G(x)u. \tag{6}$$

In (6), the vector functions $F(x)$, $G(x)$ are given by

$$F(x) = [F_1(x), F_2(x), F_3(x), F_4(x)]^T,$$

where:

$$F_1(x) = x_3, \quad F_2(x) = x_4,$$

$$F_3(x) = \frac{m_2^2 d_1^2 d_{c2}^2 \sin(x_1 - x_2)\cos(x_1 - x_2)}{Det(M)} x_3^2 +$$

$$- \frac{m_2^3 d_1^3 d_{c2}^3 \sin(x_1 - x_2)\cos^2(x_1 - x_2)}{(m_1 d_{c1}^2 + m_2 d_1^2 + I_1)} x_4^2 +$$

$$- \left(\frac{m_2 d_1 d_{c2} \sin(x_1 - x_2)}{m_1 d_{c1}^2 + m_2 d_1^2 + I_1} \right) x_4^2 + \left(\frac{m_2 d_1 d_{c2} \cos(x_1 - x_2) b_{21}}{Det(M)} \right) x_3 +$$

$$-\left(\frac{m_2^2 d_1^2 d_{c2}^2 \cos^2(x_1 - x_2) b_{11}}{Det(M)(m_1 d_{c1}^2 + m_2 d_1^2 + I_1)} + \frac{b_{11}}{m_1 d_{c1}^2 + m_2 d_1^2 + I_1}\right) x_3 +$$

$$-\left(\frac{m_2^2 d_1^2 d_{c2}^2 \cos^2(x_1 - x_2) b_{12}}{Det(M)(m_1 d_{c1}^2 + m_2 d_1^2 + I_1)}\right) x_4 +$$

$$+\left(\frac{m_2 d_1 d_{c2} \cos(x_1 - x_2) b_{22}}{Det(M)} - \frac{b_{12}}{m_1 d_{c1}^2 + m_2 d_1^2 + I_1}\right) x_4 +$$

$$+\left(\frac{m_2^2 d_1^2 d_{c2}^2 \cos^2(x_1 - x_2)(m_1 d_{c1} + m_2 d_1) g \sin(x_1)}{Det(M)(m_1 d_{c1}^2 + m_2 d_1^2 + I_1)}\right) +$$

$$+\left(\frac{(m_1 d_{c1} + m_2 d_1) g \sin(x_1)}{m_1 d_{c1}^2 + m_2 d_1^2 + I_1}\right) - \frac{m_2^2 d_1 d_{c2}^2 g \sin(x_2) \cos(x_1 - x_2)}{Det(M)},$$

$$F_4(x) = \frac{m_2^2 d_1^2 d_{c2}^2 \sin(x_1 - x_2) \cos(x_1 - x_2)}{Det(M)} x_4^2 +$$

$$+\frac{m_2 d_1 d_{c2} \sin(x_1 - x_2)(m_1 d_{c1}^2 + m_2 d_1^2 + I_1)}{Det(M)} x_3^2 + \frac{m_2 d_1 d_{c2} \cos(x_1 - x_2) b_{11}}{Det(M)} x_3 +$$

$$-\frac{(m_1 d_{c1}^2 + m_2 d_1^2 + I_1) b_{21}}{Det(M)} x_3 + \frac{m_2 d_1 d_{c2} \cos(x_1 - x_2) b_{12}}{Det(M)} x_4 +$$

$$-\frac{(m_1 d_{c1}^2 + m_2 d_1^2 + I_1) b_{22}}{Det(M)} x_4 + \frac{1}{Det(M)} \left((m_1 d_{c1}^2 + m_2 d_1^2 + I_1) m_2 d_{c2} g \sin(x_2)\right) +$$

$$+\frac{1}{Det(M)} (m_2 d_1 d_{c2} (m_1 d_{c1} + m_2 d_1) g \sin(x_1) \cos(x_1 - x_2))$$

and

$$G(x) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ \frac{Det(M) + m_2^2 d_1^2 d_{c2}^2 \cos(x_1 - x_2)}{Det(M)(m_1 d_{c1}^2 + m_2 d_1^2 + I_1)} & -\frac{m_2 d_1 d_{c2} \cos(x_1 - x_2)}{Det(M)} \\ -\frac{m_2 d_1 d_{c2} \cos(x_1 - x_2)}{Det(M)} & \frac{m_1 d_{c1}^2 + m_2 d_1^2 + I_1}{Det(M)} \end{bmatrix}.$$

The determinant of matrix M is equal to

$$Det(M) = m_2 d_{c2}^2 (m_1 d_{c1}^2 + m_2 d_1^2 + I_1 - m_2 d_1^2 \cos^2(x_1 - x_2)) +$$

$$+ (m_1 d_{c1}^2 + m_2 d_1^2) I_2 + I_1 I_2.$$

Remark 1 Each muscle changes its shape during movement of the limbs. This property influences the deformation of external shape of the limb. This proposal is the result of analysis of research published in [13, 14, 16].

Remark 2 Under the above conclusions, we can assume that the matrix of inertia and the distance from the center of gravity of each joint, are changed. In addition, changes of these parameters are dependent on the angular displacement of the arm [3, 4].

Remark 3 It is mentioned the muscles effect is omitted. It means that the muscles have effect on the exterior shape of the each link only.

At this point, we can consider human arm as switched non-linear system with state-dependent switching. Then, the switched continuous-time non-linear system can be expressed in the following form:

$$\dot{x}(t) = \begin{cases} F_{\sigma_1}(x(t)) + G_{\sigma_1}(x(t))u(t) & \text{if } x_1 \leq 0, x_2 > 0 \\ F_{\sigma_2}(x(t)) + G_{\sigma_2}(x(t))u(t) & \text{if } x_1 \leq 0, x_2 = 0 \\ F_{\sigma_3}(x(t)) + G_{\sigma_3}(x(t))u(t) & \text{if } x_1 > 0, x_2 \geq 0 \end{cases} \tag{7}$$

2.3 A Control Scheme

The control scheme is presented on Fig. 2. It is consists of N controllers and one plant. The plant is modeled as switched non-linear system. In this case, the plant is two-link human arm. The mathematical model of human arm is very simple. Albeit, it is sufficient for presented research results.

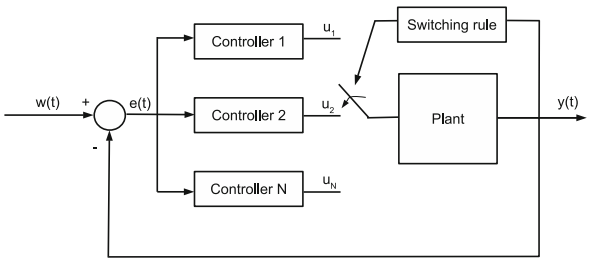


Fig. 2 Block diagram of a switching system

3 The Simulation Results

In practical applications, the PID controller is tuned in the control system which is used. Tuning rule can be briefly summarized as follows: 1. Determination of the value of K_P in order to obtain the required speed of response. 2. Select TI Integral control in order to achieve the assumed steady-state quality (this may be a necessity to adjust the value of K_P). 3. Add derivative control to reduce over-regulations and improve the control time.

The parameters of fractional and integer PID controllers, that are used in simulation experiments, are presented in Table 1. The parameters of two-link human arm, that are used during simulations, are shown in Table 2 [15].

The simulation results were obtained using Matlab Simulink package, S-Function and FOMCON (Fractional-order modeling and control toolbox for MATLAB) [23]. The first simulation experiment has initial condition is equal $x_{start} = \left[-0.52\,[rad];\, 0.17\,[rad];\, 0\,[\frac{rad}{s}];\, 0\,[\frac{rad}{s}]\right]^T$ and the goal point is equal $x_{goal} = \left[0.78\,[rad];\, 0.52\,[rad];\, 0\,[\frac{rad}{s}];\, 0\,[\frac{rad}{s}]\right]^T$. A time of simulation is set on 1.2 $[s]$. The results of experiment I are presented on Figs. 3 and 4. The parameters of second experiment are following: $x_{start} = \left[0\,[rad];\, 0\,[rad];\, 0\,[\frac{rad}{s}];\, 0\,[\frac{rad}{s}]\right]^T$, $x_{goal} = \left[-0.52\,[rad];\, 0.52\,[rad];\, 0\,[\frac{rad}{s}];\, 0\,[\frac{rad}{s}]\right]^T$. The time of simulation is equal 1.2 $[s]$. Figures 5 and 6 present time history of four elements of state vector.

Table 1 Parameters of fractional and integer order PIDcontroller

	Case I				Case II				Case III			
	Signal u_1		Signal u_2		Signal u_1		Signal u_2		Signal u_1		Signal u_2	
	PID	$PI^{\lambda}D^{\mu}$	PID	$PI^{\lambda}D^{\mu}$	PID	$PI^{\lambda}D^{\mu}$	PID	$PI^{\lambda}D^{\mu}$	PID	$PI^{\lambda}D^{\mu}$	PID	$PI^{\lambda}D^{\mu}$
K_P	100	100	100	100	100	100	100	100	100	100	100	100
K_I	5	5	5	5	7	7	7	7	9	9	9	9
K_D	16	16	15	15	12	12	11	11	12	12	10	10
λ	–	0.8	–	0.8	–	0.7	–	0.7	–	0.7	–	0.7
μ	–	0.95	–	0.95	–	0.92	–	0.92	–	0.91	–	0.91

Table 2 Parameters of two-link arm

		m (kg)	l (m)	
Link 1		1.4	0.3	
Link 2		1.1	0.33	
	l_{c1} (m)	l_{c2} (m)	I_1 (kg m^2)	I_2 (kg m^2)
Subsystem I	0.11	0.16	0.027	0.045
Subsystem II	0.1	0.14	0.018	0.04
Subsystem III	0.11	0.14	0.02	0.04

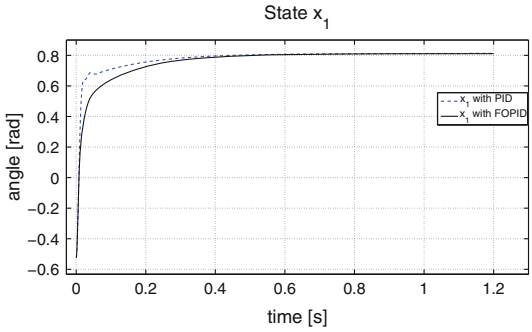


Fig. 3 The scope of state x_1 from the first experiment

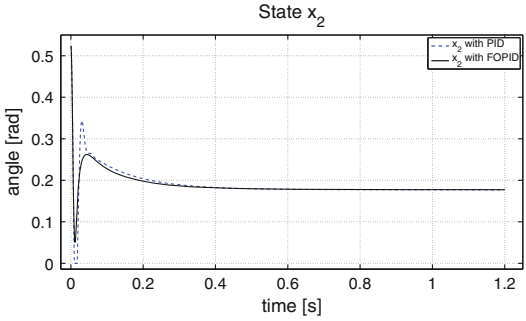


Fig. 4 The scope of state x_2 from the first experiment

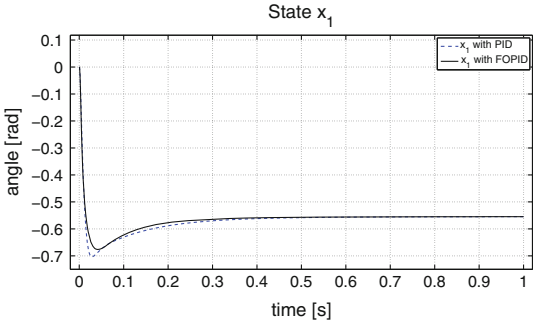


Fig. 5 The scope of state x_1 from the second experiment

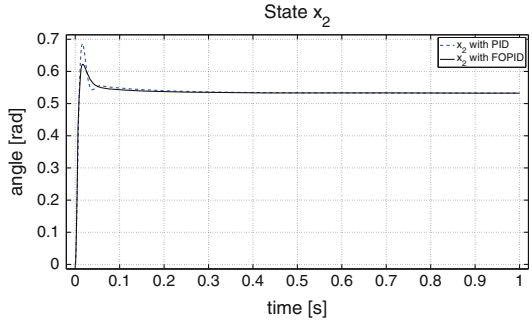


Fig. 6 The scope of state x_2 from the second experiment

4 Concluding Remarks

In the article, we present control scheme of two-link human arm using two different types PID controllers. The two-link human arm is modeled by switched non-linear systems. It may be noted that the fractional order controller copes better with the above mentioned the dynamics model of the human arm. The Figs. 3–6 show that there are less over-regulations and the system quickly reaches a steady-state. A fortiori, fractional PID controller generates better results if the initial point of the object coincides with an equilibrium point. The equilibrium point of human arm is specific, because the trigonometric functions hit limits at this point and the plant is difficult to control. At this point, we can also conclude that the better behaviour of the control system with fractional order controller is the result of the additional controller tuning using the λ and μ parameters. On the other hand, the settings of integer and fractional order controllers are not optimal [1], and the results can be valid for the considered type of object only. Furthermore, the article studies the position control only without a speed control. At this stage, it is difficult to assess the impact of other elements of the state on quality of the control signal obtained from the fractional order controllers.

It should be pointed out that the description of the human arm dynamics is very basic, obviously. However, its structure is appropriate to the analysis of control system with integer and fractional order controllers (see e.g. [10, 15, 25]).

Acknowledgments The research presented here were funded by the Silesian University of Technology grant BK-227/RAu1/2015/2.

References

1. Åström, K., Hägglund, T.: Advanced PID Control. Systems, and Automation Society, ISA-The Instrumentation (2006)
2. Babiarz, A., Bieda, R., Jaskot, K., Klamka, J.: The dynamics of the human arm with an observer for the capture of body motion parameters. Bull. Pol. Acad. Sci. Tech. Sci. **61**(4), 955–971 (2013)
3. Babiarz, A., Czornik, A., Klamka, J., Niezabitowski, M., Zawiski, R.: The mathematical model of the human arm as a switched linear system. MMAR **2014**, 508–513 (2014)
4. Babiarz, A., Klamka, J., Zawiski, R., Niezabitowski, M.: An approach to observability analysis and estimation of human arm model. In: ICCA 2014, pp. 947–952. Taichung, Taiwan (2014)
5. Babiarz, A.: On mathematical modelling of the human arm using switched linear system. In: ICNPAA 2014, vol. 1637, pp. 47–54. Miedzyzdroje, Poland (2014)
6. Badri, V., Tavazoei, M.S.: On tuning fractional order proportional-derivative controllers for a class of fractional order systems. Automatica **49**(7), 2297–2301 (2013)
7. Beyl, P., Van Damme, M., Van Ham, R., Vanderborght, B., Lefeber, D.: Design and control of a knee exoskeleton powered by pleated pneumatic artificial muscles for robot-assisted gait rehabilitation. Appl. Bion. Biomech. **6**(2), 229–243 (2009)
8. Cattin, E., Roccella, S., Vitiello, N., Sardellitti, I., Artemiadis, P.K., Vacalebri, P., Vecchi, F., Carrozza, M.C., Kyriakopoulos, K.J., Dario, P.: Design and development of a novel robotic

platform for neuro-robotics applications: the NEURobotics ARM (NEURARM). Adv. Robot. **22**(1), 3–37 (2008)
9. Chang, W.D., Hwang, R.C., Hsieh, J.G.: A self-tuning PID control for a class of nonlinear systems based on the lyapunov approach. J. Process Control **12**(2), 233–242 (2002)
10. Csercsik, D.: Analysis and control of a simple nonlinear limb model. Ph.D. thesis, Budapest University of Technology and Economics (2005)
11. Firmani, F., Park, E.J.: A comprehensive human-body dynamic model towards the development of a powered exoskeleton for paraplegics. Trans. Can. Soc. Mech. Eng. **33**(4), 745–757 (2009)
12. Haddadin, S., Krieger, K., Kunze, M., Albu-Schaffer, A.: Exploiting potential energy storage for cyclic manipulation: an analysis for elastic dribbling with an anthropomorphic robot. In: ICIRS 2011, pp. 1789–1796. San Francisco, USA (2011)
13. Lee, D., Glueck, M., Khan, A., Fiume, E., Jackson, K.: A survey of modeling and simulation of skeletal muscle. ACM Trans. Graph. **28**(4), 162–174 (2010)
14. Lee, D., Glueck, M., Khan, A., Fiume, E., Jackson, K.: Modeling and simulation of skeletal muscle for computer graphics: a survey. Found. Trends Comput. Graph. Vis. **7**(4), 229–276 (2012)
15. Li, W.: Optimal control for biological movement systems. Ph.D. thesis, University Of California (2006)
16. Neumann, T., Varanasi, K., Hasler, N., Wacker, M., Magnor, M., Theobalt, C.: Capture and statistical modeling of arm-muscle deformations. Comput. Graph. Forum **32**(2pt3), 285–294 (2013)
17. Ozbay, H., Bonnet, C., Fioravanti, A.R.: PID controller design for fractional-order systems with time delays. Syst. Control Lett. **61**(1), 18–23 (2012)
18. Pastor, P., Kalakrishnan, M., Meier, F., Stulp, F., Buchli, J., Theodorou, E., Schaal, S.: From dynamic movement primitives to associative skill memories. Robot. Auton. Syst. **61**(4), 351–361 (2013)
19. Podlubny, I.: Fractional-order systems and $PI^{\lambda}D^{\mu}$-controllers. Trans. Autom. Control **44**(1), 208–214 (1999)
20. Rosen, J., Perry, J.C.: Upper limb powered exoskeleton. Int. J. Humanoid Rob. **4**(3), 529–548 (2007)
21. Sundaravadivu, K., Arun, B., Saravanan, K.: Design of fractional order PID controller for liquid level control of spherical tank. In: ICCSCE 2011, pp. 291–295. Penang (2011)
22. Tejado, I., Valerio, D., Pires, P., Martins, J.: Fractional order human arm dynamics with variability analyses. Mechatronics **23**(7), 805–812 (2013)
23. Tepljakov, A., Petlenkov, E., Belikov, J.: FOMCON: Fractional-order modeling and control toolbox for MATLAB. In: MIXDES 2011, pp. 684–689. Gliwice (2011)
24. Tepljakov, A., Petlenkov, E., Belikov, J., Halas, M.: Design and implementation of fractional-order PID controllers for a fluid tank system. In: ACC 2013, pp. 1777–1782. Washington, USA (2013)
25. Ueyama, Y., Miyashita, E.: Optimal feedback control for predicting dynamic stiffness during arm movement. Trans. Ind. Electron. **61**(2), 1044–1052 (2014)
26. Zamani, M., Karimi-Ghartemani, M., Sadati, N., Parniani, M.: Design of a fractional order PID controller for an AVR using particle swarm optimization. Control Eng. Pract. **17**(12), 1380–1387 (2009)

Incorporating Static Environment Elements into the EKF-Based Visual SLAM

Adam Schmidt

Abstract The paper presents a visual simultaneous localization and mapping (SLAM) system extended to work with additional, static elements of the environment. Additional measurements have been introduced by incorporating the static surveillance cameras and artificial markers placed in the environment. This reduced the influence of the inherent scale ambiguity of the monocular systems and the tracking drift on the trajectory tracking. Consequently, the root mean square of the absolute trajectory error was reduced by 23 % when compared to the well-established MonoSLAM system.

Keywords SLAM · Robot navigation · Robot vision

1 Introduction

The ability to autonomously operate in an initially unknown environment is of crucial importance in mobile robotics. Over the years a significant effort has been put into development of the simultaneous localization and mapping (SLAM) systems. Although different sensors can be used for this purpose, the vision-based algorithms play the most prominent role. Several widely recognized visual SLAM systems have been developed including the MonoSLAM [3], the FastSLAM [7], the PTAM [4], the FrameSLAM [5] or the systems based on the g2o framework [6].

Most of those systems were designed for the monocular scenario which means that a inexpensive, off-the-shelf camera can be used. However, such systems suffer from the inherent ambiguity of scale of the recovered trajectories, as a single camera does not provide any information regarding the depth of the scene.

Moreover, they use the characteristic points of the scene (so called point features) to build the map of the environment. Though such an approach is generally efficient it may fail in the case of large, open spaces with few distinct features (e.g. big,

A. Schmidt (✉)
Poznan University of Technology, Poznan, Poland
e-mail: adam.schmidt@put.poznan.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_13

empty rooms) or environment with numerous similar objects (e.g. long corridors with similar doors and windows).

This paper presents a modification of the MonoSLAM system attempting to cope with those limitations. The system is extended with the artificial markers similar to those proposed by Baczyk [1]. As a result observations of objects of known scale are be introduced, which facilitates establishing scale of the trajectory. However, the markers were also placed on the mobile robot itself. As a result it was possible to incorporate static, external cameras into the system, which in turn improved the system's ability to close loops (recognize previously visited areas).

The next section present the proposed visual SLAM systems. Afterwards the design of the artificial markers and methods for their detection and pose calculation are presented. Then, the experimental setup and the results are given followed by final conclusions.

2 Visual SLAM System

2.1 Environment Model

The model of the environment is based on the probabilistic map approach of the MonoSLAM system [3] and is assumed to consist of a mobile robot, cameras and artificial markers attached to the robot or placed statically in the environment, and point features (Fig. 1):

$$x = [\, x_r \;\; x_c^1 \;\; \ldots \;\; x_c^{nC} \;\; x_a^1 \;\; \ldots \;\; x_a^{nA} \;\; x_f^1 \;\; \ldots \;\; x_f^{nF} \,]^T \tag{1}$$

where nC, nA and nF stand for the number of the cameras, of the markers and of the point features correspondingly. The robot is modeled according to the 'agile camera' model [3] and its state vector contains the Cartesian position, orientation quaternion as well as the robot's linear and angular velocities:

$$x_r = \left[r \;\; q \;\; v \;\; \omega \right]^T \tag{2}$$

The inverse depth representation [2] is used to model the point features. The state vector of each feature contains the position from which it has been observed for the first time, two angles coding the direction of the line passing through the feature and the point of its initialization and the inverse of the distance between the feature and the point of initialization:

$$x_f = \left[x_0 \;\; y_0 \;\; z_0 \;\; \gamma \;\; \theta \;\; \rho \right]^T \tag{3}$$

Finally, the state vectors of both the cameras and markers consist of a Cartesian position vector and a quaternion representing the orientation relative to the global coordinates (in the case of the static cameras and markers) or to the robot's coordinate system (in case of the onboard markers and cameras):

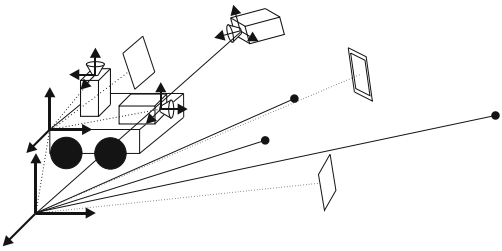


Fig. 1 The elements of the environment model

$$x_c = \left[r^c \; q^c \right]^T \tag{4}$$

$$x_a = \left[r^a \; q^a \right]^T \tag{5}$$

It is assumed that the uncertainty of the model can be represented with a single, multi-variate, zero-mean Gaussian described by the covariance matrix P.

2.2 Cameras and Markers Initialization

The poses of the of the robot's onboard markers and cameras relative to the robot's coordinate system are assummed to be known before the start of the system. The calibration based on minimizing the reprojection error of the observed calibration patterns was used to establish the poses of the onboard equipment. The detailed description of the calibration procedure is available in the paper by Schmidt et al. [10].

However, it is impossible to accurately measure the pose of the static cameras and markers within the SLAM system's map coordinate system (which is coincident with the initial pose of the robot). Therefore, the static elements of the environment have to be initialized when they are encountered by the robot for the first time. There are two possible scenarios of initialization. Either a static marker is observed by a robot's camera or a static camera observes a robot's onboard marker. In the first case the pose of the marker in the global coordinate systems is calculated as:

$$x_a^{new} = \begin{bmatrix} r^a \\ q^a \end{bmatrix} = \begin{bmatrix} R(q^r)t^{ra} + r^r \\ q^r \times q^{ra} \end{bmatrix} \tag{6}$$

$$r^{ra} = R(q^c)r^{ca} + r^c \tag{7}$$

$$q^{ra} = q^c \times q^{ca} \tag{8}$$

where r^c and q^c stand for the precalibrated position and orientation of the robot's camera, r^r and q^r describe the robot's pose and $\times$ is the Grassman product of quaternions.

In case of the observation of the robot's marker the camera's pose is calculated as:

$$x_c^{new} = \begin{bmatrix} r^c \\ q^c \end{bmatrix} = \begin{bmatrix} R(q^r)r^{rc} + r^r \\ q^a \times q^{rc} \end{bmatrix} \tag{9}$$

$$r^{rc} = R(q^a)r^{ac} + r^a \tag{10}$$

$$q^{rc} = q^a \times q^{ac} \tag{11}$$

where r^a and q^a describe the pose of the robot's marker.

2.3 Measurements and Update

The estimate of the state vector is updated using the standard EKF procedure similarly to the MonoSLAM [3] or the system by Schmidt [9]. The measurements vector h consists of visual observations of point features (h_f^i) and markers placed in the environment or on the robots (h_a^j):

$$h = \begin{bmatrix} h_f^1 \ \ldots \ h_f^{nF} \ h_a^1 \ \ldots \ h_a^{nRel} \end{bmatrix}^T \tag{12}$$

The observations of the point features are obtained by calculating the projections of the features on the current camera image and using the ORB descriptor to to find the most probable position in the neighborhood of the projected features similarly to the other feature-based solutions [3, 9].

The predicted measurement of the static marker observed by a robot's camera (Fig. 2) is calculated as:

$$h^a = \begin{bmatrix} r^{ca} \\ q^{ca} \end{bmatrix} \tag{13}$$

$$r^{ca} = R(q^c)^T \left(r^{ra} - r^c\right) \tag{14}$$

$$r^{ra} = R(q^r)^T \left(r^a - r^r\right) \tag{15}$$

$$q^{ca} = \left(q^c\right)^* \times q^{ra} \tag{16}$$

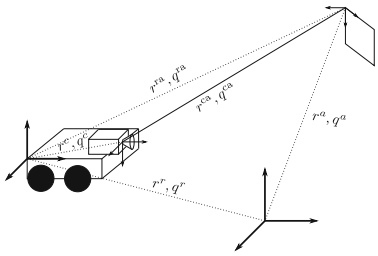


Fig. 2 The prediction of the environment marker observation

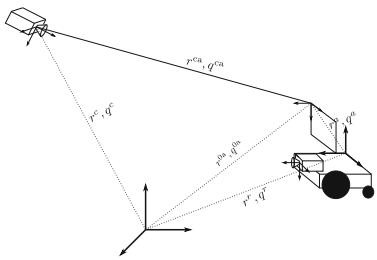


Fig. 3 The prediction of the robot's marker observation

$$q^{ra} = \left(q^r\right)^* \times q^a \tag{17}$$

where q^* is a quaterion q conjugate.

Analogously, the predicted pose of the robot's marker within the static camera's coordinate system (Fig. 3) is calculated according to:

$$h^a = \begin{bmatrix} r^{ca} \\ q^{ca} \end{bmatrix} \tag{18}$$

$$r^{ca} = R(q^c)^T \left(r^{0a} - r^c\right) \tag{19}$$

$$r^{0a} = R(q^r)r^a + r^r \tag{20}$$

$$q^{ca} = \left(q^c\right)^* \times q^{0a} \tag{21}$$

$$q^{0a} = q^r \times q^a \tag{22}$$

3 Artificial Markers

The artificial markers are objects of known size and shape that can be easily detected and identified on the image. Each marker consists of a square black frame on a white background that provides a clear border between the marker and its neighborhood. Four circles distributed inside the frame are the main part of the marker. The origin of the marker's local coordinate system is defined by the center of the black circle; the second and the fourth circle define the x and y axes. The three non-black circles are used to recognize a particular instance of the marker—they can be either red, green or blue thus coding 27 variants of the marker.

The marker detection starts with thresholding the analyzed image with a predefined value, which significantly simplifies the further analysis. Afterwards, the Suzuki's algorithm [12] is used to extract the contours and their hierarchy. Then groups of four contours on the same level of the hierarchy (i.e. lying within the same larger object) are assessed. A set of tests involving the contours' circularity, their relative position, color is used to refine the detection (Fig. 4).

Fig. 4 The marker detection: *left* thresholded image, *center* detected contours, *right* detected marker

Once the marker is detected its pose w.r.t. the observed camera can be calculated. The procedure starts with estimating the homography between the marker plane and the image plane. The rotation and position estimated from the homography matrix are used as a starting point for the second, optimization-based step. The position vector and the orientation quaternion that minimize the reprojection error between the observed centers of circles and their predicted position are estimated using the Levenberg-Marquardt algorithm.

4 Experiments

The data available in the PUT RGB-D dataset [8] was used to evaluate the proposed system. Video sequences from two onboard and a single static cameras recorded during the 9.28 m long robot trajectory consisting of 800 positions were used. Two markers were installed on the robot and four markers were placed in the environment. The selected trajectory contains elements specific for the indoor exploration: alternating turns, forward and backwards movement, loop closing, varying lighting conditions and slight motion blur.

The absolute trajectory error (ATE) metric proposed by Sturm et al. [11] was used to evaluate the results. The SLAM's outputs is a sequence of the estimated robot's positions $r(i)$ where i is the number of the frame. The known, reference positions of the robot are denoted as $r_{\mathrm{GT}}(i)$. In order to facilitate the comparison and to remove the scale ambiguity caused by using the monocular algorithms the rotation R, translation t and scale s aligning the obtained trajectory with the reference trajectory in terms of least-squares were determined. Once the trajectories were aligned, the ATE at iteration i could be calculated as:

$$\mathrm{ATE}(i) = ||r_{\mathrm{GT}}(i) - (sRr(i) + t)|| \tag{23}$$

In order to evaluate the quality of the reconstructed trajectories the maximal value of ATE was found and the RMS of the ATE was calculated:

Table 1 The RMS and maximum ATE error for the canonical system and the system using relative orientation measurements

Environment markers	Static camera	RMSE (m)	Max. ATE (m)
No	No	0.0986	0.1900
No	Yes	0.0844	0.2160
Yes	No	0.0847	0.2359
Yes	Yes	0.0756	0.1497

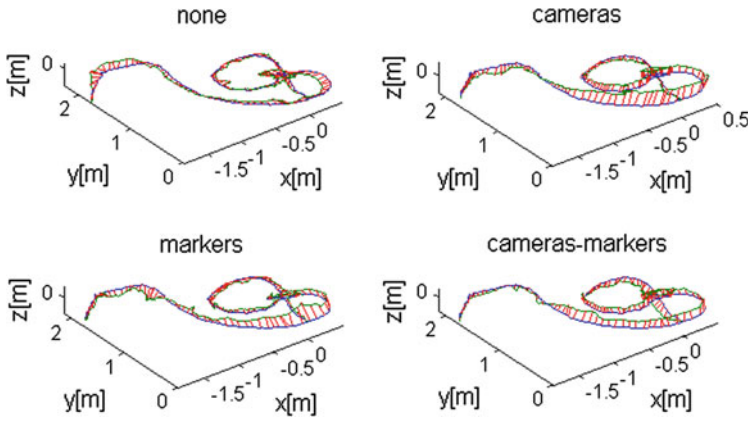


Fig. 5 The trajectories obtained for the canonical system and the system using relative orientation measurements: *blue* gt trajectory, *green* aligned trajectory

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\text{ATE}(i))^2} \tag{24}$$

Table 1 presents the numerical values of the RMSE and maximum ATE while Fig. 5 contains the obtained trajectories. Incorporating the environment markers allowed for the reduction of the RMSE by almost 15 % which confirms the observations of Baczyk [1]. A similar result was achieved by using observations of a robot marker by the static camera. Even bigger reduction of the tracking error was obtained when using both the static camera and the environment markers which decreased the RMSE by 23 %.

5 Conclusions

This paper presented an extension of a canonical, EKF-based SLAM system with additional static elements of the environment. The poses of artificial markers and cameras placed both on the robot and in the environment were incorporated into the system's state vector. As a result, additional measurements e.g. observations of

static markers through the robot's cameras and observations of the robot's markers by static cameras could be used.

As it was shown, the incorporation of the static elements significantly improves the tracking precision of the visual SLAM system. It can be especially important while deploying the system in feature-deficient environments in which the purely feature-based tracking may fail. The main shortcoming of such extension is the necessity of modifying the environment and possibly installing additional cameras. However, the preexisting surveillance infrastructure can be easily integrated into the visual SLAM system increasing its accuracy without any additional costs.

The future work will focus on evaluating the performance of the system in a large-scale environment and assessing its scalability.

Acknowledgments This research was supported by the Polish National Science Centre grant funded according to the decision DEC-2011/01/N/ST7/05940.

References

1. Baczyk, R., Kasiński, A.: Visual simultaneous localisation and map-building supported by structured landmarks. Int. J. Appl. Math. Comput. Sci. **20**(2), 281–293 (2010)
2. Civera, J., Davison, A.J., Montiel, J.: Inverse depth parametrization for monocular SLAM. IEEE Trans. Robot. **24**(5), 932–945 (2008)
3. Davison, A.J., Reid, I.D., Molton, N.D., Stasse, O.: MonoSLAM: real-time single camera SLAM. IEEE Trans. Pattern Anal. Mach. Intell. **29**(6), 1052–1067 (2007)
4. Klein, G., Murray, D.: Parallel tracking and mapping for small AR workspaces. In: ISMAR 2007, pp. 225–234, Nara, Japan (2007)
5. Konolige, K., Agrawal, M.: FrameSLAM: from bundle adjustment to real-time visual mapping. IEEE Trans. Robot. **24**(5), 1066–1077 (2008)
6. Kummerle, R., Grisetti, G., Strasdat, H., Konolige, K., Burgard, W.: g2o: a general framework for graph optimization. In: ICRA 2011, pp. 3607–3613, Shanghai, China (2011)
7. Montemerlo, M., Thrun, S., Koller, D., Wegbreit, B., et al.: FastSLAM: a factored solution to the simultaneous localization and mapping problem. In: AAAI 2002, Edmonton, Canada (2002)
8. Schmidt, A., Fularz, M., Kraft, M., Kasiński, A., Nowicki, M.: An indoor RGB-D dataset for the evaluation of robot navigation algorithms. In: Blanc-Talon, J., Kasinski, A., Philips, W., Popescu, D., Scheunders, P. (eds.) Advanced Concepts for Intelligent Vision Systems, LNCS, vol. 8192, pp. 321–329. Springer, Switzerland (2013)
9. Schmidt, A., Kasiński, A.: The visual SLAM system for a hexapod robot. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) Computer Vision and Graphics, LNCS, vol. 6375, pp. 260–267. Springer, Berlin Heidelberg, Germany (2010)
10. Schmidt, A., Kasiński, A., Kraft, M., Fularz, M., Domagała, Z.: Calibration of the multi-camera registration system for visual navigation benchmarking. Int. J. Adv. Rob. Syst. **11**, 83 (2014)
11. Sturm, J., Magnenat, S., Engelhard, N., Pomerleau, F., Colas, F., Burgard, W., Cremers, D., Siegwart, R.: Towards a benchmark for RGB-D slam evaluation. In: RGB-D Workshop on Advanced Reasoning with Depth Cameras at Robotics: RSS 2011, Los Angeles, USA (2011)
12. Suzuki, S., et al.: Topological structural analysis of digitized binary images by border following. Comput. Vis. Graph. Image. Process. **30**(1), 32–46 (1985)

Prediction-Based Perspective Warping of Feature Template for Improved Visual SLAM Accuracy

Adam Schmidt

Abstract The paper presents an improved method for feature matching in the visual simultaneous localization and mapping (SLAM) system. The appearance of the point feature's neighborhood observed from a different camera pose is estimated according to the predicted displacement of the camera. As a result the precision of feature matching increases and so does the accuracy of the trajectory's reconstruction. The proposed method was compared with the state-of-the-art feature detectors and descriptors in the context of visual SLAM. The obtained results place it on par with the best feature descriptors in terms of the system's accuracy while having significantly smaller computational requirements.

Keywords SLAM · Robot navigation · Robot vision · Feature matching

1 Introduction

The simultaneous localization and mapping (SLAM) is one of the most intensively explored areas of the mobile robotic. Over the last years, the visual systems have been receiving a significant amount of attention mainly due to the availability of inexpensive cameras, simplicity of the measurement models and rich information content of images.

Almost all of the contemporary visual SLAM systems such as MonoSLAM [5], FastSLAM [13], PTAM [8], the FrameSLAM [9] and the g2o framework [10] use matching of the characteristic image points (so called point features) to estimate the trajectory of the camera.

The feature matching methods range from basic template matching [1], tracking the orientation of image patches [12] to application of various state-of-the-art point detectors and descriptors such as the FAST [14] detector with the BRIEF [3] descriptor or ORB [15], SIFT [11] and SURF [2] algorithms. Many attempts have been made to characterize the desired properties of the feature matching algorithms [6] and to evaluate their usefulness in the context of the robot navigation [7, 18–20]. The

A. Schmidt (✉)
Poznan University of Technology, Poznan, Poland
e-mail: adam.schmidt@put.poznan.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_14

general conclusion of those is that the simple methods such as the template matching cannot cope with the features' appearance changes caused by the movement of the camera, varying illumination etc. At the same time the more robust methods usually are not fast enough for the real-time operation.

This paper presents an attempt to predict the appearance of the feature's neighborhood observed from a different camera position. The camera displacement is acquired from the visual SLAM system. As a result a new matching method joining the low complexity of the template matching with the robustness of the descriptor based solutions is developed.

The next section outlines the idea of the visual SLAM system. Then the features and maps representations are described. The final sections present the results of the performed experiments and the concluding remarks.

2 Visual SLAM System

2.1 Environment Model

The system is based on the probabilistic map approach presented in the MonoSLAM [5]. It is assumed that the environment consists of a mobile robot, cameras attached to the robot, local maps and point features:

$$x = [\, x_r \; x_c^1 \; \ldots \; x_c^{nC} \; x_m^1 \; \ldots \; x_m^{nM} \; x_f^1 \; \ldots \; x_f^{nF} \,]^T \tag{1}$$

where nC, nM and nF are the number of robot cameras, local maps and point features correspondingly. The uncertainty of the state estimate is modeled as a single, multivariate Gaussian described with the covariance matrix P.

2.2 Prediction

At each iteration of the extended Kalman filter (EKF) the movement of the robot is predicted using the 'agile model' [5]. The agile model is based on the assumption that the robot's movement is caused by random accelerations. The environment is considered to be static, meaning that the state estimates of all the other elements remain unchanged.

2.3 Measurements and Update

The state estimate is updated according to the observations of the point features' projected onto the current image observed by one of the robot's cameras (Fig. 1). Thus, the measurements vector is defined as:

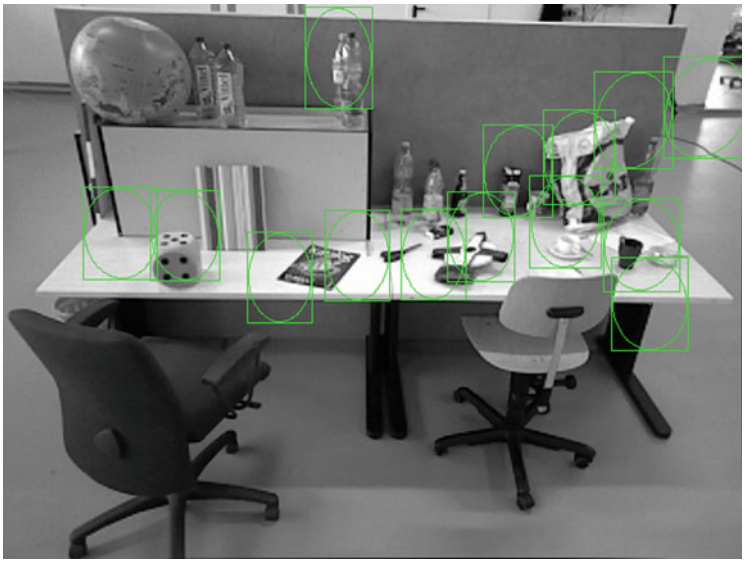


Fig. 1 Observations of the point features

$$h = \left[h_f^1 \ \ldots \ h_f^N \right]^T \tag{2}$$

where N is the number of point features observed at the current iteration. The update of the state vector's estimate is executed according to the standard EKF procedure.

2.4 Maintenance

After each iteration of the EKF the maintenance stage is executed. The features that are not visible often enough or which tend to be wrongly matched are discarded from the system. Thus, the length of the state vector is kept within predefined limits. Analogously, the maps containing too few features are also removed. Afterwards, if the number of visible features is too low, new local maps and features are initialized as described in Sect. 3.2.

3 Local Maps and Features

3.1 Model

The proposed system uses a streamlined representation of features and local maps. This model is a simplification of the Civiera's inverse depth (ID) parametrization [4] and has been already used in a system proposed by Schmidt [16]. The original ID representation consists of the Cartesian coordinates of the point from which the

feature has been initialized (POI), azimuth and elevation angles expressed in the global coordinates coding a line passing through feature and the POI, and finally the inverse of the distance between the POI and the feature:

$$x_{id} = \left[x_0 \; y_0 \; z_0 \; \phi \; \theta \; \rho \right]^T \tag{3}$$

This representation can be easily decomposed into two separate parts. The first, called a local map, describes the pose of the camera during the features' initialization. The map's state vector contains the position vector and the orientation quaternion:

$$x_m = \left[r_m \; q_m \right]^T \tag{4}$$

As a result the features can be expressed w.r.t. the local map's coordinates. The POIs of all the features lies in the map's origin and can be omitted. The state vector of a simplified inverse depth (SID) feature consists of the azimuth and elevation angles expressed in the local coordinates and the inverse of the distance between the feature and the map's origin:

$$x_{sid} = \left[\phi \; \theta \; \rho \right]^T \tag{5}$$

3.2 Initialization

The current pose of the robot's camera in the global coordinate system used for the map initialization is given by:

$$r_m = r_r + R(q_r) r_c \tag{6}$$

$$q_m = q_r \times q_c \tag{7}$$

where $R(q)$ is the rotation matrix equivalent to the quaternion q and $\times$ stands for the Hamilton quaternion product.

The initialization of a map is followed by adding a number of features to the system. The azimuth and elevation angles are expressed within the map's coordinate system. Thus, the state of the ith feature depends only on its observed image coordinates. The initial estimate of the inverse depth is set to a predefined value(ρ_0):

$$\left[h_x^i \; h_y^i \; h_z^i \right]^T = p^{-1} \left(\left[u^i \; v^i \right]^T \right) \tag{8}$$

$$\phi^i = \arctan \left(\frac{h_y^i}{\sqrt{(h_x^i)^2 + (h_z^i)^2}} \right) \tag{9}$$

$$\theta^i = \arctan \left(\frac{h_x^i}{h_z^i} \right) \tag{10}$$

$$\rho^i = \rho_0 \tag{11}$$

where u^i and v^i are the image coordinates of the ith feature's projection, p^{-1} is the inverse of the camera's projection function and h_x^i, h_y^i, h_z^i define the vector passing through both the camera focal point and the feature.

3.3 Prediction Based Template Matching

The proposed, simplified feature representation and the introduction of local maps serve as a base for an alternative approach to robust, correlation based feature matching. The main notion behind the proposed algorithm is that the local maps correspond to the past poses of the robot cameras. Therefore, the appearance of the point feature at the current camera position can be predicted according to the current estimate of the relative pose of the camera and the map, the intrinsic parameters of the camera and the image used to initialize the camera. In order to make it possible, the image used to detect and add new features to the SLAM system is stored during the initialization of the map. It is assumed that the neighborhood of the point feature is planar and lies on the surface normal to the axis of the camera used to initialize the feature.

The matching is performed on the current image observed by the camera. The neighborhood of the feature in the image used for the initialization is transformed to the square template of size s. The predicted position of the feature projection on the current image is given by previously calculated $h_f = [\, u \;\; v \,]^T$. The corners of the $s \times s$ square encompassing the predicted feature projection are given as:

$$c_1^C = \begin{bmatrix} u - 0.5s \\ v - 0.5s \end{bmatrix} \tag{12}$$

$$c_2^C = \begin{bmatrix} u - 0.5s \\ v + 0.5s \end{bmatrix} \tag{13}$$

$$c_3^C = \begin{bmatrix} u + 0.5s \\ v + 0.5s \end{bmatrix} \tag{14}$$

$$c_4^C = \begin{bmatrix} u + 0.5s \\ v - 0.5s \end{bmatrix} \tag{15}$$

The estimated position of the feature in the camera coordinates is calculated according to:

$$C_0^C = \frac{1}{\rho} m^c \tag{16}$$

$$m^c = R(q^c)^T R(q^r)^T R(q^m) m(\phi, \theta) \tag{17}$$

$$m(\phi, \theta) = \begin{bmatrix} \cos(\phi) \sin(\theta) \\ -\sin(\phi) \\ \cos(\phi) \cos(\theta) \end{bmatrix} \tag{18}$$

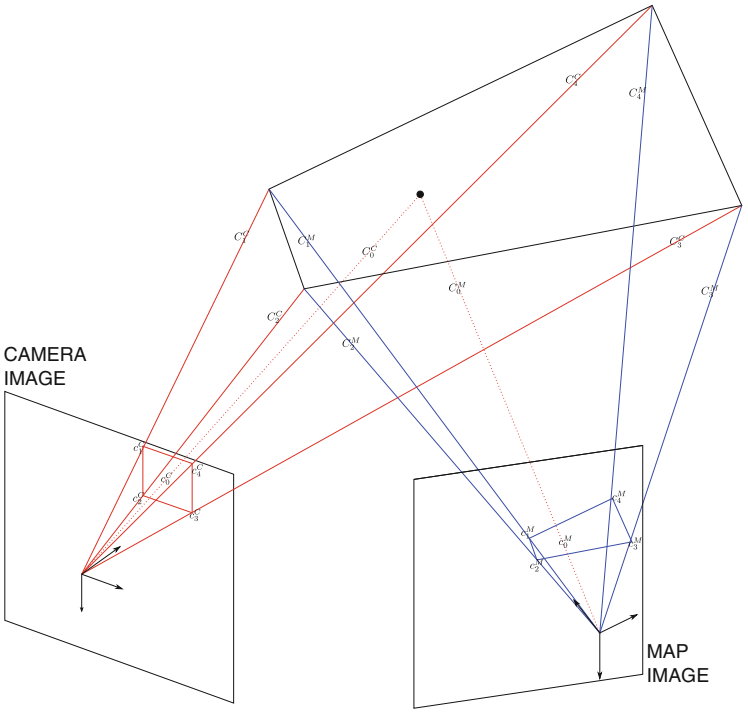


Fig. 2 The warping of a template—the *blue quadrilateral* on the stored map image is transformed into the *red square* on the current camera image

Finally, the map Z-axis corresponding to the past direction of the camera axis expressed in the current coordinates of the camera is given as:

$$z^C = R\left((q^c)^* \times q^m\right)\begin{bmatrix}0\\0\\1\end{bmatrix} \tag{19}$$

The projection of points c_i^C on the surface defined by its normal z^C and point C_0^C are calculated as:

$$C_i^C = p^-1(c_i^C)C_i^C\frac{C_0^C \cdot z^C}{C_i^C \cdot z^c} \tag{20}$$

where $p^-1(c_i^C)$ is the inverse projection of the point c_i^C calculated according to the selected camera model. Afterwards, the points can be transformed to the map coordinate system and projected on the initialization image by using the stored camera parameters:

$$C_i^M = R(q^m)^T \left(R(q^c) C_i^C + t^c - t^m \right) \tag{21}$$

$$c_i^M = p(C_i^M) \tag{22}$$

Finally, the quadrilateral defined by points c_i^M is mapped to the square of size $s \times s$ by an easily calculated perspective transform and bilinear interpolation. The newly computed, warped feature template is used to find the actual position of the feature projection on the current image using the correlation based matching (Figs. 2 and 3).

4 Experiments and Results

The performance and efficiency of the presented feature matching method was evaluated using the data available in the PUT RGB-D database [17]. A video sequence containing elements characteristic for the indoor exploration scenario such as alternating turns, forward and backwards movement, loop closing, varying lighting conditions and slight motion blur was selected. The sequence consisted of 800 images and length of the robot's trajectory equaled 9.28 m.

The prediction-based warping approach was compared with basic template matching and several state-of-the art feature matching algorithms: BRIEF [3], ORB [15], SIFT [11] and SURF [2]. In the case of template matching and the BRIEF descriptor the features were detected using the FAST [14] detector.

Table 1 contains the maximum absolute trajectory error (ATE), the root-mean square error (RMSE) and the average processing time for different feature matching approaches. The proposed, prediction-based template matching achieved the smallest RMSE of all the evaluated algorithms. The maximum ATE of the presented approach was worse only than that of the FAST + BRIEF combination. It is worth noting, that all the multi-scale algorithms performed significantly worse than the proposed method. Moreover, warping the template reduced the max. ATE by almost 50 % and

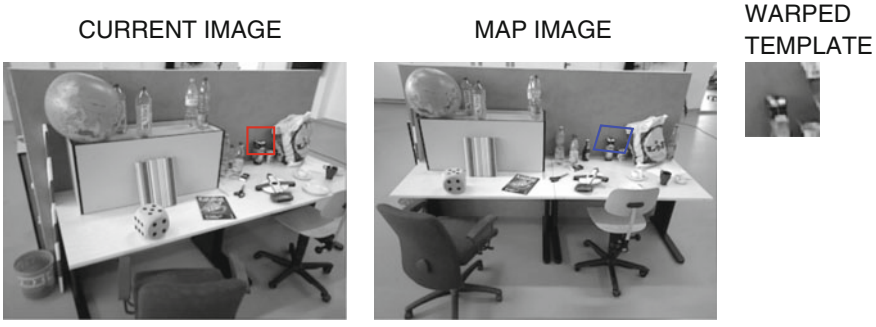


Fig. 3 Example of the template warping

Table 1 The maximal ATE, RMSE and the processing time for various matching approaches

Algorithm	Max. ATE (m)	RMSE (m)	Processing time (s)
FAST + template	0.3433	0.0836	0.014
FAST + warped	0.1720	0.0498	0.012
FAST + BRIEF	0.1234	0.0630	0.004
ORB	0.2400	0.1220	0.020
SIFT	0.2796	0.1256	0.159
SURF	0.4629	0.1841	0.237

the RMSE by over 40 % when compared to the basic template matching. In terms of the computational efficiency the warped template matching is three times slower than the combination of FAST and BRIEF but significantly faster than SIFT and SURF. Still, the processing time of 0.012 s per frame is sufficient for the real-time operation of the system.

5 Conclusions

This paper present a new, simple approach to feature matching using the prediction based, perspective warping of the features' templates. The method was compared with the state-of-the-art algorithms. According to the obtained results the proposed method outperforms the current solutions in terms of the RMSE and is sufficiently fast for using in the real-time systems.

The future work will focus on incorporating the depth information obtained from the RGBD sensors (e.g. Kinect) to model the features' neighborhood more accurately. Moreover, an attempt to parallelize the template warping will be made to improve the processing speed of the proposed method.

Acknowledgments This research was supported by the Polish National Science Centre grant funded according to the decision DEC-2011/01/N/ST7/05940.

References

1. Banks, J., Corke, P.: Quantitative evaluation of matching methods and validity measures for stereo vision. Int. J. Robot. Res. **20**(7), 512–532 (2001)
2. Bay, H., Ess, A., Tuytelaars, T., van Gool, L.: Speeded-up robust features (SURF). Comput. Vis. Image Underst. **110**(3), 346–359 (2008)
3. Calonder, Michael, Lepetit, Vincent, Strecha, Christoph, Fua, Pascal: BRIEF: binary robust independent elementary features. In: Daniilidis, Kostas, Maragos, Petros, Paragios, Nikos (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 778–792. Springer, Heidelberg (2010)

4. Civera, J., Davison, A.J., Montiel, J.: Inverse depth parametrization for monocular SLAM. IEEE Trans. Robot. **24**(5), 932–945 (2008)
5. Davison, A.J., Reid, I.D., Molton, N.D., Stasse, O.: MonoSLAM: real-time single camera slam. IEEE Trans. Patt. Anal. Mach. Intell. **29**(6), 1052–1067 (2007)
6. Gil, A., Mozos, O.M., Ballesta, M., Reinoso, O.: A comparative evaluation of interest point detectors and local descriptors for visual SLAM. Mach. Vis. Appl. **21**(6), 905–920 (2010)
7. Hartmann, J., Klussendorff, J., Maehle, E.: A comparison of feature descriptors for visual SLAM. In: ECMR 2013, pp. 56–61, Barcelona, Spain (2013)
8. Klein, G., Murray, D.: Parallel tracking and mapping for small AR workspaces. In: ISMAR 2007, pp. 225–234, Nara, Japan (2007)
9. Konolige, K., Agrawal, M.: FrameSLAM: from bundle adjustment to real-time visual mapping. IEEE Trans. Robot. **24**(5), 1066–1077 (2008)
10. Kummerle, R., Grisetti, G., Strasdat, H., Konolige, K., Burgard, W.: G^2o: A general framework for graph optimization. In: ICRA 2011, pp. 3607–3613, Shanghai, China (2011)
11. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**(2), 91–110 (2004)
12. Molton, N., Davison, A.J., Reid, I.: Locally planar patch features for real-time structure from motion. In: BMVC 2004, pp. 1–10, London, UK (2004)
13. Montemerlo, M., Thrun, S., Koller, D., Wegbreit, B. et al.: FastSLAM: A factored solution to the simultaneous localization and mapping problem. In: AAAI 2002, pp. 593–598, Edmonton, Canada (2002)
14. Rosten, E., Drummond, T.: Fusing points and lines for high performance tracking. In: ICCV 2005, vol. 2, pp. 1508–1511, Beijing, China (2005)
15. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: an efficient alternative to SIFT or SURF. In: ICCV 2011, pp. 2564–2571, Barcelona, Spain (2011)
16. Schmidt, A.: The EKF-based visual SLAM system with relative map orientation measurements. In: Chmielewski, L., Kozera, R., Shin, B.S., Wojciechowski, K. (eds.) ICCVG 2014. LNCS, vol. 8671, pp. 570–577. Springer, Heidelberg (2014)
17. Schmidt, A., Fularz, M., Kraft, M., Kasiński, A., Nowicki, Michał: An indoor RGB-D dataset for the evaluation of robot navigation algorithms. In: Blanc-Talon, J., Kasinski, A., Philips, W., Popescu, D., Scheunders, Paul (eds.) ACIVS 2013. LNCS, vol. 8192, pp. 321–329. Springer, Heidelberg (2013)
18. Schmidt, A., Kraft, M.: The impact of the image feature detector and descriptor choice on visual slam accuracy. In: Choras, R.S. (ed.) Image Processing and Communications Challenges 6, AISC, vol. 313, pp. 203–210, Springer, Switzerland (2015)
19. Schmidt, A., Kraft, M., Fularz, M., Domagala, Z.: Comparative assessment of point feature detectors and descriptors in the context of robot navigation. J. Autom. Mob. Robot. Intell. Syst. **7**(1) (2013)
20. Schmidt, A., Kraft, M., Kasiński, A.: An evaluation of image feature detectors and descriptors for robot navigation. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L., Wojciechowski, K. (eds.) ICCVG 2010, Part II. LNCS, vol. 6375, pp. 251–259. Springer, Heidelberg (2010)

Interpolation Method of 3D Position Errors Decreasing in the System of Two Cameras

Tadeusz Szkodny

Abstract This chapter proposes the method of 3D position errors decreasing in the system of two cameras. The analysis of determining accuracy of the 3D coordinates of points on the plane template in the shape of rectangle is presented. Each of these points lie at the corners of squares with side of 8 mm. An images of these points were obtained using Edimax IC-7100P cameras from two different points of view and analysed. The position and orientation coordinates of camera relatively to the reference system were calculated. The coordinates of points on the ideal image (without optical distortions) were determined. After reading from the image real coordinates, optical distortion model coefficients of the camera were calculated. After that, errors caused by optical distortion were determined. The coordinates read from the image were corrected and coordinates of observed points in the reference system were calculated. Next to decreasing computed 3D position errors the interpolation method was proposed. In this method the interpolation of errors between ideal and real coordinates of image points was used. Finally, calculated coordinates were compared to them real values and them maximal differences were determined.

Keywords Computer processing of 3D images · Vision detection of the positions · Errors analysis of stereo vision

1 Introduction

One of the basic component of computer intelligence of robots is software, that calculates coordinates of position and orientation of manipulated objects, seen by the cameras. The designing of such software must take into account the errors of coordinates read from the camera matrix. These errors cause inaccuracies of calculations of points coordinates in reference system, associated with technical station.

In order to determine accuracy of calculated coordinates of observed points, analysis of errors is needed. These errors are caused by: reading errors of coordinates from

T. Szkodny (✉)
Institute of Automatic Control, Silesian University of Technology, Gliwice, Poland
e-mail: tadeusz.szkodny@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_15

the matrix of the camera, optical distortions of the camera, errors of parameters that describes optical system of the cameras and errors of calculations. During designing of the vision system, minimization of mentioned errors is needed.

To Edimax IC-7100P camera study, template with points surrounded by circles (Fig. 1) is used. These points lie in the corners of squares with side of 8 mm. In the figure x and y are axis of the reference system, whereas X and Y are axis of the auxiliary coordinate system. Auxiliary system is useful to set camera above the template.

User can read points coordinates directly from picture in pixels using Microsoft Paint program, but this way is connected with possibility of making mistakes. To decrease risk of making such mistakes, reading of these coordinates in this work has been made automatically using image processing algorithm [8] implemented in C# language, in Microsoft Visual C# 2010 Express environment.

To compensate optical distortion errors, correction of the read coordinates has been used. Mathematical model of optical distortions used here is presented in works [1, 5].

The parameters which describe the coordinate system of camera $x_c y_c z_c$ in reference system xyz are coordinates of position and orientation, focal length and size of the pixel. Calculation of every parameter can be made using iteration methods, which minimize square form of errors [2, 4]. Errors of each mentioned parameters occurs in this form. However fundamental disadvantage of these methods is large number of calculations which cause great numerical errors and long time of calculations. In this work, coordinates of position and orientation was calculated using fast and accurate *Camera* algorithm [7]. Precision of calculations of this algorithm amounted to 10^{-6} mm. Focal length and size of the pixels were taken from camera datasheet.

All programs used to calculations in this paper were written in Matlab, on a computer with an processor Intel Pentium T3200 CPU, with a frequency of 2 GHz.

In this work, precision of calculation of 192 points from Fig. 1 was analyzed in two different settings of cameras above this template. In second chapter, results of calculations of cameras position and orientation coordinates are presented. Third chapter contains calculations of mathematical model coefficients of optical

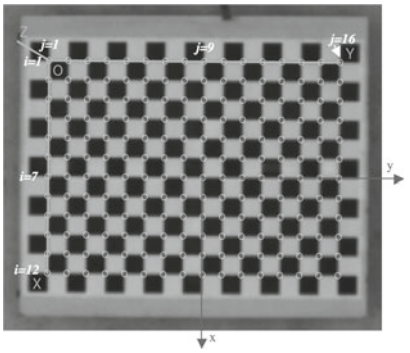


Fig. 1 The template used to camera Edimax IC-7100P study

distortions. Fourth and fifth chapter are about analysis of 3D coordinates calculations errors in reference system, using of the mathematical model of optical distortion and interpolation of coordinates. Sixth chapter summarizes all of the studies.

2 Position and Orientation of Camera

Here, calculation of position and orientation coordinates of cameras in two different angles with respect to plane of template from Fig. 1 is performed. Beginning point O_c of camera coordinates system $x_c y_c z_c$ has been associated with the center of camera matrix. Cameras settings are presented in Fig. 2. Images of template from the Camera 1 and the Camera 2 are presented in the Figs. 3 and 4.

Coordinates of all 192 points from the template are read in pixels with usage of algorithm of image processing [8] implemented in C# language in Microsoft Visual C# 2010 Express environment. From Edimax IC-7100P camera data sheet, can be read, that size of the pixel is equal to $2.8 \times 10^{-3} \times 2.8 \times 10^{-3}$ mm and focal length $f_c = 5.01$ mm. After multiplication of coordinates in pixels by the pixel size, we obtain coordinates of points read from the image in mm, in $x_c y_c$-system. Coordinates of points in reference system xy are easy to determine. These points are placed in the corners of 8 by 8 mm square.

Position and orientation of camera system $x_c y_c z_c$ with respect to reference system are described by the homogeneous matrix $\mathbf{T}_c$ of transformation, like in the Eq. (1).

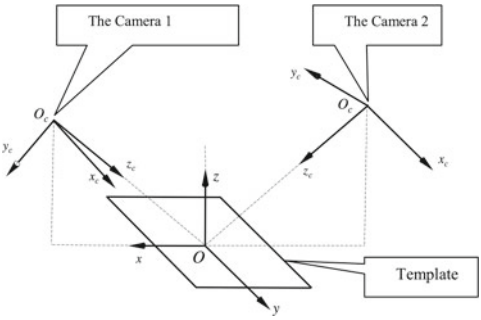


Fig. 2 Cameras settings

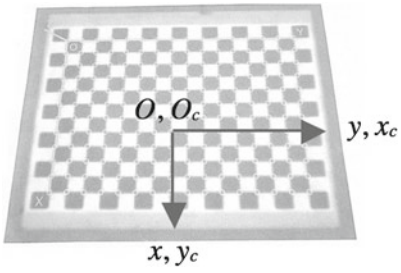
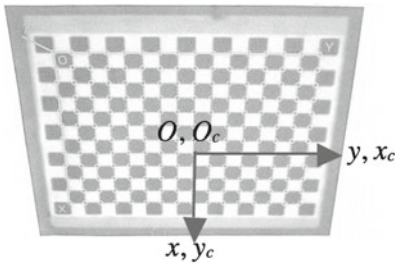


Fig. 3 Image 1

Fig. 4 Image 2



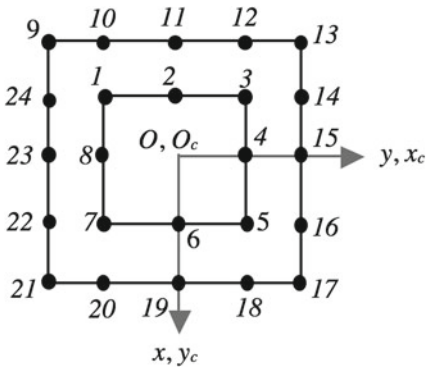
$$\mathbf{T}_c = Trans(d_x, d_y, d_z)\, Rot(z, \gamma)\, Rot(y, \beta)\, Rot(x, \alpha). \tag{1}$$

It is notation of successive transformations with respect to reference system xyz. Mentioned transformations are: rotation about axis x by angle α, rotation about axis y by angle β, rotation about axis z by angle γ, displacement d_z along axis z, displacement d_y along axis y, displacement d_x along axis x [3, 6].

These coordinates are determined by *Camera* algorithm [7]. Input parameters of these algorithms are coordinates α, β, γ, d_x, d_y, d_z; coordinates ${}^c z_A$, ${}^c z_B$, ${}^c z_C$ of three points A, B, C from template in the system $x_c y_c z_c$; coordinates ${}^c x_{Ac}$, ${}^c y_{Ac}$, ${}^c x_{Bc}$, ${}^c y_{Bc}$, ${}^c x_{Cc}$, ${}^c y_{Cc}$ of points A, B, C in the system $x_c y_c z_c$ (read from camera); coordinates x_A, y_A, z_A, x_B, y_B, z_B, x_C, y_C, z_C of points A, B, C in the system xyz; focal length f_c; and accuracy of calculations *delta*.

Calculations of coordinates of cameras from Fig. 2 was made by means of programs *kalibr1B* and *kalibr2B*. Each of these programs created 5056 sets of points A, B, C, next calculated the camera coordinates for each of these sets, and finally averaged these coordinates. For each camera these calculations lasted about 10 s. These sets were created from the 24 points lying beyond the beginning of coordinate system O_c, but closest to this beginning were chosen. These points are shown in Fig. 5. Read coordinates of these points have small errors caused by optical distortions. These points are located on the squares with side length of 16 and 8 mm, which centre approximately coincide with point O_c. Coordinates $z_A = z_B = z_C = 0$ mm.

Fig. 5 The 24 points from which 5056 sets were created



For initial values of input parameters α, β, γ, d_x, d_y, d_z, ${}^c z_A$, ${}^c z_B$, ${}^c z_C$, x_A, y_A, z_A, x_B, y_B, z_B, x_C, y_C, z_C calculated roughly using geometrical dependencies and ${}^c x_{Ac}$, ${}^c y_{Ac}$, ${}^c x_{Bc}$, ${}^c y_{Bc}$, ${}^c x_{Cc}$, ${}^c y_{Cc}$ read from camera, camera coordinates α, β, γ, d_x, d_y, d_z were calculated with accuracy $delta = 10^{-6}$ mm [7].

Equation (2a) describes averaged coordinates of the Camera 1 from Fig. 2 and Eq. (2b) describes matrix $\mathbf{T}_c$.

$$\alpha = 210.4624°,\ \beta = 0.7818°,\ \gamma = 89.9822°,\ d_x = 162.8905\,\text{mm},$$
$$d_y = 3.8324\,\text{mm},\ d_z = 277.1421\,\text{mm}, \tag{2a}$$

$$\mathbf{T}_c = \begin{bmatrix} -0.0003 & 0.8620 & -0.5070 & 162.8905 \\ 0.9999 & -0.0072 & -0.0116 & 3.8324 \\ -0.0136 & -0.5069 & -0.8619 & 277.1421 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2b}$$

Equation (3a) describes averaged coordinates of the Camera 2 from Fig. 2 and Eq. (3b) describes matrix $\mathbf{T}_c$.

$$\alpha = 144.3522°,\ \beta = -0.0617°,\ \gamma = 90.6657°,\ d_x = -200.9648\,\text{mm},$$
$$d_y = -3.1109\,\text{mm},\ d_z = 282.7646\,\text{mm}, \tag{3a}$$

$$\mathbf{T}_c = \begin{bmatrix} -0.0116 & 0.8126 & 0.5828 & -200.9648 \\ 0.9999 & 0.0088 & 0.0076 & -3.1109 \\ 0.0011 & 0.5828 & -0.8126 & 282.7646 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3b}$$

3 The Optical Distortions Errors

For simplification, rows and columns are introduced. Rows are consist of points, lying on lines, which are parallel to axis y on Fig. 1. Each row consist of 16 points. Number of rows is equal to 12, according to Fig. 1. Columns are consist of points, lying on lines, which are parallel to axis x on Fig. 1. Number of columns is equal to 16. P_{ij} is the point of ith row and jth column. Coordinates ${}^c x_c(i, j)$ and ${}^c y_c(i, j)$ of image of points P_{ij} of template, read from the camera system $x_c y_c$, have errors $\Delta_* {}^c x_c(i, j)$ and $\Delta_* {}^c y_c(i, j)$, caused by optical distortions. These errors can be calculated from mathematical description of distortions by means of coefficients k_1, k_2, k_3, p_1 and p_2 [1, 5]. Equations (4a) and (4b) describes these errors. Errors $\Delta {}^c x_c(i, j)$ and $\Delta {}^c y_c(i, j)$ can be determine from coordinates read from camera. These errors are described by Eq. (4c).

$$\Delta_* {}^c x_c(i,j) = {}^c x_{ci}(i,j)[k_1 {}^c r_{ci}(i,j)^2 + k_2 {}^c r_{ci}(i,j)^4 + k_3 {}^c r_{ci}(i,j)^6] + \\ + 2p_1 \cdot {}^c x_{ci}(i,j) \cdot {}^c y_{ci}(i,j) + p_2[{}^c r_{ci}(i,j)^2 + 2 \cdot {}^c x_{ci}(i,j)^2], \tag{4a}$$

$$\Delta_* {}^c y_c(i,j) = {}^c y_{ci}(i,j)[k_1 {}^c r_{ci}(i,j)^2 + k_2 {}^c r_{ci}(i,j)^4 + k_3 {}^c r_{ci}(i,j)^6] + \\ + 2p_2 \cdot {}^c x_{ci}(i,j) \cdot {}^c y_{ci}(i,j) + p_1[{}^c r_{ci}(i,j)^2 + 2 \cdot {}^c y_{ci}(i,j)^2], \tag{4b}$$

$$\Delta {}^c x_c(i,j) = {}^c x_c(i,j) - {}^c x_{ci}(i,j), \qquad \Delta {}^c y_c = {}^c y_c(i,j) - {}^c y_{ci}(i,j), \\ {}^c r_{ci}(i,j)^2 = {}^c x_{ci}(i,j)^2 + {}^c y_{ci}(i,j)^2. \tag{4c}$$

In Eqs. (4a)–(4c) occurs ideal coordinates of points ${}^c x_{ci}(i,j)$ and ${}^c y_{ci}(i,j)$, with no optical distortions. These coefficients can be calculated from homogeneous form $\mathbf{r}(i,j)$ of vector that describes point P_{ij} in reference system. We can note that form as follows:

$$\mathbf{r}(i,j) = \begin{bmatrix} x(i,j) \\ y(i,j) \\ z(i,j) \\ 1 \end{bmatrix} \tag{5}$$

Coordinates of point P_{ij} in reference system, that occurs in Eq. (5), can be note using indexes as follows: $x(i,j) = (i-7) \cdot 8\,\text{mm}$, $y(i,j) = (j-9) \cdot 8\,\text{mm}$. Point P_{79} is the origin O of the reference system. All points are lying on the plane xy, so $z(i,j) = 0$. Since camera matrix $\mathbf{T}_c$ of transformation is known, calculation of homogeneous form ${}^c\mathbf{r}(i,j)$ of vector that describes point P_{ij} in the camera system $x_c y_c z_c$ can be performed.

$$\mathbf{r}(i,j) = \begin{bmatrix} x(i,j) \\ y(i,j) \\ z(i,j) \\ 1 \end{bmatrix} = \mathbf{T}_c \, {}^c\mathbf{r} \rightarrow {}^c\mathbf{r}(i,j) = \begin{bmatrix} {}^c x(i,j) \\ {}^c y(i,j) \\ {}^c z(i,j) \\ 1 \end{bmatrix} = \mathbf{T}_c^{-1} \begin{bmatrix} x(i,j) \\ y(i,j) \\ z(i,j) \\ 1 \end{bmatrix}. \tag{6}$$

Coordinate ${}^c x_{ci}(i,j)$ can be obtained from coordinate ${}^c x(i,j)$ calculated from Eq. (6). From geometrical dependences, shown in the Fig. 5 results dependence (7a) that describes coordinate ${}^c x_{ci}(i,j)$ (Fig. 6).

$$\frac{{}^c x_{ci}(i,j)}{f_c} = \frac{-{}^c x(i,j)}{{}^c z(i,j) - f_c} \rightarrow {}^c x_{ci}(i,j) = -\frac{{}^c x(i,j)}{\frac{{}^c z(i,j)}{f_c} - 1} \tag{7a}$$

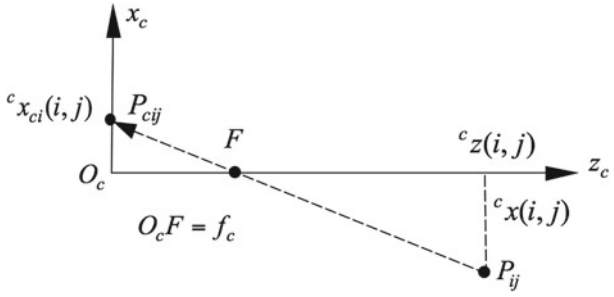


Fig. 6 The coordinates x_c of point P_{ij} and its image P_{cij}

Using similar geometrical dependencies the formula (7b) can be derived.

$$ {}^c y_{ci}(i,\, j) = -\frac{{}^c y(i,\, j)}{\frac{{}^c z(i,j)}{f_c} - 1} \tag{7b} $$

Using Eqs. (6), (7a) and (7b) errors $\Delta\, {}^c x_c(i,\, j)$, $\Delta\, {}^c y_c(i,\, j)$ and ${}^c r_{ci}(i,\, j)^2$ occurring in Eqs. (4a)–(4c) can be calculated. Since these values are known, it allows to apply Eqs. (4a) and (4b) for calculation of coefficients k_1, k_2, k_3, p_1 and p_2. If following sum is created

$$ S = \sum_{i=1}^{12} \sum_{j=1}^{16} \{[\Delta\, {}^c x_c(i,\, j) - \Delta_*\, {}^c x_c(i,\, j)]^2 + [\Delta\, {}^c y_c(i,\, j) - \Delta_*\, {}^c y_c(i,\, j)]^2\}, $$

unknown coefficients can be calculated by using minimally square method. Results of these calculations are presented by expressions (8a)–(9d). Calculations were made for two cameras from the Fig. 2.
For the Camera 1:

$$ k_1 = -0.0033\,\mathrm{mm}^{-2},\ k_2 \div k_3 = 0,\ p_1 \div p_2 = 0; \tag{8a} $$

$$ k_1 = -0.0050\,\mathrm{mm}^{-2},\ k_2 = 0.0085\,\mathrm{mm}^{-4},\ k_3 = 0,\ p_1 \div p_2 = 0; \tag{8b} $$

$$ k_1 = 0.0087\,\mathrm{mm}^{-2},\ k_2 = -0.0170\,\mathrm{mm}^{-4},\ k_3 = 0.0044\,\mathrm{mm}^{-6},\ p_1 \div p_2 = 0; \tag{8c} $$

$$ k_1 = 0.0091\,\mathrm{mm}^{-2},\ k_2 = -0.0190\,\mathrm{mm}^{-4},\ k_3 = 0.0053\,\mathrm{mm}^{-6}, $$
$$ p_1 = 0.0015\,\mathrm{mm}^{-2},\ p_2 = 0.0003\,\mathrm{mm}^{-2}. \tag{8d} $$

For the Camera 2:

$$k_1 = -0.0063\,\mathrm{mm}^{-2},\ k_2 \div k_3 = 0,\ p_1 \div p_2 = 0; \tag{9a}$$

$$k_1 = -0.0015\,\mathrm{mm}^{-2},\ k_2 = -0.0053\,\mathrm{mm}^{-4},\ k_3 = 0,\ p_1 \div p_2 = 0; \tag{9b}$$

$$k_1 = -0.0026\,\mathrm{mm}^{-2},\ k_2 = -0.0027\,\mathrm{mm}^{-4},\ k_3 = -0.0014\,\mathrm{mm}^{-6},\ p_1 \div p_2 = 0; \tag{9c}$$

$$k_1 = -0.0004\,\mathrm{mm}^{-2},\ k_2 = -0.0060\,\mathrm{mm}^{-4},\ k_3 = 0.0045\,\mathrm{mm}^{-6},\ p_1 = -0.0039\,\mathrm{mm}^{-2},\ p_2 = 0.0003\,\mathrm{mm}^{-2}. \tag{9d}$$

Equations (8a) and (9a) present error description using one coefficient; Eqs. (8b) and (9b)—using two coefficients; Eqs. (8c) and (9c)—using three coefficients; Eqs. (8d) and (9d)—using five coefficients.

4 The Calculation Coordinates Errors

Coefficients described by Eqs. (8a)–(9d) can be applied to calculate errors $\Delta_* {}^c x_c(i, j)$ and $\Delta_* {}^c y_c(i, j)$, described by Eqs. (4a)–(4b). After calculating these errors correction of coordinates of points ${}^c x_c(i, j)$ and ${}^c y_c(i, j)$ (read from camera matrix in the coordinate system $x_c y_c$) can be made. By correction it means subtraction of errors $\Delta_* {}^c x_c(i, j)$ and $\Delta_* {}^c y_c(i, j)$ from the coordinates ${}^c x_c(i, j)$ and ${}^c y_c(i, j)$. Coordinates after this correction are note by ${}^c x_{ccor}(i, j)$ and ${}^c y_{ccor}(i, j)$. Equation (10) describes corrected coordinates.

$${}^c x_{ccor}(i, j) = {}^c x_c(i, j) - \Delta_* {}^c x_c(i, j),\ {}^c y_{ccor}(i, j) = {}^c y_c(i, j) - \Delta_* {}^c y_c(i, j). \tag{10}$$

From corrected coordinates ${}^c x_{ccor}(i, j)$ and ${}^c y_{ccor}(i, j)$ of two cameras, coordinates $x_{cor}(i, j)$, $y_{cor}(i, j)$ and $z_{cor}(i, j)$ of points P_{ijcor} in the reference system xyz can be calculated. These coordinates are shown in the Fig. 7. In order to calculate coordinates $x_{cor}(i, j)$, $y_{cor}(i, j)$ and $z_{cor}(i, j)$, coordinates $x_{ccor}(i, j)$, $y_{ccor}(i, j)$, $z_{ccor}(i, j)$ of point P_{cijcor} and x_F, y_F, z_F of focal F in reference system are necessary. Equations (11) and (12) describes that coordinates by means of the coordinates ${}^c x_{ccor1}(i, j)$, ${}^c y_{ccor1}(i, j)$, ${}^c x_{ccor2}(i, j)$ and ${}^c y_{ccor2}(i, j)$. The ${}^c x_{ccor1}(i, j)$, ${}^c y_{ccor1}(i, j)$ are coordinates of Camera 1, and ${}^c x_{ccor2}(i, j)$, ${}^c y_{ccor2}(i, j)$—coordinates of Camera 2 from Fig. 2. The matrices $\mathbf{T}_c$, and others coordinates, and lengths f of the two cameras were marked similarly.

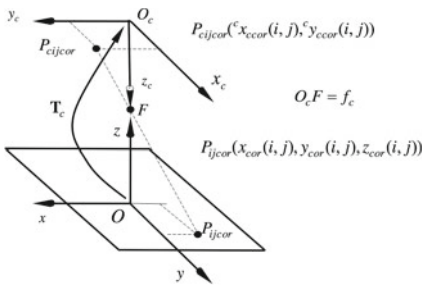


Fig. 7 The coordinates x of points P_{ijcor} and P_{cijcor}

$$\begin{bmatrix} x_{ccor1}(i,j) \\ y_{ccor1}(i,j) \\ z_{ccor1}(i,j) \\ 1 \end{bmatrix} = \mathbf{T}_{c1} \begin{bmatrix} {}^c x_{ccor1}(i,j) \\ {}^c y_{ccor1}(i,j) \\ 0 \\ 1 \end{bmatrix},$$
$$\begin{bmatrix} x_{ccor2}(i,j) \\ y_{ccor2}(i,j) \\ z_{ccor2}(i,j) \\ 1 \end{bmatrix} = \mathbf{T}_{c2} \begin{bmatrix} {}^c x_{ccor2}(i,j) \\ {}^c y_{ccor2}(i,j) \\ 0 \\ 1 \end{bmatrix}. \tag{11}$$

$$\begin{bmatrix} x_{F1} \\ y_{F1} \\ z_{F1} \\ 1 \end{bmatrix} = \mathbf{T}_{c1} \begin{bmatrix} 0 \\ 0 \\ f_{c1} \\ 1 \end{bmatrix}, \begin{bmatrix} x_{F2} \\ y_{F2} \\ z_{F2} \\ 1 \end{bmatrix} = \mathbf{T}_{c2} \begin{bmatrix} 0 \\ 0 \\ f_{c2} \\ 1 \end{bmatrix}. \tag{12}$$

Equation (13) describes straight line connecting points $P_{cijcor1}$, $F1$ and P_{ijcor}.

$$\frac{x_{cor}(i,j) - x_{ccor1}(i,j)}{x_{F1} - x_{ccor1}(i,j)} = \frac{y_{cor}(i,j) - y_{ccor1}(i,j)}{y_{F1} - y_{ccor1}(i,j)} = \\ = \frac{z_{cor}(i,j) - z_{ccor1}(i,j)}{z_{F1} - z_{ccor1}(i,j)}. \tag{13}$$

Equation (14) describes straight line connecting points $P_{cijcor2}$, $F2$ and P_{ijcor}.

$$\frac{x_{cor}(i,j) - x_{ccor2}(i,j)}{x_{F2} - x_{ccor2}(i,j)} = \frac{y_{cor}(i,j) - y_{ccor2}(i,j)}{y_{F2} - y_{ccor2}(i,j)} = \\ = \frac{z_{cor}(i,j) - z_{ccor2}(i,j)}{z_{F2} - z_{ccor2}(i,j)}. \tag{14}$$

From Eqs. (13) and (14) we can created six systems of three equations with three unknown coordinates $x_{cor}(i,j)$, $y_{cor}(i,j)$ and $z_{cor}(,j)$. To solve these systems the program *odl0* was written. Accuracy of calculation coordinates of points in the template can be describe by the maximum absolute values of distance differences

Table 1 Maximum values of $\Delta r(i, j)$ result from the distortion model

n	0	1 (Eqs. 8a and 9a)	2 (Eqs. 8b and 9b)	3 (Eqs. 8c and 9c)	5 (Eqs. 8d and 9d)
i	12	12	12	12	12
j	2	2	2	2	16
max $\Delta r(i, j)$	1.2342 mm	1.2005 mm	1.1465 mm	1.1459 mm	1.0767 mm

$\Delta r(i, j) = \sqrt{\Delta x(i, j)^2 + \Delta y(i, j)^2 + \Delta z(i, j)^2}$, where $\Delta x(i, j) = |x(i, j) - x_{cor}(i, j)|$, $\Delta y(i, j) = |y(i, j) - y_{cor}(i, j)|$, and $\Delta z(i, j) = |z(i, j) - z_{cor}(i, j)|$. As a reminder: $x(i, j) = (i - 7) \cdot 8\,\text{mm}$, $y(i, j) = (j - 9) \cdot 8\,\text{mm}$, $z(i, j) = 0$. Results of calculation of maximal values $\Delta r(i, j)$ for these two cameras are presented below in Table 1. In the table n is number of coefficients distortions describing by Eqs. (8a)–(9d).

For $n = 0$ calculations were done without corrections, i.e. for coordinates ${}^c x_{ccor}(i, j)$ and ${}^c y_{ccor}(i, j)$ respectively equal to ${}^c x_c(i, j)$ and ${}^c y_c(i, j)$. It is easy to observe, that coordinates calculation accuracy increases with the increasing number n of the optical error model coefficients. The greatest error appears when no any coefficient were accounted ($n = 0$). On the other hand, the smallest errors are obtained when all five coefficients k_1, k_2, k_3, p_1 and p_2 are applied.

From calculations for coordinates ${}^c x_{ccor}(i, j)$ and ${}^c y_{ccor}(i, j)$ respectively equal to ideal coordinates ${}^c x_{ci}(i, j)$ and ${}^c y_{ci}(i, j)$, computed from Eqs. (6), (7a) and (7b), results $max\,\Delta r(i, j) = max\,\Delta r(7, 10) = 2.9754 \cdot 10^{-12}$ mm! So small error indicates that the mathematical model of the optical distortion (4a) and (4b) is poorly correct for local nature of these distortions. This indicates the possibility of a greater reduction of the calculation error by taking into account their local character.

In the Sect. 5 a method for the better reducing errors, based on their local character, is presented.

5 The Interpolation Method of the Coordinates Calculations Error Decreasing

From the fourth point results a very small calculation errors of the coordinates $x_{cor}(i, j)$, $y_{cor}(i, j)$ and $z_{cor}(i, j)$ for the ideal coordinates ${}^c x_{ci}(i, j)$ and ${}^c y_{ci}(i, j)$. We can calculate the ideal coordinates for each point of the template. Therefore, to calculate the position of each point on the template is very simple, because for each of these points can be calculate the ideal coordinates from Eqs. (6), (7a) to (7b).

So far, we considered only the coordinates ${}^c x_c(i, j)$ and ${}^c y_c(i, j)$ of image points located in the corners of the tetragon $P_1P_2P_3P_4$ (see Fig. 8), and squares $P_1P_2P_3P_4$ of the template (see Fig. 9). In general, the points P_s can lie outside of these corners, e.g. within these tetragons. Then we can read from the camera only ${}^c x_c$ and ${}^c y_c$ coordinates of these points. Coordinates of these points in the reference system we

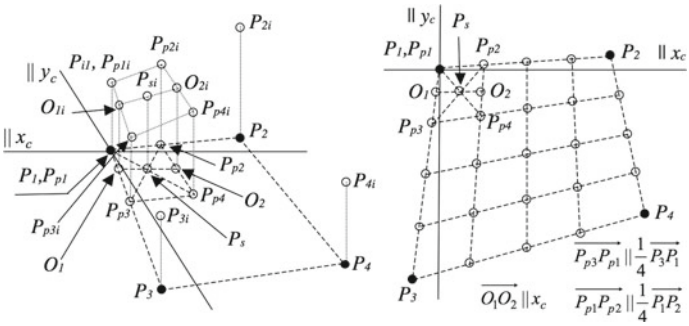


Fig. 8 Illustration of the interpolation of ideal coordinates $P_s\,P_{si}$

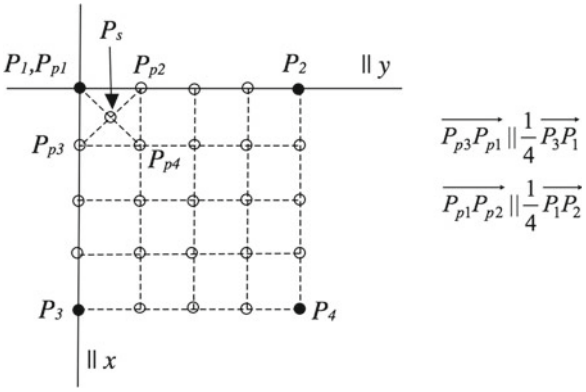


Fig. 9 Illustration of the additional points on template

do not know, therefore the calculating of the corresponding ideal coordinates is impossible.

The local nature of the cameras optical error for points P_s we consider using interpolation over the surface $x_c y_c$ of the ideal coordinates ${}^c x_{ci}$ and ${}^c y_{ci}$ camera coordinate systems. Are interpolated the ideal coordinates because they give very little calculation errors of coordinates in the reference system xyz. Figure 8 shows the interpolation. The points P_s and $P_{p1} \div P_{p4}$ from Fig. 8 correspond to the same points from Fig. 9. In the Fig. 8 the ideal coordinates are indicated in the form of vertical sections. The section $P_1 P_{1i}$ is the ideal coordinate ${}^c x_{ci}(i, j)$ or ${}^c y_{ci}(i, j)$ of point $P_1 = P_{ij}(i, j)$. Likely sections $P_2 P_{2i}$, $P_3 P_{3i}$, $P_4 P_{4i}$ are respectively equal to: ${}^c x_{ci}(i, j + 1)$ or ${}^c y_{ci}(i, j + 1)$ of point $P_2 = P_{ij}(i, j + 1)$, ${}^c x_{ci}(i + 1, j)$ or ${}^c y_{ci}(i + 1, j)$ of point $P_3 = P_{ij}(i + 1, j)$, ${}^c x_{ci}(i + 1, j + 1)$ or ${}^c y_{ci}(i + 1, j + 1)$ of point $P_4 = P_{ij}(i + 1, j + 1)$.

To increase the accuracy of interpolation was used program *podwoj* that creates points P_p, N-times more concentrated than the template points P_{ij} from Fig. 1. In the Figs. 8 and 9 is shown the division for $N = 4$. Next for points P_p the ideal coordinates were calculated by means of the program *podwoj*. These coordinates are illustrated in the Fig. 8 by means of the vertical sections $P_{p1} P_{p1i}$, $P_{p2} P_{p2i}$, etc.

To describe the sets of calculated points P_p it was applied the matrix $\mathbf{P}_p$. Components (i, j) of this matrix describes a point of ith row and jth column. Rows are consist of points, lying on lines, which are parallel to axis y in Fig. 9. For number of the division $N = 4$ each row consist of 61 points, number of rows is equal to 45. Columns are consist of points, lying on lines, which are parallel to axis x in Fig. 9.

The interpolation of ideal coordinates of points P_s, lying in the middle of tetragon $P_{p1}P_{p2}P_{p3}P_{p4}$ were used to analysis of the position coordinates calculation errors (see Fig. 8). The coordinates of the points P_s, were calculated by program *punktxx*. To describe the sets of calculated points P_s it was applied the matrix $\mathbf{P}_s$. Components (i, j) of this matrix describes a point of ith row and jth column. Rows are consist of points, lying on lines, which are parallel to axis x_c in Fig. 8. For number of the division $N = 4$ each row consist of 60 points, number of rows is equal to 44. Columns are consist of points, lying on lines, which are parallel to axis y_c in Fig. 8. The ideal coordinates P_sP_{si} of points were calculated by means of interpolation on the base of forth values $P_{p1}P_{p1i}$, $P_{p2}P_{p2i}$, $P_{p3}P_{p3i}$ and $P_{p4}P_{p4i}$. The program *doklB1B2* was calculated of maximal errors $\Delta r(i, j)$ for the dividing number $N = 1, 2, 4, 8, 16$.

Now the errors $\Delta r(i, j) = \sqrt{\Delta x(i, j)^2 + \Delta y(i, j)^2 + \Delta z(i, j)^2)}$, where $\Delta x(i, j) = |x_{Ps}(i, j) - x_{Pscor}(i, j)|$, $\Delta y(i, j) = |y_{Ps}(i, j) - y_{Pscor}(i, j)|$, and $\Delta z(i, j) = |z_{Ps}(i, j) - z_{Pscor}(, j)|$. $x_{Ps}(i, j) \div z_{Ps}(i, j)$ are coordinates of points P_s illustrated in Fig. 9. $x_{Pscor}(i, j) \div z_{Pscor}(i, j)$ are coordinates of points P_{scor} computed by program *odl0*. Results of these calculations are presented in the Table 2. The third row of Table 2 contains the maximum of the computation time t_p, designated by the program *czas*. It is computation time of the coordinates of single point P_s in the reference system xyz. Table 2 shows the lowest value of the maximum error $\Delta r(i, j) = 0.0771$ mm for the number of division $N = 16$. Practical industrial robot positioning accuracy is in the order $0.1 - 0.5$ mm. Thus, the achieved calculation accuracy is sufficient for the needs of the industrial robot control.

The analysis of the data in Table 2 show that we can still reduce the value of this error by increasing the number of dividing N. This is accompanied by increasing the calculation time t_p of single points coordinates. Increasing this number cause decrease the length of the sides of the tetragon $P_{p1}P_{p2}P_{p3}P_{p4}$. We can't reduce the length to the side length of a single pixel on the sensor of the camera. For $N = 16$ minimum length of sides of the tetragon $P_{p1}P_{p2}P_{p3}P_{p4}$ is equal to approximately

Table 2 Maximum values of $\Delta r(i, j)$ result from interpolation of the ideal coordinates

N	1	2	4	8	16
size($\mathbf{P}_s$)	[11 × 15]	[22 × 30]	[44 × 60]	[88 × 120]	[176 × 240]
size($\mathbf{P}_p$)	[12 × 16]	[23 × 31]	[45 × 61]	[89 × 121]	[177 × 241]
maxt_p	0.0312 s	0.0624 s	0.1092 s	0.3588 s	1.3416 s
i	1	1	1	1	1
j	12	23	46	91	180
max $\Delta r(i, j)$	1.1418 mm	0.5974 mm	0.3028 mm	0.1530 mm	0.0771 mm

two lengths of the pixel side. Therefore, further increasing N to 32 would reduce the size to the size of the pixel squares. Then the ideal coordinate interpolation could be incorrect.

From the comparison of the minimum value of the maximum error calculations presented in Table 1 ($\Delta r(i, j) = 1.0767$ mm) and Table 2 ($\Delta r(i, j) = 0.0771$ mm) results greater accuracy of calculations using interpolation of the ideal coordinates, proposed at this point.

6 Summary

The research presented in this chapter shows an analysis of the effectiveness of the decreasing of the calculation position error of the template points. These calculations are based on visual information of two cameras. The application of a mathematical model of optical errors cameras are less effective than interpolation method of ideal coordinates proposed here.

The studies shows, that accuracy of calculation the coordinates of points on the template using cameras depends on the optical errors description. The more coefficients from k_1, k_2, k_3, p_1, p_2 is accounted in errors description $\Delta_*{}^c x_c(i, j)$ and $\Delta_*{}^c y_c(i, j)$, described by Eqs. (4a) and (4b), the smaller are coordinates errors $\Delta r(i, j)$ of points determined in the space xyz.

To achieve greater accuracy of the calculations it can be used the interpolation method of the ideal coordinates proposed in this work. This method allows to achieve increasing the accuracy of calculations by increasing the number of divisions N.

Studies presented here should be treated as preliminary step of designing vision system with two cameras. The minimum value of the maximum error $\Delta r(i, j)$ determines the accuracy of the system. The accuracy of the system of two cameras presented in this work is determined by $0.0771 \approx 0.1$ mm (see Table 2). The accuracy is sufficient for the needs of the industrial robot control.

Acknowledgments The research presented here were funded by the Silesian University of Technology grant BK-227/RAu1/2015/2.

References

1. Beyer, H.A.: Geometric and radiometric analysis of a CCD-camera based photogrammetric close-range system. Ph.D. thesis, Diss. Techn. Wiss. ETH Zurich, Zurich (1992)
2. Chesi, G., Garulli, A., Vicino, A., Cipolla, R.: On the Estimation of the Fundamental Matrix: A Convex Approach to Constrained Least-Squares. ECCV 2000. LNCS, vol. 1842. Springer, Berlin (2000)
3. Craig, J.: Introduction to Robotics. Addison Wesley, Reading (1986)
4. Golub, G.H., Van Loan, C.F.: Matrix Computations. Johns Hopkins University Press, Baltimore (1996)
5. Kiełczewski, M.: The Calibraction of Camera, http://etacar.put.poznan.pl/marcin.Kielczewski

6. Szkodny, T.: Foundation of Robotics. Silesian University of Technology Publication Company, Gliwice (2012)
7. Szkodny, T.: Calculation of the location coordinates of an object observed by a camera. In: Gruca, D.A., Czachórski, T., Kozielski, S. (eds.) Man-Machine Interactions 3. AISC, vol. 242, pp. 139–151. Springer, Cham (2014)
8. Szkodny, T., Meller, A., Palka, K.: Accuracy of determining the coordinates of points observed by camera. In: Zhang, X., Liu, H., Chen, Z., Wang, N. (eds.) Intelligent Robotics and Applications. LNCS, vol. 8918, pp. 273–284. Springer (2014)

Part IV
Bio-Data Analysis and Mining

Parameter Estimation in Systems Biology Models by Using Extended Kalman Filter

Michal Capinski and Andrzej Polanski

Abstract Models in systems biology, which reflect complex dynamic biological phenomena are most often described as ordinary differential equations (ODE). Characteristic properties of these differential equations is nonlinearity and large size (number of state variables). These models also contain large numbers of unknown parameters. So the main challenge in developing models in systems biology is estimation of numerous unknown parameters in nonlinear differential equations. There are already numerous approaches to parameter estimation in systems biology models. However, main difficulties speed of convergence and multiple minima (multiple solutions) are still obstacles in achieving solutions of sufficient efficiency. In this chapter we propose a new approach based on combination of extended Kalman filtering dynamical optimization with spline approximation of solutions to ODE, for parameter estimation in systems biology models. We present the main idea and we show comparisons to some published results.

Keywords Parameter estimation · Systems biology · Extended kalman filter · Dynamic · Optimization · Spline approximation

1 Introduction

Parameter estimation in nonlinear, high dimensional systems is the major problem in systems biology. There is a lot references in literature [8] or [9], which developed a range of optimization methods including steepest descent gradient search techniques and methods suitable for global optimization, such as simulated annealing, genetic programming or traditional nonlinear programming (NLP) algorithms, such as sequential quadratic programming (SQP), sequential penalty function, the trust region approach and etc. [2, 12]. An interesting and efficient method has been proposed in the chapter [16] were spline approximation of the solution to analyzed

M. Capinski (✉) · A. Polanski
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: michal_capinski@interia.pl

A. Polanski
e-mail: andrzej.polanski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_16

dynamical model was used. Depending on the complexity of the model differential equations the authors used either linear or non-linear programming as optimization tool combined with the parameter estimator. Two examples were shown, enzyme kinetic system and a cell cycle model.

Several authors have recently developed sequential estimation methods for state and parameter estimation from systems biology models, described as state-space models [6, 10, 13–15]. These approaches use parametric methods based on Kalman filtering optimization and non-parametric particle filtering methods.

In the article we present the idea of combination EKF as optimization method with of spline approximation of measured data. This idea allows for obtaining good compromise between computational efficiency and robustness to multiple minima. We show comparisons of different methods of parameter estimation in systems biology.

2 Classical Methods

We construct an error function E_D (1) that quantifies the difference between a model with parameters α and the data, then we use optimization method that finds the value of α that minimizes $E_D(\alpha)$ e.g.

$$E_D = \sum_{i=1}^{N} \|x(t_i, \alpha) - x_{data_i}\|^2 \tag{1}$$

where $x(t_i, \alpha)$—numerically obtained an approximate solution.

If $E_D(\alpha)$ has only a few, local minima apart from the global minimum, then we use methods that iteratively step downhill, such as the Nelder–Mead simplex method [11] or the Levenberg-Marquardt method [7].

3 Method Based on Combining Spline Theory with Linear Programming (LP)

In many bio-system models, $f(x, \theta)$ is autonomous system and linear in θ as follows [16]:

$$\dot{x}(t) = \Phi(x(t))\theta, x(t_0) = x_0 \tag{2}$$

where $\Phi(x) \in R^{n\times k}$ is a matrix and its elements are a function of the state x.

Replace $x(t)$ by the B-spline approximation $\hat{x}(t)$ and integrate (2) yield

$$\tilde{x}(\theta, t_j) = \left(\int_{t=t_0}^{t_j} \Phi(\hat{x}(t))dt \right) \cdot \hat{\theta} + \hat{x}_0 = \hat{\Psi}_j \cdot \hat{\theta} + \hat{x}_0 \tag{3}$$

where $\hat{\Psi}_j$—represents the transition matrix.

Optimization problem can be transformed into the following augmented optimization problem with introducing the slack variables α as follows:

$$P : \min_{\hat{\theta},\alpha}\left\{\sum_{i,j}\omega_{i,j}\cdot\alpha_{i,j}\right\}$$
$$s.t. = \begin{cases} (i) & -\alpha_{ij} \le \gamma_i(t_j) - \tilde{x}_i(\theta, t_j), \\ (ii) & \alpha_{ij} \ge \gamma_i(t_j) - \tilde{x}_i(\theta, t_j), \\ (iii) & \tilde{x}_i(\theta, t_j) = \hat{\Psi}_j \cdot \hat{\theta} + \hat{x}_0, \\ (iv) & \alpha_{ij} \ge 0 \\ (v) & \theta_L \le \hat{\theta} \le \theta_U \end{cases} \tag{4}$$

where:
$\hat{\theta} \in R^k$ is the set of parameters to be estimated,
$\gamma_i(t_j)$—measured data,
θ_L and θ_U—are simple structural constraints such as the parameter's upper/lower bounds.

It is a Linear Programming (LP) problem with variable α, θ, which is a convex problem with a wealth of fast and efficient routines available [1].

4 Method Based on Spline Theory with Nonlinear Programming (NLP)

Biological pathway dynamics can be modeled by the following continuous ODEs [16]:

$$\dot{x}(t) = f(x(t), u(t), \theta), \quad x(t_0) = x_0, y(t) = g(x(t)) + \eta(t); \tag{5}$$

where:
$x \in R^n$ is the system's state vector,
$\theta \in R^k$ is the system's parameter vector,
$u(t) \in R^p$ is system's input,
$y \in R^m$ denotes the measured data subject to a Gaussian white noise $\eta(t)\ N(0, \sigma^2)$,
x_0 is the initial state,
$f(\cdot)$ is a set of nonlinear transition functions describing the dynamical properties of biological system,
$g(\cdot)$ represents a measurement function.

The parameter estimation problem of nonlinear dynamical systems described in (1) formulated with as a nonlinear programming problem (NLP) P_0 with differential-algebraic constraints [16]:

$$P_0 : \min_{\hat{\theta},p} \sum_{j=0}^{N} (\gamma(t_j) - \hat{\gamma}((t_j), \hat{\theta})^T w_j (\gamma(t_j) - \hat{\gamma}(t_j, \hat{\theta})$$

$$s.t. = \begin{cases} (i) \quad \hat{x_i}(t_j) = {b_i}^T(t_j) \cdot p_i, \\ (ii) \quad \dot{\hat{x}_i}(t_j) = \dot{b_i}^T(t_j) \cdot p_i, \\ \\ (iii) \quad ||\dot{\hat{x}_i}(t_j) - f(\hat{x}(t_j), u(t), \hat{\theta})||_2^2 = 0, \\ \\ (iv) \quad \hat{\gamma}(t_i) = g(\hat{x}(t_j)) \\ (v) \quad C_{eq}(\hat{x}(t_j), \dot{\hat{x}}(t_j), \hat{\theta}) = 0, \\ (vi) \quad C_{eq}(\hat{x}(t_j), \dot{\hat{x}}(t_j), \hat{\theta}) \leq 0, \\ (vii) \quad \theta_L \leq \hat{\theta} \leq \theta_U, \\ \qquad i = 1, 2, \ldots, n, \quad j = 0, 1, \ldots, N. \end{cases} \tag{6}$$

where:
$\gamma_i(t_j)$—measured data,
$\hat{x_i}(t_j)$—is B-spline approximation of estimated variable $\hat{x}$,
b_i—B-spline basis functions,
p_i—is the weighting coefficient of B-spline,
$\dot{b_i}$—the set of the derivatives of the basis functions,
$C_{eq}(\hat{x}(t_j), \dot{\hat{x}}(t_j), \hat{\theta})$—algebraic equation constraints,
θ_L and θ_U—are simple structural constraints such as the parameter's upper/lower bounds.

5 Extended Kalman Filter

The well-known Kalman filter [4] is an optimal state estimator in the case of linear, Gaussian systems. His variations is the extended Kalman filter (EKF), which is a recursive state estimation algorithm for noisy nonlinear systems of the form [3]:

$$x_k = f_k(x_{k-1}, u_k) + w_k \tag{7}$$

$$z_k = h_k(x_k) + v_k \tag{8}$$

where: $x_k \in R^n$—denotes the state, $z_k \in R^q$ are the outputs, $u_k \in R^p$—are the inputs. The noise processes w_k and v_k are uncorrelated and zero mean with positive-definite covariances $Q_k > 0$ and $R_k > 0$.

The two-step Kalman filter is given by the prediction equation:

$$\begin{aligned} \hat{x}_k &= f_k(x_{k-1}, u_k) \\ P_k &= F_{k-1} P_{k-1} F_{k-1}^T + Q_{k-1} \end{aligned} \tag{9}$$

and innovation equations:

$$\begin{aligned} \hat{x}_k &= \hat{x}_k + G_k(z_k - h_k(\hat{x_k})) \\ K_k &= P_k H_k^T (H_k P_k H_k^T + R_k)^{-1} \\ P_k &= (I - K_k H_k) P_k \end{aligned} \tag{10}$$

where:
F_k—Jacobian of state transition,
H_k—Jacobian of observation matrices,
P_k—state covariance,
Q_k—process noise covariance,
R—measurement noise covariance.

6 Extended Kalman Filter with Spline Decomposition of Solutions ODE

The approach, which we propose in this paper is using extended Kalman filtering algorithm (7)–(10) as a nonlinear optimization engine for differential algebraic system (6). EKF becomes an optimizer for nonlinear programming problem (NLP) with spline approximation coefficients.

Nonlinear programming problem (NLP) P_0 with differential-algebraic constraints (6) can be reformulated to Lagrangian function [16]:

$$\begin{aligned} &L(\hat{\theta}, p, \lambda) = \\ &= \sum_{j=0}^{N} (\gamma(t_j) - \hat{\gamma}((t_j), \hat{\theta})^T w_j (\gamma(t_j) - \hat{\gamma}(t_j, \hat{\theta}) + \lambda \cdot ||\hat{\dot{x}_i}(t_j) - f(\hat{x}(t_j), u(t), \hat{\theta})||_2^2 \end{aligned} \tag{11}$$

where:
λ—is the Lagrange multiplier,
p—the weighting coefficients of B-spline,
$\hat{\theta}$—is the set of parameters to be estimated.

Let us denote the function returned in the right-hand side of (11) by

$$\begin{aligned} &\Phi_k(\hat{\theta}, p, \lambda) = \\ &= \min_{\hat{\theta}, p} \sum_{j=0}^{N} (\gamma(t_j) - \hat{\gamma}((t_j), \hat{\theta})^T w_j (\gamma(t_j) - \hat{\gamma}(t_j, \hat{\theta}) + \lambda \cdot ||\hat{\dot{x}_i}(t_j) - f(\hat{x}(t_j), u(t), \hat{\theta})||_2^2 \end{aligned} \tag{12}$$

Combining (11) and (12) with (7)–(10) leads to:

$$\begin{aligned} x_k &= [\hat{\theta}_{1k} \ldots \hat{\theta}_{nk}\ p_{1k} \ldots p_{mk}\ \lambda_k] \\ x_k &= \Phi_k(x_{k-1}, u_k) + w_k \\ z_k &= h_k(x_k) + v_k \end{aligned} \tag{13}$$

where:
x_k—estimation of k-state,
n—number of parameters,
m—number of the weighting coefficients of B-spline.

Prediction equation:

$$\begin{aligned} \hat{x}_k &= \Phi_k(x_{k-1}, u_k) \\ P_k &= F_{k-1} P_{k-1} F_{k-1}^T + Q_{k-1} \end{aligned} \tag{14}$$

Innovation equations:

$$\begin{aligned} \hat{x}_k &= \hat{x}_k + G_k(z_k - h_k(\hat{x}_k)) \\ K_k &= P_k H_k^T (H_k P_k H_k^T + R_k)^{-1} \\ P_k &= (I - K_k H_k) P_k \end{aligned} \tag{15}$$

Equations (13)–(15) are extended Kalman filtering recursions, which provide as an output solutions to parameter estimation problem.

7 Results

In this section we present comparisons of different methods for parameter estimation in systems biology models given by high dimensional ODEs. Comparisons are based on the dynamical model for molecular regulation in extracellular signal-regulated kinases (ERK) pathway. The model includes 11 ODEs and 11 unknown model parameters.

The scenario for comparisons includes analyses of time signals obtained by computer simulations in ERK pathway model with known values of parameters by different algorithms. Estimated values of parameters are compared to true values and serve for grading qualities of different approaches.

7.1 Test Model—RKIP Regulated ERK Pathway Model

The RKIP regulated ERK signaling pathway [5], as shown in Fig. 1, is a circle representing a state for the concentration of a protein, e.g. a circle with x1 denotes the concentration of the activated protein Raf-1. A rectangular bar contains kinetic

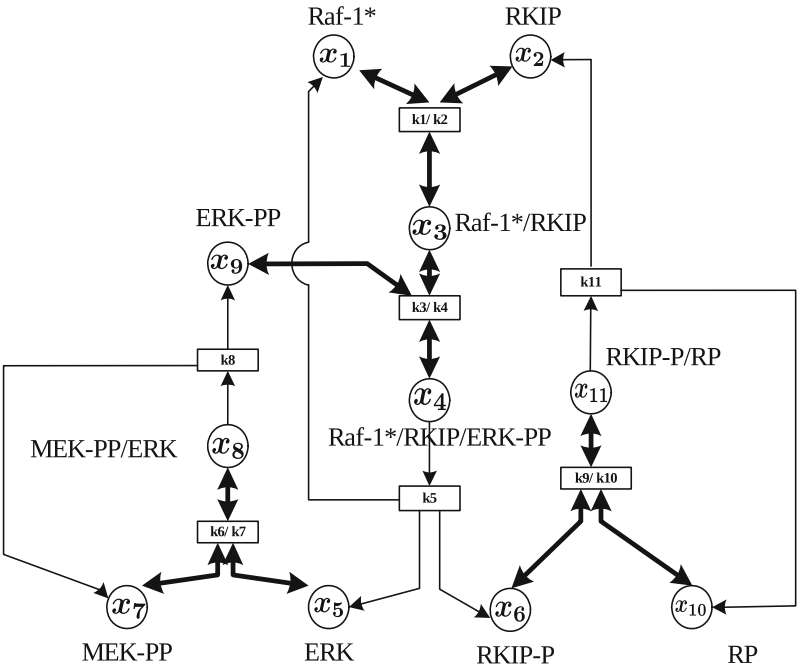


Fig. 1 Graphical representation of ERK signaling pathway regulated by RKIP

parameters of reaction which denote the reaction rates with respect to corresponding reactions and which should be estimated. The directed arc (arrows) connecting a circle and a bar represents a direction of a signal flow. The bi-directional thick arrows represent a association and a dissociation rate at same time. The thin unidirectional arrows represent a production rate of products.

The corresponding ODE model is shown as following:

$$
\begin{aligned}
\dot{x}_1 &= -k_1 x_1 x_2 + k_2 x_3 + k_5 x_4 \\
\dot{x}_2 &= -k_1 x_1 x_2 + k_2 x_3 + k_{11} x_{11} \\
\dot{x}_3 &= k_1 x_1 x_2 - k_2 x_3 - k_3 x_3 x_9 + k_4 x_4 \\
\dot{x}_4 &= k_3 x_3 x_9 - k_4 x_4 - k_5 x_4 \\
\dot{x}_5 &= k_5 x_4 - k_6 x_5 x_7 + k_7 x_8 \\
\dot{x}_6 &= k_5 x_4 - k_9 x_6 x_{10} + k_{10} x_{11} \\
\dot{x}_7 &= -k_6 x_5 x_7 + k_7 x_8 + k_8 x_8 \\
\dot{x}_8 &= k_6 x_5 x_7 - k_7 x_8 - k_8 x_8 \\
\dot{x}_9 &= -k_3 x_3 x_9 + k_4 x_4 + k_8 x_8 \\
\dot{x}_{10} &= -k_9 x_6 x_{10} + k_{10} x_{11} + k_{11} x_{11} \\
\dot{x}_{11} &= k_9 x_6 x_{10} - k_{10} x_{11} - k_{11} x_{11}
\end{aligned}
\tag{16}
$$

Table 1 Statistical results of parameter estimation of RKIP regulated ERK pathway model without noise

Mean estimation ± standard deviation				
Param.	True value	Classical methods	Splines with LP method	EKF with NLP method
k_1	0.53	0.5227 ± 3.4892e−10	0.5300 ± 7.3052e−11	0.5300 ± 4.2162e−11
k_2	0.0072	0.0078 ± 5.7698e−07	0.0068 ± 2.7542e−07	0.0078 ± 1.3512e−08
k_3	0.625	0.6615 ± 4.7160e−11	0.6247 ± 3.7939e−11	0.6249 ± 7.9981e−11
k_4	0.00245	0.0020 ± 1.9784e−08	0.0022±1.9387e−08	0.0024±1.3177e−08
k_5	0.0315	0.0312 ± 3.8692e−09	0.0315 ± 4.0889e−09	0.0314 ± 8.3144e−09
k_6	0.8	0.7922 ± 8.9341e−10	0.8000 ± 2.0321e−10	0.8000 ± 1.3479e−10
k_7	0.0075	0.0073 ± 2.9893e−06	0.0075 ± 5.5388e−06	0.0075 ± 4.7521e−06
k_8	0.071	0.0711 ± 8.2174e−07	0.0705 ± 3.8577e−07	0.0709 ± 1.4231e−07
k_9	0.92	0.9207 ± 1.0241e−08	0.9196 ± 1.2879e−09	0.9200 ± 1.9244e−09
k_{10}	0.00122	0.0014 ± 6.1887e−06	0.0012 ± 1.8735e−06	0.0012 ± 5.1912e−06
k_{11}	0.87	0.8622 ± 1.3784e−06	0.8721 ± 1.6347e−06	0.8699 ± 2.0361e−06
J	–	0 ± 2.8219e−16	0 ± 4.3535e−15	0 ± 8.0816e−15

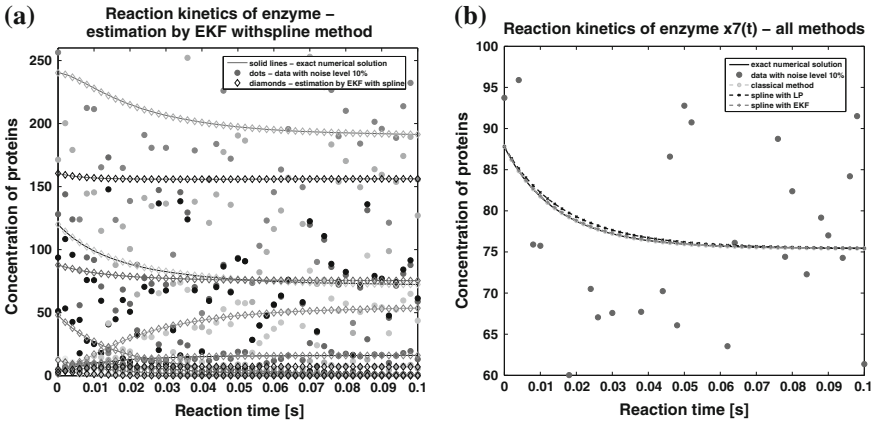


Fig. 2 **a** Reaction kinetics of enzymes—estimation by EKF with spline method, **b** Reaction kinetics of one enzyme $x_7(t)$—estimation by all methods

7.2 Results of Estimation Parameters of RKIP by All Methods

To quantify the fitness of the estimated model, the following relative squared error (RSE) measure J was employed:

$$J = \frac{1}{N \cdot n} \sum_{i=1}^{N} \sum_{j=0}^{N} \left(\frac{\hat{x}(t_j) - x_j(t_j)}{x_j(t_j)} \right)^2 \tag{17}$$

The RSE obtain in condition without noise is almost zero for all methods (Table 1). They differ by time of simulation,the shortest was for B-spline with LP.

The EKF algorithm is sensitive for start point,when the initial guess of the state and/or state parameters are very close to the true values (Fig. 2). Convergence is not achieved when the initial conditions differ significantly from the corresponding true values.

8 Conclusion

The advantage of the new method is a faster convergence in terms of iterations compared to classical methods, although the cost of each iteration is higher and convergence depends on initial conditions.

Results obtained by B-spline approximation and LP are comparable to these obtain by EKF with NLP method. All the mean estimated parameters were within a relative tolerance better than 5 %, besides a 10 % observation noise level. EKF as optimizer is interesting for further study and other modifications.

Acknowledgments This chapter was partially financially supported by the NCN Opus grant UMO-2011/01/B/ST6/06868 to AP. Computations were performed with the use of the infrastructure provided by the NCBIR POIG.02.03.01-24-099/13 grant: GCONiI—Upper-Silesian Center for Scientific Computations.

References

1. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
2. Fletcher, R.: Practical Methods of Optimization. Wiley, New York (2013)
3. Humpherys, J., Redd, P., West, J.: A fresh look at the kalman filter. SIAM Rev. **54**(4), 801–823 (2012)
4. Kalman, R.E.: A new approach to linear filtering and prediction problems. J. Fluids Eng. **82**(1), 35–45 (1960)
5. Kwang-Hyun, C., Sung-Young, S., Hyun-Woo, K., Olaf, W., Brian, M.F., Walter, K.: Mathematical modeling of the influence of RKIP on the ERK signaling pathway. In: Priami, Corrado (ed.) CMSB 2003. LNCS, vol. 2602, pp. 127–141. Springer, Heidelberg (2003)
6. Lillacci, G., Khammash, M.: Parameter estimation and model selection in computational biology. PLOS Comput. Biol. **6**(3), e1000696 (2010)
7. Marquardt, D.W.: An algorithm for least-squares estimation of nonlinear parameters. J. Soc. Ind. Appl. Math. **11**(2), 431–441 (1963)
8. Mendes, P., Kell, D.: Non-linear optimization of biochemical pathways: applications to metabolic engineering and parameter estimation. Bioinformatics **14**(10), 869–883 (1998)
9. Moles, C.G., Mendes, P., Banga, J.R.: Parameter estimation in biochemical pathways: a comparison of global optimization methods. Genome Res. **13**(11), 2467–2474 (2003)
10. Nakamura, K., Yoshida, R., Nagasaki, M., Miyano, S., Higuchi, T.: Parameter estimation of in silico biological pathways with particle filtering towards a petascale computing. PSB 2009, pp. 227–238. Fairmont Orchid, Hawaii (2009)

11. Nelder, J.A., Mead, R.: A simplex method for function minimization. Comput. J. **7**(4), 308–313 (1965)
12. Powell, M.: A fast algorithm for nonlinearly constrained optimization calculations. In: Watson, G. (ed.) Numerical Analysis. LNM, vol. 630, pp. 144–157. Springer, Berlin (1978)
13. Quach, M., Brunel, N., d'Alché Buc, F.: Estimating parameters and hidden variables in non-linear state-space models based on odes for biological networks inference. Bioinformatics **23**(23), 3209–3216 (2007)
14. Sun, X., Jin, L., Xiong, M.: Extended kalman filter for estimation of parameters in nonlinear state-space models of biochemical networks. PLOS One **3**(11), e3758 (2008)
15. Yang, J., Kadirkamanathan, V., Billings, S.A.: In vivo intracellular metabolite dynamics estimation by sequential monte carlo filter. In: CIBCB 2007, pp. 387–394. Honolulu (2007)
16. Zhan, C., Yeung, L.F.: Parameter estimation in systems biology models using spline approximation. BMC Syst. Biol. **5**(1), 14 (2011)

Nucleotide Composition Based Measurement Bias in High Throughput Gene Expression Studies

Roman Jaksik, Wojciech Bensz and Jaroslaw Smieja

Abstract High throughput gene expression profiling methods suffer from various sources of measurement bias inherent to the experimental procedures used. Most of the commonly used data standardization methods, designed to reduce the sample-to-sample variability of technical origin, do not account for probe- or transcript-specific effects. However, the efficiency of RNA isolation, cDNA synthesis and amplification does depend on the percentage of GC nucleotides in the transcript sequences and therefore constitutes a strong bias for the analysis of gene expression data. This work is focused on analysis of how and to what extent GC-content bias of oligonucleotide microarray probes affects the measurement data. We propose a mechanism explaining this phenomenon, the implications of GC-content bias for differentially expressed genes (DEGs) detection, and propose a new data standardization method, which by using sample-specific background intensity estimation and LOESS regression, allows to counteract the described effects.

Keywords Microarray probes sequences · High throughput gene expression studies

1 Introduction

Oligonucleotide microarrays are one of the most common tools used for measuring how gene expression levels change under different physical and/or chemical conditions. Microarray data analysis helps to identify the most important genes in a specific cellular response mechanism or to find a characteristic gene expression pattern of a particular disease. Such analysis requires appropriate statistical processing of the data In order to separate signal changes induced by the experimental factors from the

R. Jaksik (✉) · W. Bensz · J. Smieja
Institute of Automatic Control, Silesian University of Technology, Gliwice, Poland
e-mail: roman.jaksik@polsl.pl

W. Bensz
e-mail: wojciech.bensz@polsl.pl

J. Smieja
e-mail: jaroslaw.smieja@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_17

background changes caused by inaccuracies of the measurements and errors of the methods used. Challenges identified at that stage of data processing led to studies of compatibility of different microarray platforms, [2, 3, 6, 7, 14, 15] involving the standardization of protocols and data analysis pipelines [8, 9]. Appropriate selection of a statistical method for microarray processing is a prominent subject of scientific discussion although many data analysis related issues remain unresolved. A typical microarray is manufactured using special photolithographic technique used to attach hundreds of thousands of different single stranded oligonucleotide sequences on the surface of a glass slide. Oligonucleotides complementary to characteristic fragments of known DNA or RNA sequences are arranged in groups termed probes [11]. Quantification of the levels of transcripts in a sample is achieved by spreading it on the surface of an microarray and providing the conditions allowing hybridization of the transcripts to their complementary probes and measuring the amount of material hybridized to specific probes using a fluorescence-based method. In our previous study [10] we have shown that the variance of signal intensity between probes with various nucleotide sequences that target a single gene is substantially larger than the variance between signal of identical probes between biological replicates. This suggests a very strong influence of probe specific factors on the estimation of gene expression levels. Among various known factors, GC composition of the probe is of the highest importance, and due to large variations in the percentage of GC nucleotides between probes on a single microarray it can be the primary source of measurement inconsistencies [10]. Signal levels of probes with varying nucleotide composition may be influenced by the efficiency of hybridization process, which depends strongly on sequence melting temperatures related to its GC content. Probes of higher GC content are prone to non-specific binding as GC pairs contain stronger triple hydrogen bonds. On the other hand, probes with low GC content form weaker bounds during hybridization, which might be broken upon elution of non-specifically attached cRNA. Since GC content of the probe set reflects GC content of the transcripts, the factors that differentially compromise efficiency of cRNA synthesis can also influence signal levels for probes with varying GC content. Differences in signal levels between probes of varying GC content are assumed to reflect presence of mRNA of particular nucleotide composition in the investigated mRNA pool. However, they might as well be resulting from one of the three mRNA properties unveiling at different stages of the experimental protocol:

- Isolation—mRNA of higher GC content is more stable, hence for high mRNA degradation level during the isolation process, mRNA pool may become enriched in these sequences
- Amplification—cDNA richer in GC is being transcribed at a slower rate, due to smaller polymerase efficiency [1]
- Hybridization—probes of high GC content create stronger bonds with cRNA of interest, additionally influencing non-specific hybridization levels [13].

All aforementioned factors might lead to differences in signal levels for probes of differing GC content that are independent of the transcript levels. Since those factors characterize the most basic technical differences between microarrays, they might

lead to significantly different signal levels measured for transcripts that in fact do not differ between examined samples (particularly those of extreme GC nucleotides proportions).

2 Results

2.1 Differences Between Signal Levels of Probes of Different GC Content

Boxplots in Fig. 1 show median values for signal levels of probes in respect to their GC content. The analysis was conducted for two datasets Affy-HuGene and MAQC-133P2, based on microarray platforms of distinct properties. Affy-HuGene is a benchmark dataset, provided by the manufacturer, that includes 33 HuGene-1_0-st microarrays (with exon-specific probes), loaded with RNA samples isolated from different tissues. MAQC-133P2 is a data set obtained using HG-U133_Plus_2 microarrays (with 3′-UTR specific probes), that originates from the MicroArray Quality Control (MAQC) project [14]. Differences between panels A and B shown on Fig. 2 may result from significant differences between microarray platforms used, however the differences may also be experiment-specific. Figure 2 shows median values for signal of probes with different GC content, calculated for HG-U133A microarrays. Data from all 28202 microarrays, originating from various experiments, is plotted, without any preprocessing (no data standardization algorithms used). Microarrays were divided into 12 groups based on their average total fluorescence level. Median plot of raw expression in respect to probe GC content was calculated for each group. The figure shows that the relation between average fluorescence level of an entire array

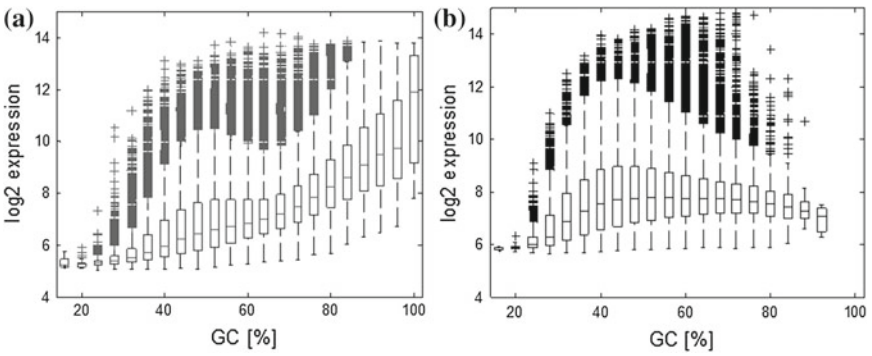


Fig. 1 Boxplots showing signal levels of probes of different GC content (mean non-processed probe signal in logarithmic scale). **a** Data from Affy-HuGene experiment, **b** Data from MAQC-133P2 experiment. Plus signs denote outliers

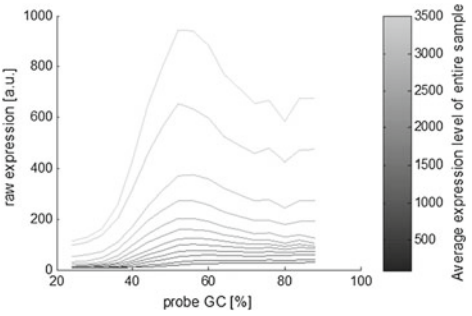


Fig. 2 Dependency between probe GC content and microarray signal level for microarray groups of varying mean total fluorescence (averaged data from 28202 samples analyzed on HG-U133A platform)

and the signal level of probes with different GC content varies significantly. Additionally the relation between signal of high GC content probes ($\tilde{80}$ %) and probes with an intermediate GC content ($\tilde{50}$ %) changes, depending on the average fluorescence level of the array. This is most likely a result of technical issues rather than effect of actual change of transcripts in the biological material. Average fluorescence of all microarray probes depends on the time of amplification and efficiency of cRNA labeling. Change in the shape of the curve observed together with decrease in average fluorescence level of an microarray suggests influence of some additional factors.

2.2 Influence of GC-Content-Related Bias on the Results of a Microarray Experiment

Figure 2 shows that changes in the average fluorescence intensity can affect the ratio of signal intensity between probes with high and low GC content. This might significantly affect the interpretation of microarray data since substantial differences in the average signal level between samples that originate from a single experiment can be observed very often. In the experiments conducted in our laboratory (in particular, the one labeled E01_Me45) we observe correlation of expression level changes between pairs of compared samples and the GC content of corresponding genes. We hypothesize that this phenomenon is a result of the GC-content bias. The experiment E01_Me45 was designed to test the influence of ionizing radiation on the changes of gene expression, at various time points, after treating Me45 cells (human melanoma) with 4 Gy of ionizing radiation, relative to the control (untreated) cells. The results show correlation between the change of expression level after irradiation and the GC content of the transcripts, which we are unable to justify with the biological mechanisms involved in the radiation stress response.

Additionally, the correlation of expression level changes and transcript GC content is stronger than correlation with the mean GC content of the probe set (see Table 1). It suggests that signals of single probes are not only influenced by their properties, but the GC content related properties of the transcripts as well. In order to check

Table 1 Spearman's correlation coefficients for the expression fold change after the irradiation, and the GC contents of the transcriptome (r_1), or the probe set (r_2)

Time after irradiation	1 h	12 h	24 h
r_1	−0.440	−0.508	−0.181
r_2	−0.425	−0.479	−0.121
Correlation coefficient equality test (p-value)	0.046	3.46e − 05	3.30e − 11

All the correlation coefficients are statistically significant (p-value $< 10^{-9}$). Correlation coefficients were obtained from the E01_Me45 experiment data

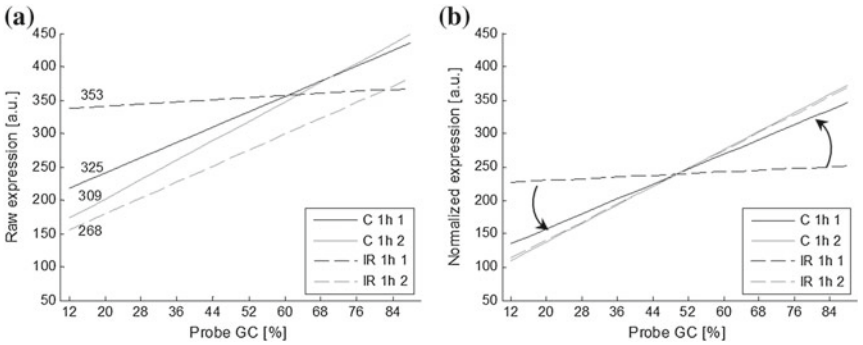


Fig. 3 Regression line calculated for the GC content and signal levels of probes of 4 separate samples from the E01 experiment. Chart **a** shows raw data (the numbers at the lines are mean fluorescence levels of particular microarrays). Chart **b** shows data after the RMA background correction and quantile normalization

if a similar phenomenon influences the signal levels in the E01_Me45 experiment, characterized by strong correlation of expression level changes after the irradiation with GC contents of the transcripts, normalization algorithms were compared by their influence on signal level ratio for probes of varying GC content. In order to simplify interpretation of the results, instead of the median value of signals obtained from probes that differ in the GC content (used in Fig. 2), the regression line fitted to data obtained for all probes of the microarray was used. Figure 3a shows the differences between ratios of signals originating from probes of differing GC content (for non-standardized data), expressed as the regression line slope coefficient. After data preprocessing, involving RMA background correction and quantile normalization, the distributions of signals of all probes became equal. Nevertheless, differences of strengths of signals coming from probes of high and low GC contents were not compensated, as the normalization process leads only to a shift of the regression line, but does not change its slope (Fig. 3b). The slope, resulting from technical differences between samples, is most drastically different for the pair of samples from the first biological repetition (C_1h_1 and IR_1h_1, angle between the corresponding regression lines is marked with black arrows). Differences in the signals ratio of high- and low-GC probes, leads to large difference of expression measured for corresponding

transcripts in samples obtained from control cells (C) and irradiated cells (IR), that cannot be compensated by the quantile normalization. This can lead to an incorrect estimation of expression changes between the compared samples, for GC-rich and GC-poor transcripts, resulting in an increased false-positive rate of algorithms used to detect DEGs.

2.3 Correction of the Probe Sets Signal Levels for Removing GC-Content-Related Bias

Differences in the nucleotide composition are usually compensated by adding or removing nucleotides from either 5′ or 3′-end of the oligonucleotide, affecting the entire sequence length. However, Affymetrix microarrays due to the specificity of their design utilize probes with a constant length of 25 nucleotides significantly limiting the possibilities of GC content manipulation. Additionally the manufacturer did not compensate for varying probe GC content assuming that microarrays would be used to compare expression of the same genes between biological samples only, and not to compare expression of different genes or transcripts. Strong disproportions in GC contents of the probes may render the expression levels of different genes, measured on the same microarray, incomparable. Properties of the probes are assumed invariant from one microarray to another, hence, the measured signals should be comparable if a normalization algorithm cancelling the differences of technical origin between individual microarrays is used. Figure 1 indicates, that additional signal level correction for probe GC content bias is necessary in order to obtain comparable expression levels for different probes varying in the GC content. On the other hand, such correction might significantly alter obtained expression values, potentially leading to loss of information, if the physical phenomena behind it are not carefully characterized. Since this paper is concerned with identification of DEGs only, compensation of differences originating from the GC contents of individual probes of the same microarray is not required. Instead the goal is to compensate the differences between identical probes from different microarrays. The GC bias correction was performed according to a similar method proposed by Benjamini in 2012 [4] for a related problem in the deep sequencing experiments. The method is based on data scaling that utilizes coefficients of the curve, fitted to data from each individual sample, using LOESS regression (LOcally Estimated Scatterplot Smoothing). The regression curve is very close to the median value of probes signal presented in Fig. 2 with the advantage of not being as heavily dependent on single measurements that contribute to the median of very low and high CG content probes. Scaling is performed after regression curve fitting to each individual sample by applying (1):

$$\hat{S}_{m,p} = S_{m,p} \frac{K_m(GC_p)}{\hat{K}(GC_p)} \tag{1}$$

where $\hat{S}_{m,p}$ is a signal of probe p, acquired from microarray m after the correction, $S_{m,p}$ signal before the correction, $K_m(GC_p)$ is a coefficient for probes of particular GC nucleotides number obtained through regression curve fitting to the sample data m. $\hat{K}$ is the mean coefficient calculated from all the regression lines for N probes of the experiment (2):

$$\hat{K}(GC) = \frac{1}{N}\sum_{m=1}^{N} K_m(GC) \tag{2}$$

where GC is the number of G and C nucleotides in the analyzed probes (0–25).

The proposed microarray data processing algorithm denoted as csGC-RMA (corrected sample-based GC-RMA) consists of the following steps:

- Background correction based on a modified GC-RMA method, utilizing mismatch probes signals of each individual sample for the non-specific hybridization estimation
- Adjustment for imbalance of signals that originate from probes with varying GC content, based on a LOESS local regression and linear data transformation
- quantile normalization (common to all RMA algorithm modifications as well as for PLIER and FARMS methods)
- median polish summarization (used in all modifications of RMA algorithm).

Assessment of the correction quality was performed in two stages. First stage is based on the E01_Me45 experiment data, in which the magnitude of correlation coefficient between the changes of probe sets signal levels and the GC content of the transcripts was evaluated. The goal of the second stage is to assess the impact of the algorithm on the effectiveness of DEGs detection, based on two publicly available data sets, designed specifically for the evaluation of pre-processing algorithms. Graphs in Fig. 4 show the regression lines fitted to the data without any additional correction, processed with standard GC-RMA method (graph A) and after correction

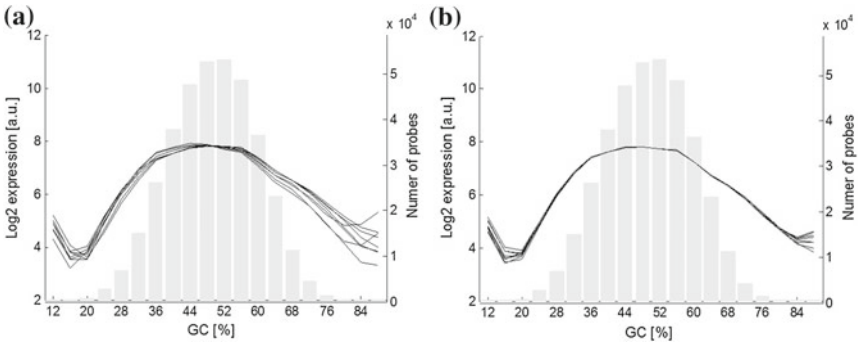


Fig. 4 Regression lines fitted to data processed with GC-RMA algorithm (**a**) and with rhe proposed csGC-RMA method (**b**). The *grey* histograms and the corresponding ordinate axis on the *right* depict the numbers of probes of specific GC content

Table 2 Spearman's correlation coefficient r_s between the transcript GC content and expression change after the irradiation, calculated for the E01_Me45 experiment data with and without applying the correction for the GC content of the probes

Time after irradiation	1 h	12 h	24 h
r_s without correction	−0.440	−0.508	−0.181
r_s with correction	−0.031	−0.084	−0.021

and processing with the csGC-RMA (graph B). Decreased correlation (Table 2) and good fitness of the regression lines, which is an obvious consequence of the applied scaling based on their shape, do not confirm the quality of the proposed method. Excessive flattening of the signal may lead to similar effect, although not avoiding artificial diminishing of the biological differences between samples. However the assessment of the impact of proposed method on detection of genes of interest in the E01 data set is not possible due to lack of the a priori knowledge of the gene expression alterations triggered by irradiation in Me45 cells. In order to assess sensitivity and specificity of the deg detection, two additional data sets were used, namely the GoldenSpike and the PlatinumSpike [5]. These two data sets, were obtained using known RNA mixture, enriched with more than ten additional transcripts, assumed to constitute the only (non-technical) source of observed variation in the sample pairs.

The PlatinumSpike contains similar number of augmented and diminished quantity transcripts, whereas the GoldenSpike, being the older data set, contains only augmented transcripts, hence the latter is inferior in terms of resemblance of the actual experimental data. For both data sets tested, the csGC-RMA method proved superior over the standard GC-RMA algorithm, allowing for both greater specificity and sensitivity of the DEG detection, particularly for the PlatinumSpike data set. The observed increase of area under the ROC curves is a result of decrease in the number of false positive hits (specificity gain) and simultaneous increase in the number of

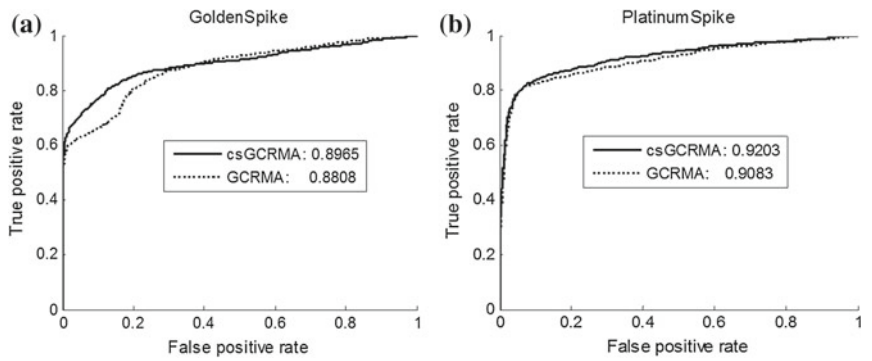


Fig. 5 ROC (Receiver Operating Characteristic) *curves* for the GoldenSpike data (**a**) and the PlatinumSpike data (**b**) after processing with standard GC-RMA and proposed csGC-RMA methods. Numeric values shown in the legends are the areas under corresponding *curves*

true positive hits (sensitivity gain). It should be noted that while application of the probe GC content correction or the GCRMA modification separately yields some gain in the detection quality, only the combination of the two allows for results this good (Fig. 5).

3 Discussion

Classical analysis of microarray experiments suggests that ionizing radiation influences expression levels of transcripts with varying GC content. The findings presented in this work, however, show that negative correlation of the GC transcript content and the change of expression level measured using a microarray may, at least partially, be attributed to technical issues related to sample processing. Existing methods of microarray data processing do not compensate for the observed differences between samples, resulting from the variations in signal level of AT or GC rich probes, in some cases even enhancing this effect. Similar problem concerns the deep sequencing data, but on a larger scale [4, 12]. The impact of deviations of average hybridization level for probes differing in the GC content has strong implications on virtually all possible analysis schemes:

- comparison of signal levels between multiple genes, measured using a single microarray,
- comparison of gene expression ratios across multiple samples (screening for markers using sample classification algorithms)
- comparison of gene expression changes across pairs of samples where the technical repetition plays a major role.

We have shown how nucleotide composition of probes affects their signal level, and how this process varies between individual microarrays. The discussed effect exists in various microarray platforms and is currently not compensated for by any of the commonly used data standardization algorithms. We also showed that the observed effect may negatively impact the process of DEG identification, by overestimating expression of transcripts with extreme probe GC contents. Moreover, it can significantly decrease the sensitivity of DEG detection due to increased variance of technical or biological repetitions. The proposed correction based on the probe GC content allows for better DEG identification, as shown using two exemplary data sets: GoldenSpike and PlatinumSpike. Application of this correction may in fact come with some risk, as the expression differences between all genes of high and low GC ratio are affected and artificially decreased. Finally, it is worth noting, that the correction is based on an assumption, that GC content of all genes with augmented or diminished expression should be similar, which cannot be biologically justified. Similar assumptions are required by other microarray pre-processing methods, for instance that the total number of up- and down-regulated genes should be similar. Such assumptions may be false in many situations, although they are necessary due to the current limitations of large scale gene expression studies.

Acknowledgments This work was supported by the Polish National Centre for Research and Development grant number POIG.02.03.01-00-040/13. Calculations were carried out using the computer cluster Ziemowit (http://ziemowit.hpc.polsl.pl) funded by the Silesian BIO-FARMA project No. POIG.02.01.00-00-166/08 in the Computational Biology and Bioinformatics Laboratory of the Biotechnology Centre in the Silesian University of Technology.

References

1. Arezi, B., Xing, W., Sorge, J.A., Hogrefe, H.H.: Amplification efficiency of thermostable dna polymerases. Anal. Biochem. **321**(2), 226–235 (2003)
2. Barnes, M., Freudenberg, J., Thompson, S., Aronow, B., Pavlidis, P.: Experimental comparison and cross-validation of the affymetrix and illumina gene expression analysis platforms. Nucleic Acids Res. **33**(18), 5914–5923 (2005)
3. Beekman, J.M., Boess, F., Hildebrand, H., Kalkuhl, A., Suter, L.: Gene expression analysis of the hepatotoxicant methapyrilene in primary rat hepatocytes: an interlaboratory study. Environ. Health Perspect. **114**(1), 92–99 (2006)
4. Benjamini, Y., Speed, T.P.: Summarizing and correcting the gc content bias in high-throughput sequencing. Nucleic Acids Res. **40**(10), e72 (2012)
5. Choe, S.E., Boutros, M., Michelson, A.M., Church, G.M., Halfon, M.S.: Preferred analysis methods for affymetrix genechips revealed by a wholly defined control dataset. Genome Biol. **6**(2), R16 (2005)
6. Dobbin, K.K., Beer, D.G., Meyerson, M., Yeatman, T.J., Gerald, W.L., et al.: Interlaboratory comparability study of cancer gene expression analysis using oligonucleotide microarrays. Clin. Cancer Res. **11**(2 Pt 1), 565–572 (2005)
7. Guo, L., Lobenhofer, E.K., Wang, C., Shippy, R., Harris, S.C., et al.: Rat toxicogenomic study reveals analytical consistency across microarray platforms. Nat. Biotechnol. **24**(9), 1162–1169 (2006)
8. Hockley, S.L., Mathijs, K., Staal, Y.C.M., Brewer, D., Giddings, I., van Delft, J.H.M., Phillips, D.H.: Interlaboratory and interplatform comparison of microarray gene expression analysis of HepG2 cells exposed to benzo(a)pyrene. OMICS **13**(2), 115–125 (2009)
9. Irizarry, R.A., Warren, D., Spencer, F., Kim, I.F., Biswal, S., et al.: Multiple-laboratory comparison of microarray platforms. Nat. Methods **2**(5), 345–350 (2005)
10. Jaksik, R., Marczyk, M., Polanska, J., Rzeszowska-Wolny, J.: Sources of high variance between probe signals in affymetrix short oligonucleotide microarrays. Sensors **14**(1), 532–548 (2013)
11. Pease, A.C., Solas, D., Sullivan, E.J., Cronin, M.T., Holmes, C.P., Fodor, S.P.: Light-generated oligonucleotide arrays for rapid dna sequence analysis. Proc. Natl. Acad. Sci. **91**(11), 5022–5026 (1994)
12. Risso, D., Schwartz, K., Sherlock, G., Dudoit, S.: GC-content normalization for RNA-Seq data. BMC Bioinform. **12**(1), 480 (2011)
13. Schuster, E.F., Blanc, E., Partridge, L., Thornton, J.M.: Estimation and correction of non-specific binding in a large-scale spike-in experiment. Genome Biol. **8**(6), R126 (2007)
14. Shi, L., Reid, L.H., Jones, W.D., Shippy, R., Warrington, J.A., et al.: The microarray quality control (maqc) project shows inter-and intraplatform reproducibility of gene expression measurements. Nat. Biotechnol. **24**(9), 1151–1161 (2006)
15. Shi, L., Tong, W., Fang, H., Scherf, U., Han, J., et al.: Cross-platform comparability of microarray technology: intra-platform consistency and appropriate data analysis procedures are essential. BMC Bioinform. **6**(Suppl 2), S12 (2005)

Application of a Morphological Similarity Measure to the Analysis of Shell Morphogenesis in Foraminifera

Maciej Komosinski, Agnieszka Mensfelt, Paweł Topa and Jarosław Tyszka

Abstract This work evaluates the genotype-to-phenotype mapping defined by one of the models of growth of foraminifera. Foraminifera are simple unicellular organisms with very diverse morphologies. To analyze the mapping, a morphological similarity measure is needed that compares 3D structures. One of the key components of the similarity estimation algorithm is Singular Value Decomposition (SVD). Since this algorithm is heavily used and its performance is important, four SVD implementations have been compared in this work. Distance matrices of the phenotypes obtained for equally distant genotypes were computed using the similarity measure. For the visualization of the phenotype space, multidimensional scaling techniques were used. Visual comparison of the genotype and the phenotype spaces revealed characteristics and potential weaknesses of the analyzed model of foraminifera growth, and demonstrated usefulness of the proposed approach.

Keywords Similarity · Morphology · Genotype · Phenotype · Foraminifera

M. Komosinski (✉) · A. Mensfelt
Institute of Computing Science, Poznan University of Technology, Poznan, Poland
e-mail: maciej.komosinski@cs.put.poznan.pl

A. Mensfelt
e-mail: agnieszka.mensfelt@cs.put.poznan.pl

P. Topa · J. Tyszka
Institute of Geological Sciences, Polish Academy of Sciences, Research Centre in Cracow, Kraków, Poland
e-mail: paweltopa@gmail.com

J. Tyszka
e-mail: ndtyszka@cyf-kr.edu.pl

P. Topa
Department of Computer Science, AGH University of Science and Technology, Kraków, Poland

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_18

1 Introduction

Applications of similarity measures for the analysis of three-dimensional constructs range from evolutionary design to artificial life and theoretical biology. Such measures are very useful when a large population of structures needs to be automatically compared. Estimating similarity allows one to classify morphologies, construct hierarchies of morphologies, discover clusters and investigate the correlation between phenotypes and fitness of individuals [5–7].

Modeling of organism morphogenesis benefits from the use of such a similarity measure as well. Although foraminifera are unicellular organisms whose shells (also called "tests") are usually smaller than 1 mm, they have very diverse morphology. Taxonomy of those organisms include over 10,000 living and fossil species, which are of great interest to biologists and micropalaeontologists. Foraminifera build shells consisting of one or more chambers. In the simplest case, a single chamber contains an opening called the aperture. Openings of subsequent chambers compose the communication line which is called the local communication path. There are two basic chamber morphologies in polythalamous (multichambered) foraminifera: globular and tubular. The models considered in this work are focused on foraminiferal shells composed of globular chambers. This new class of foraminifera is called Globothalamea and is based on results of molecular and morphogenetic modeling [14].

1.1 The Model of Foraminifera Morphogenesis

Models of growth of foraminifera are actively developed [17, 18, 20]. The model used in this work describes foraminiferal morphology with 7 parameters [8, 19] that determine the location and size of subsequent chambers [10]. The growth of a foraminifer starts from a single chamber for which the location of its center, O_0, is arbitrarily defined. The location of the aperture U_0 is calculated according to the local communication path length minimization principle: it is a point on the surface of the shell which is nearest to the center, O_0.

In the two-dimensional case, calculation of the new (ith) chamber center, O_i, starts with the determination of the reference axis, which is the line passing through both the previous and the current aperture. In the first step, O_i is located exactly on the current aperture U_{i-1}. It is then moved along the reference axis according to the translation factor parameter TF. Next, it is deviated from the reference axis according to the deflection angle $\Delta\phi$. The size of the new chamber is determined by the scaling factors K_x and K_y. In the three-dimensional case the procedure is similar, yet there are two additional parameters: the rotation angle $\Delta\beta$ and the scaling factor K_z.

After computing the location and the size of the new chamber, the new aperture can be found—again, it is a point on the surface of the new chamber with the shortest distance to the previous aperture. The new aperture cannot be located inside any previous chamber. The 7 parameters of this model of growth are enumerated in Table 1, and sample foraminiferal morphologies are shown in Fig. 1.

Table 1 The parameters comprising foraminifera genotype

Symbol	Name	Range from	Range to	Example in Fig. 1
N	Number of chambers	1	15	1, 2, 4, 6, 8, 9, 10
K_x	Scale in x	1.00	1.10	1.00
K_y	Scale in y	1.00	1.10	1.00
K_z	Scale in z	1.00	1.10	1.00
TF	Translation factor	−1.00	1.00	−0.02
$\Delta\phi$	Deflection angle	−3.14	3.14	0.64
$\Delta\beta$	Rotation angle	−3.14	3.14	0.72

Angles are expressed in radians. The TF range ensures that subsequent chambers are joined together

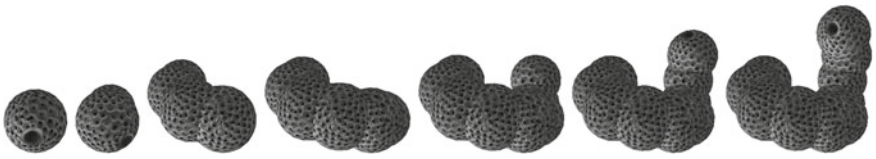


Fig. 1 A sample growth sequence of a foraminifer. The number of chambers increases from 1 to 10. The dark spot is the aperture. Values of growth parameters are shown in Table 1

The 7 parameters can be considered high-level genes, and the resulting 3D shell morphology corresponds to a phenotype. The process of morphogenesis (growing a phenotype based on a genotype) can be considered a mapping between a genotype and a phenotype [9]. While this model of growth has been introduced to mimic and simulate biological processes, it is important to be able to analyze the genotype-to-phenotype mapping formally—in terms of the relationship between the space of genotypes and the space of phenotypes. This is where similarity measures are required.

Measuring similarity of genotypes is straightforward in case of numbers, as long as we assume that the traditional meaning of similarity of values is reasonable (the lower the difference between values, the higher their similarity). To measure similarities between morphologies, we need a procedure that compares three-dimensional structures. The concept that describes the correspondence between differences in genotypes and phenotypes is called *locality* [16, Chap. 3]. High locality is obtained when neighboring genotypes are expressed as neighboring phenotypes. The locality has impact on the relative topology of both spaces, and in consequence, on optimization techniques and evolutionary processes.

2 Morphological Similarity Measure

2.1 The Algorithm

The similarity measure used to estimate differences in foraminiferal morphologies considers 3D structures as undirected graphs [5–7]. In case of foraminifera, each chamber constitutes one vertex, and all vertices are connected with edges producing a linear graph structure (Fig. 2). The similarity estimation algorithm consists of three main steps: alignment of the two structures that are compared, construction of the matching function, and calculation of the dissimilarity components. Two components are taken into account when computing distance between vertices: difference in vertex degree (*iDeg*) and geometrical distance (*iGeo*). The importance of each of the components can be adjusted using weights.

The approach taken for the alignment is based on the distribution of points in a three-dimensional space. To position the morphologies according to the distributions of vertices, the SVD transform [12] is used. It is applied separately to both structures. After the transform is computed, the center of the structure is located in the origin of the coordinate system. The axis with the highest variance becomes the first axis of the structure, and the axis with the second highest variance becomes the second axis of the structure. This method was proved to provide good alignment for the geometry of the structures [7].

The matching algorithm is a heuristic. Vertices in both structures are sorted by vertex degree in a descending order. The vertices with the same degree are grouped together. The procedure starts with groups of vertices having the highest degree in each structure. The algorithm tries to find a match (the least distant vertex from the other structure) for the vertices which are unmatched yet in both structures, starting from the vertices with the lowest indexes. When all of the vertices from a group are matched, the next group is taken into account. Once the matching function is constructed, the overall dissimilarity between the two compared structures can be determined.

It is desired for the dissimilarity measure to be a metric. For this purpose, it must satisfy non-negativity, identity of indiscernibles, symmetry and triangle inequality. The similarity measure outlined above always satisfies the first three conditions. The last condition—the triangle inequality—can be extremely rarely violated when the $iGeo$ component is considered [7].

Fig. 2 A linear graph (10 vertices and 9 edges) representing the 10-chamber foraminiferal morphology from Fig. 1

Table 2 Comparison of libraries/routines for SVD

Name	Type	Description
OpenCV	BSD license	Open-source library for computer vision [4]
Eigen	MPL2 license	Template C++ library [2]
NR	Commercial	Numerical recipes [13, 15]
MATLAB	Commercial	C++ interface for a computing environment [11]

2.2 Comparison of Performance of Four SVD Implementations

Since the SVD transform is frequently computed during estimation of morphological similarity, its performance is important. For this reason, we have compared four C++ implementations of SVD: two open-source libraries, a routine from Numerical Recipes, and the MATLAB® Math Library. Table 2 lists these libraries.

Figure 3 compares the performance of the four SVD implementations on various 3D structures. The performance was similar for two sets containing morphologies with the low average number of vertices (Fig. 3a, b); MATLAB and Eigen libraries were slightly slower. For the set with high average number of vertices, MATLAB library outperformed other libraries and the Eigen library was significantly slower than other libraries (Fig. 3c). Since foraminifera have simple morphologies that usually have no more than 20 vertices, both NR and OpenCV were considered best choices in terms of speed and ease of use.

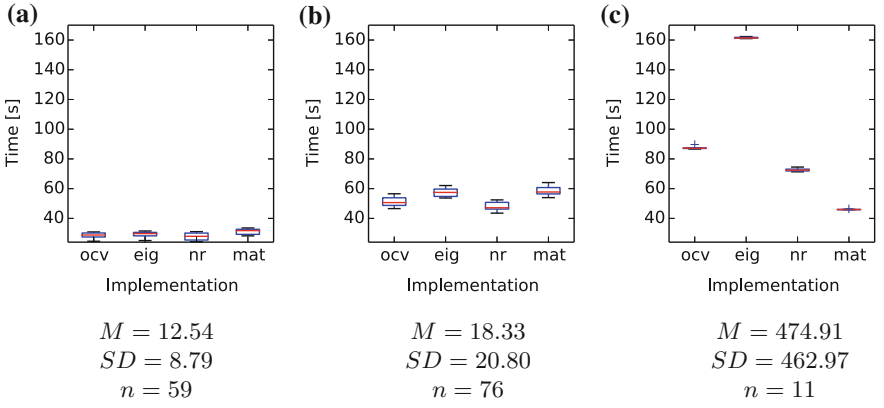


Fig. 3 Time of calculating the full distance matrix ($n \times n$) for three sets of morphologies of increasing complexity. *M* and *SD* are the mean and the standard deviation of the number of vertices in a set containing *n* morphologies. All tests were performed on Intel Core i7-4770 with 8 GB of RAM running Windows 7. The one-thread program was compiled with Visual Studio 2014

3 Application of Similarity Measure to the Analysis of the Genotype-Phenotype Mapping in Foraminifera

To evaluate the characteristics of the genotype-to-phenotype mapping in foraminifera morphogenesis, the relationship between genotype and phenotype spaces was analyzed. 5 out of 7 genes were kept constant, while the remaining two were varied from the minimal to the maximal value (Table 3). For the purpose of the analysis of each pair of parameter values, $32 \times 32 = 1024$ genotypes and

Table 3 Parameter values used in the analysis

Visualization	N	K_x	K_y	K_z	TF	$\Delta\phi$	$\Delta\beta$
Figure 4	5	1	1	1	-0.1	$[-3.14; 3.14]$	$[-3.14; 3.14]$
Figure 5	5	1	1	1	$[-0.99; 0.99]$	$[-3.14; 3.14]$	0
Figure 6	5	1	1	1	$[-0.99; 0.99]$	0	$[-3.14; 3.14]$

Constant values are indicated by a single number. Ranges of the varied parameter values are shown in square brackets

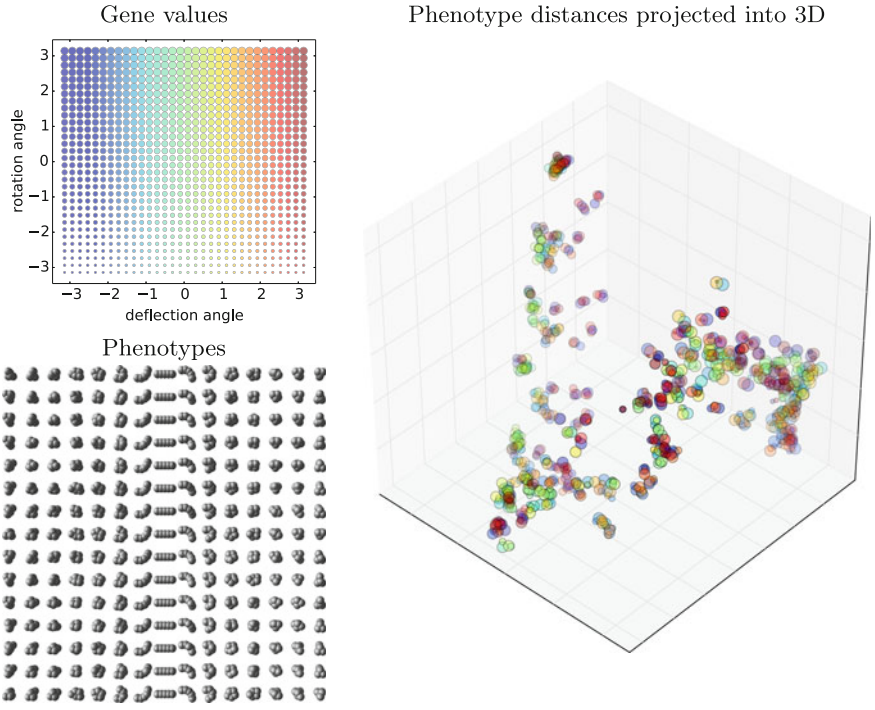


Fig. 4 The relationship between the genotype and the phenotype spaces for different values of deflection angle $\Delta\phi$ and rotation angle $\Delta\beta$. The number of phenotypes shown in the grid is reduced from 32×32 to 15×15 for legibility. The 3D projection of the phenotype distance matrix preserves 62 % of total variance

the same number of corresponding phenotypes were generated (we used 32 evenly spaced values for each of the two varied parameters).

For the visualization of the phenotype space, multidimensional scaling was employed [3]. The distance matrix that resulted from estimating similarity of all pairs of phenotypes was projected into three dimensions where the Euclidean distances best approximate the original distance matrix. These 3D coordinates were plotted using different colors and sizes that correspond to variable parameter values (genes). A small random jitter was added to 3D coordinates to avoid identical morphologies (with zero dissimilarity) to be plotted one over the other, and to expose their density.

Phenotype grid in Fig. 4 reveals symmetry of the deflection and rotation angles. It can also be seen in the 3D distances plot—morphologies corresponding to the same absolute values of the angles are grouped together. For instance, on the right side of the distance plot, groups comprising of two small and two big circles are visible. Sizes of the circles in those groups correspond to extreme rotation angle values, and colors correspond to opposite deflection angle values.

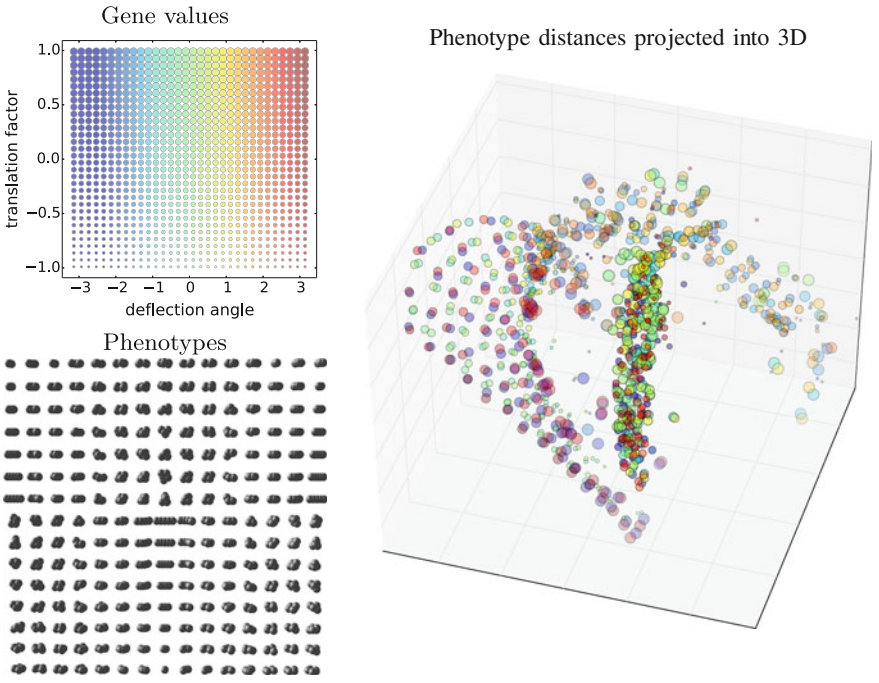


Fig. 5 The relationship between the genotype and the phenotype spaces for different values of translation factor TF and deflection angle $\Delta\phi$. The number of phenotypes shown in the grid is reduced from 32×32 to 15×15 for legibility. The 3D projection of the phenotype distance matrix preserves 47 % of total variance

Extreme values of the angles yield the deflection of about $+180$ and $-180°$, both of which locate the center of the new chamber in a similar position. This indicates that for locality to be preserved and for successful search in the genotype space, the operation of modifying both angles should respect their cyclic nature, and for the two angles, the genotype grid should be considered a torus.

Phenotype grid in Fig. 5 demonstrates that the positive translation factor and the extreme deflection angle values produce similar morphologies to the negative translation factor and the deflection angle values close to 0. In the 3D distances plot, this is represented by lines comprised of red and blue circles (extreme values of the deflection angle) and of light blue and green circles (deflection angle values near 0). Morphologies corresponding to the extreme deflection angle and the negative translation factor, or the deflection angle value near 0 and the positive translation factor, are more similar to each other—in the 3D distance plot they form a dense, elongated cluster. However, phenotypes with small positive and negative values of translation factor that are very close in the genotype space are mapped into distant phenotypes (different morphologies)—this discontinuity is clearly visible in the phenotype grid.

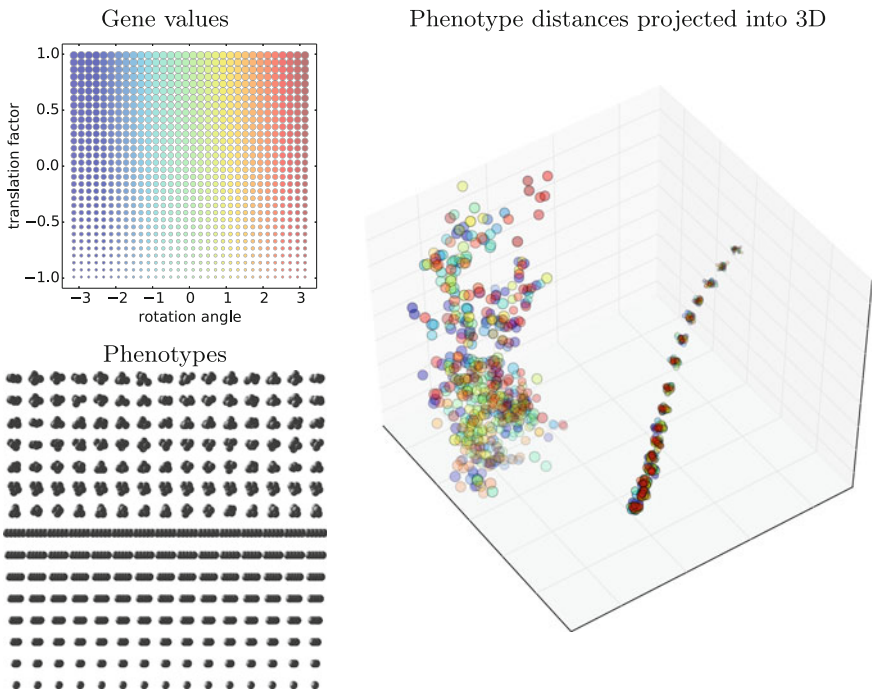


Fig. 6 The relationship between the genotype and the phenotype spaces for different values of translation factor TF and rotation angle $\Delta\beta$. The number of phenotypes shown in the grid is reduced from 32×32 to 15×15 for legibility. The 3D projection of the phenotype distance matrix preserves 53 % of total variance

Figure 6 reveals two groups, one of which is dense and the other is sparse. Closer analysis of both groups indicates that for $TF \leq 0$, the value of the rotation angle has no influence on morphology. Such morphologies form the linear dense group, where their geometries are different only because of different values of translation factors. Morphologies in the sparse group ($TF > 0$) depend on both the translation factor and the rotation angle. The presence of such distinctive groups and the lack of smooth transition between them indicates that there is a discontinuity in the interpretation of gene values, and this would be disadvantageous from the search and optimization point of view. An interesting discovery is the fact that rotation angle has any influence at all for $TF > 0$. Since deflection angle is zero in this experiment, based on the theoretical model, rotation should not have any influence on morphology. However, the implementation uses a finite number of point samples on the chamber spheres to find the communication path with minimal length, and the analysis presented here may have discovered an artifact caused by this sampling.

4 Conclusions

Visual comparison of the genotype and the phenotype spaces performed for three pairs of parameters revealed characteristics and potential weaknesses of the foraminifera model of morphogenesis. Genotypes with extreme values of translation and rotation angles correspond to similar morphologies. Genotypes with equally distributed values of rotation angle and translation factor are mapped into two distinct groups of morphologies with no smooth transition possible. Although further investigation of the model is needed, preliminary results reported here suggest low locality of the mapping. Note however that the mapping was not devised with optimization in mind, it was rather expected to model biological reality in an extremely simple way, using just a few key parameters. Low locality may also be a property of biological genotype-to-phenotype mappings [1] and as such, it may be considered a feature that should be included in the model, not a disadvantage.

There is a great potential for application of this methodology to real organisms, although there are some challenges to tackle. The fundamental problem is that high-level genes are in reality represented by complex genetic and epigenetic processes responsible for morphogenesis [18]. A more realistic approach could take into account real molecular genetic data based on DNA, RNA, or protein sequences [14, 20]. Real morphologies of foraminiferal shells are also based on chamber arrangement patterns—however, chambers are not defined by their theoretical centers. The most promising method to test would be to follow apertures and foramina that form graph-like communication lines reconstructed based on high-resolution X-ray computed tomography.

A more detailed analysis of the model of foraminifera growth is needed, including a numerical evaluation of the genotype—phenotype mapping. The results of such evaluation increase the understanding of the relationships between genes and phenes, and facilitate the development of an improved model of foraminifera morphogenesis.

Acknowledgments The research presented in the paper received partial support from Polish National Science Center (DEC-2013/09/B/ST10/01734).

References

1. Aguirre, J., Buldú, J.M., Stich, M., Manrubia, S.C.: Topological structure of the space of phenotypes: the case of RNA neutral networks. PloS One **6**(10), e26324 (2011)
2. Benoît, J., Guennebaud, G.: Eigen library. http://eigen.tuxfamily.org
3. Cox, T.F., Cox, M.A.A.: Multidimensional Scaling, 2nd edn. Chapman and Hall/CRC, Boca Raton (2000)
4. Itseez: OpenCV library. http://opencv.org/
5. Komosinski, M., Koczyk, G., Kubiak, M.: On estimating similarity of artificial and real organisms. Theory Biosci. **120**(3–4), 271–286 (2001)
6. Komosinski, M., Kubiak, M.: Taxonomy in alife. Measures of similarity for complex artificial organisms. In: Kelemen, J., Sosík, P. (eds.) Advances in Artificial Life, LNCS, vol. 2159, pp. 685–694. Springer, Berlin Heidelberg (2001)
7. Komosinski, M., Kubiak, M.: Quantitative measure of structural and geometric similarity of 3D morphologies. Complexity **16**(6), 40–52 (2011)
8. Komosinski, M., Mensfelt, A., Topa, P., Tyszka, J., Ulatowski, S.: Foraminifera: genetics, morphology, simulation, evolution. http://www.framsticks.com/foraminifera
9. Komosinski, M., Ulatowski, S.: Genetic mappings in artificial genomes. Theory Biosci. **123**(2), 125–137 (2004)
10. Labaj, P., Topa, P., Tyszka, J., Alda, W.: 2D and 3D numerical models of the growth of foraminiferal shells. In: Sloot, P.M.A., Abramson, D., Bogdanov, A.V., Dongarra, J.J., Zomaya, A.Y., Gorbachev, Y.E. (eds.) Computational Science–ICCS 2003. LNCS, vol. 2657, pp. 669–678. Springer, Berlin Heidelberg (2003)
11. MathWorks: Matlab. http://www.mathworks.com/products/matlab/
12. Meyer, C.: Matrix Analysis and Applied Linear Algebra. SIAM Society for Industrial and Applied Mathematics, Philadelphia (2001)
13. Numerical Recipes Software: SVD implementation. http://www.nr.com/webnotes/nr3web2.pdf
14. Pawlowski, J., Holzmann, M., Tyszka, J.: New supraordinal classification of foraminifera: molecules meet morphology. Mar Micropaleontol **100**, 1–10 (2013)
15. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes: The Art of Scientific Computing, 3rd edn. Cambridge University Press, New York (2007)
16. Rothlauf, F.: Representations for Genetic and Evolutionary Algorithms. Springer Verlag, Berlin (2006)
17. Topa, P., Komosinski, M., Bassara, M., Tyszka, J.: eVolutus: a configurable platform designed for ecological and evolutionary experiments tested on Foraminifera. In: Man-Machine Interactions 4. AISC, Springer (in press)
18. Topa, P., Tyszka, J., Bowser, S.S., Travis, J.L.: DPD model of foraminiferal chamber formation: simulation of actin meshwork–plasma membrane interactions. In: Wyrzykowski, R., Dongarra, J., Karczewski, K., Waśniewski, J. (eds.) Parallel Processing and Applied Mathematics. LNCS, vol. 7204, pp. 588–597. Springer, Berlin Heidelberg (2012)
19. Tyszka, J., Topa, P.: A new approach to modeling of foraminiferal shells. Paleobiology **31**(3), 526–541 (2005)
20. Tyszka, J. et al.: Morphogenetic bridge between molecules and morphology. In: FORAMS 2014, p. 6. The Grzybowski Foundation, Concepción, Chile (2014)

The Resection Mechanism Promotes Cell Survival After Exposure to IR

Monika Kurpas, Katarzyna Jonak and Krzysztof Puszynski

Abstract Ataxia telangiectasia mutated (ATM) protein kinase detects double-strand breaks (DSBs) caused by such environmental factors like ionizing radiation (IR), while ataxia telangiectasia mutated and Rad-3 related (ATR) is activated by the presence of single-stranded DNA areas (ssDNA). Moreover, biological reports show that ATR can be also activated in DSBs repair pathway. Based on experimental reports, we confirmed that the factor responsible for ATR activation may be ssDNA formedS after resection of DSBs by repair complexes. In this study, we propose a novel stochastic mathematical model of ATR-ATM-p53 pathways. The model demonstrates the process of resection and helps to explain the impact of the investigated modules on DNA damages repair. Our results show that the resection of DNA ends accelerates DNA damage repair. Disorders in the mechanisms of DNA repair and resection cause decrease in viability of cells population.

Keywords ATR · ATM · DNA repair · Mathematical model · Resection · Stochastic simulations · HR · SSA

1 Introduction

The proper functioning of each organism depends on the accurate transfer of undamaged genetic information from one cell to its daughters. This process may be interrupted by many factors, for example, by DNA lesions causing failures of DNA replication, transcription and other processes. The formation of the damages may appear due to the action of exernal and internal agents, like ultraviolet radiation (UV), reactive oxygen species, chemicals or replication errors. Even if the cells

M. Kurpas (✉) · K. Jonak · K. Puszynski
Institute of Automatic Control, Silesian University of Technology, Gliwice, Poland
e-mail: monika.kurpas@polsl.pl

K. Jonak
e-mail: katrzyna.jonak@polsl.pl

K. Puszynski
e-mail: krzysztof.puszynski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_19

are exposed on so many damaging factors, our organisms are still able to function correctly. The reason is the existence of number of mechanisms that evolved to the most efficient DNA repair [2].

DNA repair pathways involve the activity of damage sensor kinases, such as ataxia telangiectasia mutated (ATM), which detects double-strand breaks (DSBs) occuring in the cell in response to various agents, like ionizing radiation (IR), drugs or endogenous abnormalities, such as replication errors. The kinase after autophosphorylation amplifies the signal from repair proteins, among others Mre11-Rad50-Nbs1 (MRN) complex. Additionally, ATM phosphorylates and activates other signal transducers important in the damage detection pathways: checkpoint kinase 1 (Chk1) and 2 (Chk2). These three protein kinases phosphorylate and activate p53 transcription factor called "the guardian of the genome". P53 is responsible for cell fate determination—it is involved in regulation of number of pathways leading to i.e. DNA damage repair or apoptosis [2].

Ataxia telangiectasia mutated and Rad3-related (ATR) is activated in the presence of single-stranded DNA (ssDNA). Such forms might occur at stalled replication forks or during DNA repair process. Single stranded DNA is coated by replication protein A (RPA) complex, which recruits among others Rad9-Rad1-Hus1 (9-1-1) and ATR-ATRIP (ATR-interacting protein) complexes. ATR after its autophosphorylation and activation by 9-1-1 complex becomes able to phosphorylate and activate Chk1, Chk2 and p53. ATM, ATR, Chk1 and Chk2 not only positively regulate p53, but also cause increased deactivation and/or degradation of major p53 inhibitor mouse double minute 2 homolog (Mdm2), the E3 ubiquitin-protein ligase. Mdm2 is transcriptionally activated by p53 and in turn ubiquitinates this transcription factor leading to its proteosomal degradation.

DSBs may be repaired by non-homologous end joining (NHEJ), homologous recombination (HR) or single-strand annealing (SSA) mechanisms. HR requires the presence of second sequence which could be used as a template to faithful reconstruction of damaged DNA chain. It occurs mainly in these stages of cell cycle when genetic material exists in two copies (S, and G2 phase) [2]. SSA mechanism repairs DSBs between two repeated sequences of nucleotides. It does not require separate indentical or similar molecule of DNA, but only single DNA duplex, which is processed through resection and then ligated [2].

The HR and SSA mechanisms use the resection during DSBs ends processing. The resulting fragments of ssDNA are coated by RPA complex and activate ATR detection module [5]. Here, we do not focus on HR or SSA precise mechanism, but we demonstrate dependencies between DSBs occurrence and ATR module activation.

Better understanding of the processes taking place in the cell, without costly and long lasting biological experiments, becomes possible through the use of the systems biology tools. In this study, we describe a simple mathematical model of detection of DSBs and ssDNA, here also called single-strand breaks (SSBs). We confirmed that DSBs resection increases effectiveness of repair and lowers apoptotic fractions of irradiated cells.

2 ATR-ATM-p53 Mathematical Model

2.1 Existing Models

Interactions between the components of p53 and ATM pathways have been already well studied and analyzed using mathematical modelling approach [7, 12, 14, 17, 18]. However, the ATR part is usually neglected. Only two existing models of ATM-p53 considers their interactions with ATR. In Zhang et al. ATR and ATM are treated together as one damage detector element [18]. This approach causes a lack of possibilty to show complex interactions of these two kinases in response to various stimuli. The model shows only deterministic effects of IR treatment. In Batchelor et al. two models of ATR (activated by UV) and ATM (activated by γ irradiation) were described [1]. There was also attempt to show the crosstalk between these two modules according to [5], but the results shown marginal difference between response of the model taking into account activation of ATM after UV and ATR after γ irradiation.

To our knowledge, there are none detailed models of the ATR-ATM-p53 pathways containing DSBs resection. There are some models of homologous recombination but they focus on the precise mechanism of resection, not on the overall effect. In Rodriguez et al. [15], authors focused on reconstruction of FA/BRCA network which contains ATM and ATR detector proteins, but they are not the main subject of that study.

2.2 Model Assumptions

The ATR-ATM-p53 model was built using basic laws known from the biochemistry—the law of mass action and the Michaelis-Menten kinetics. According to the Haseltine-Rawlings postulate [4], this hybrid model binds deterministic (Runge-Kutta 4th order method) and stochastic (direct Gillespie method [3]) approaches. We use ordinary differential equations (ODE) for deterministic description of fast reactions, like phosphorylation events, and stochastic propensities to describe slow reactions, such as genes activation and damage formation.

Our model is not developed for any specific cell line. We rather consider general model of the hypothetical cell with generally accepted dependencies. The model may be later fitted for the given cell line if at least part of its parameters will be obtained.

Our model distinguishes two compartments—nucleus and cytoplasm. It is divided on three major modules: ATM, ATR and p53 part. Details about the p53 signaling pathway are available in [13].

A simplified model of the DNA damage detection is acivated by irradiation resulting in DNA breaks. In case of ssDNA caused by UV, ATR part of the model is activated. DSBs (induced by IR) activates both ATM part and indirectly ATR part as a result of resection (Fig. 1). The output of the model is p53 level.

According to Kracikova et al. results [8], we assumed that cell fate depends on p53 level. If continously elevated above the threshold, p53 promotes apoptosis by cooperating with Bax protein in release of cytochrome c and other apoptotic factors from mitochondria [11]. Apoptotic condition is recognized as the level of active p53 increased above the $2.2 \cdot 10^5$ threshold for more than 6 h. Number of DNA breaks and protein molecules decrease after apoptosis, but we do not consider this as a repair—we assumed that after cell's death all of its components are rapidly degraded and are not further investigated.

In our model, we assumed also persistent cellular stress level triggered by spontaneous ssDNA formation, what results in continuous basal activation of detector proteins and p53.

The state of the gene for each molecule in the stochastic model takes value 0, 1 or 2 depending on the number of active gene copies. In the deterministic approximation, its value is in the range [0, 2], what represents the average state

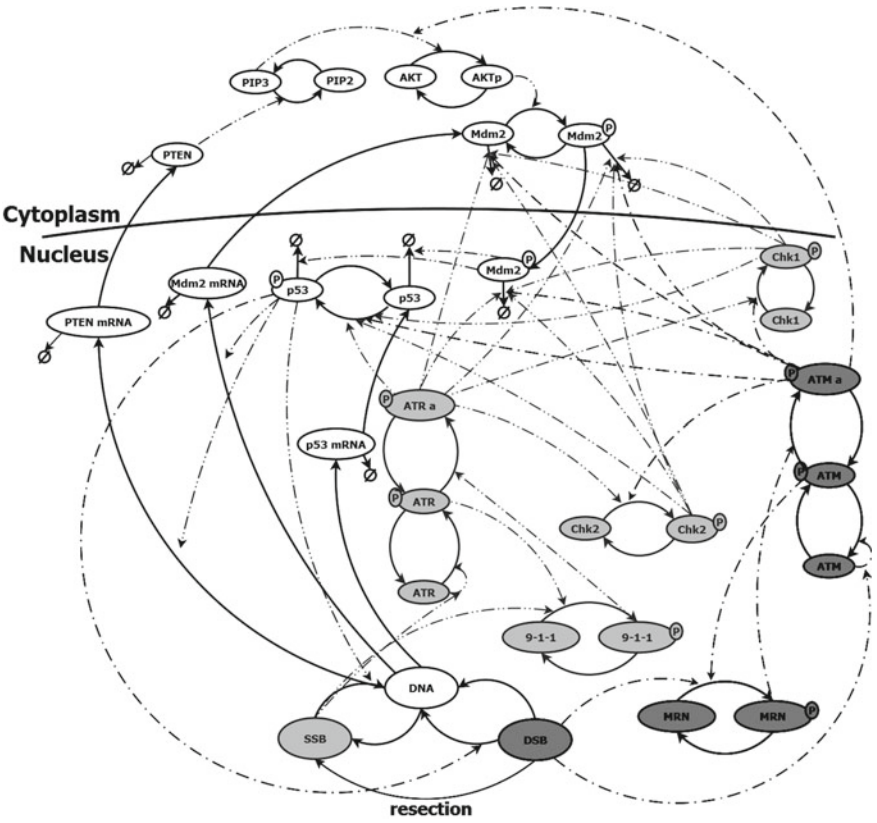


Fig. 1 ATR-ATM-p53 model scheme with three modules: p53 (*black* and *white*), ATR marked with *light gray* and ATM marked with *dark gray*. *Solid lines* represent change of protein form; *dashed lines* describe the interactions occurring in the path; *p* stands for phosphorylation; *a* represents full activation of protein or complex

Table 1 The selected parameters of the ATR and ATM parts

Name	Description	Value
ra_8	Spontaneous *SSB* formation	1.4×10^{-3}
rd_{DAM}	*SSB* damage caused by $1\,\mathrm{J/m^2}$ UVC	52
rd_{REP}	*SSB* repair rate	15×10^{-11}
rn_{SAT}	Saturation coefficient in *SSBs* repair	50
q_0	*Mdm2* and *PTEN* genes spontaneous activation	1×10^{-4}
q_1	*Mdm2* and *PTEN* genes activation by $p53_p$	5×10^{-13}
q_2	*Mdm2* and *PTEN* genes spontaneous deactivation	3×10^{-3}
md_{DAM}	*DSB* damage caused by *IR*	0.0585
md_{REP}	*DSBs* repair rate	0.001
mm_1	MM for *DSB* repair	10
res_p	Percent of DSB repaired by ends resection	20
res_t	*DSB* to *SSB* transformation	1

of cell population. For most of the proteins, transcription, translation and degradation were omitted in order to simplify the description. We assumed that total number of these proteins is constant and they can only change their state between active and inactive form.

Resection process in our model is described gradually: we have some amount of DSBs which can be repaired by ends resection: DSB_{res} and the other DSBs. The percentage of resection-repaired DSB is given in Zhou et al. and it is equal to 20 % [19]. In the first step, we assumed resection understood as requiring MRN complex transformation of DSB to structure containing ssDNA, which in the next step is recognized by ATR module and repaired by factors engaged by the ATR-p53 pathway.

2.3 The Model Equations, Variables and Parameters

We obtained our model's parameters from literature. Part of them was given directly, but part of them was calculated using program described in [6] which measure fold change between intensity of tracks in western blot experiments. Parameters which cannot be obtained were estimated by fitting the model to the remain data, with keeping the protein levels corresponding to reality. For all the model parameters n stands for number of molecules or active genes, and MM stands for Michaelis-Menten constant (Table 1).

The current model is based on ATR [9] and p53 [13] models. We added relations between the ATR-p53 and the ATM pathway, for example ATM dependent Chk1, Chk2, p53, Mdm2 and Akt activation.

Number of SSBs: First parameter provides basic activation of the *ATR-p53* pathway, whereas next term describes UVC dose-dependent and damaging coefficient-dependent (rd_{DAM}) *SSBs* formation. The last positive term represents

DSBs resected to *SSBs* with participation of *MRN* complexes. The negative part describes *SSBs* repair, which depends on number of $p53$ teramers, repair rate (rd_{REP}) and number of repair complexes (rn_{SAT}).

$$\frac{d}{dt}SSB(t) = ra_8 + rd_{DAM}UV(t) + res_p\frac{MRN_p(t)}{MRN_p(t)+1}DSB_{res}(t) - rd_{REP}P53_p^2(t)\frac{SSB(t)}{SSB(t)+rn_{SAT}} \tag{1}$$

Number of DSBs which are not repaired by resection: First term describes IR dose-dependent and damaging coefficient-dependent (md_{DAM}) *DSBs* formation. These *DSBs* need not to be transformed to *SSBs* ($[1 - res_p]$ coefficient) before repair. The last part describes *DSBs* repair, which depends on number of $p53$ tetramers, repair rate (md_{REP}) and Michaelis-Menten coefficient for *DSBs* repair (mm_1).

$$\frac{d}{dt}DSB(t) = (1 - res_p)md_{DAM}IR(t) - md_{REP}\frac{DSB(t)}{DSB(t)+mm_1}\frac{q_0 + q_1P53_p(t)^2}{q_2 + q_0 + q_1P52p(t)^2} \tag{2}$$

Number of DSBs repaired by resection: First term describes IR dose-dependent and damaging coefficient-dependent (md_{DAM}) *DSBs* formation. This part of *DSBs* need to be transformed to *SSBs* (res_p coefficient) before repair. Second term represents *DSBs* to *SSBs* resection regulated by *MRN* complexes with coefficient res_t.

$$\frac{d}{dt}DSB_{res}(t) = res_p \cdot md_{DAM}IR(t) - res_t\frac{MRN_p(t)}{MRN_{p(t)+1}}DSB_{res}(t) \tag{3}$$

Total number of DSBs: Total number of *DSBs* caused by ionizing irradiation.

$$DSB_{tot}(t) = DSB(t) + DSB_{res}(t) \tag{4}$$

3 Results

We tested cellular response to selected doses of UVC and IR according to the literature (for more details, please see [9, 13]). We fitted ATM and ATR part together to achieve the proper response to both stress agents. We performed 200 stochastic experiments, where doses of irradiation were given at time equal 24 h after beginning of the experiment and observed over the next 196 h.

3.1 Response to UV

We examined the ATR-ATM-p53 model response to 18 J/m^2 UV (Fig. 2a). The result was consistent with our previous study where we had shown that after 18 J/m^2 UVC 28000 lesions are formed and are repaired during 24 h. We chose this dose because we considered it as apoptotic threshold—above this dose the majority of cells died [9].

3.2 Response to IR Shows Importance of Resection Mechanism

We simulated cells treated with dose of 10 Gy. According to [10, 16] 1 Gy of IR causes 35 DSB. We fitted our model to above data. The ATM part of our model was based on the above assumption. We exposured simulated cells to IR dose of 10 Gy what results in about 350 DSBs formation. Here we show the outcome of the DSBs resection—level of DSBs is lower than we expect (Fig. 3a), because part of lesions is continously transformed to ssDNA what results in ATR module activation after IR (Fig. 3b). The peak of SSBs level is very short because they are very quickly recognized and processed by repair complexes. Stimulation of both modules results in increased level of active p53 (Fig. 3c) and repair of the damage.

In comparison, in the model of DSBs repair without resection all of 350 established lesions are formed (Fig. 3j). The time of DNA repair takes significantly much longer than in case with enabled DSBs resection. About 10 % greater is also apoptotic fraction among cells lacking resection mechanism (Fig. 2b).

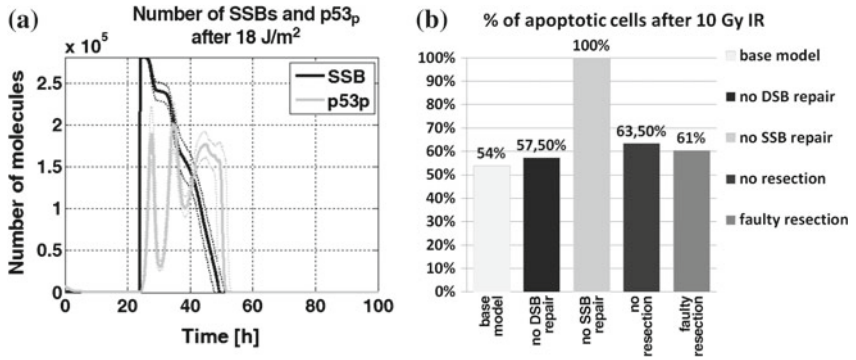


Fig. 2 **a** Response of ATR-ATM-p53 system to UV dose of 18 J/m^2. Result for 200 stochastic simulations. *Solid line*—median; *dashed line*—upper and lower quartile of the result; **b** Comparison of apoptotic fractions after 10 Gy of IR in various mutations of DSBs repair. Fractions measured for 200 cells

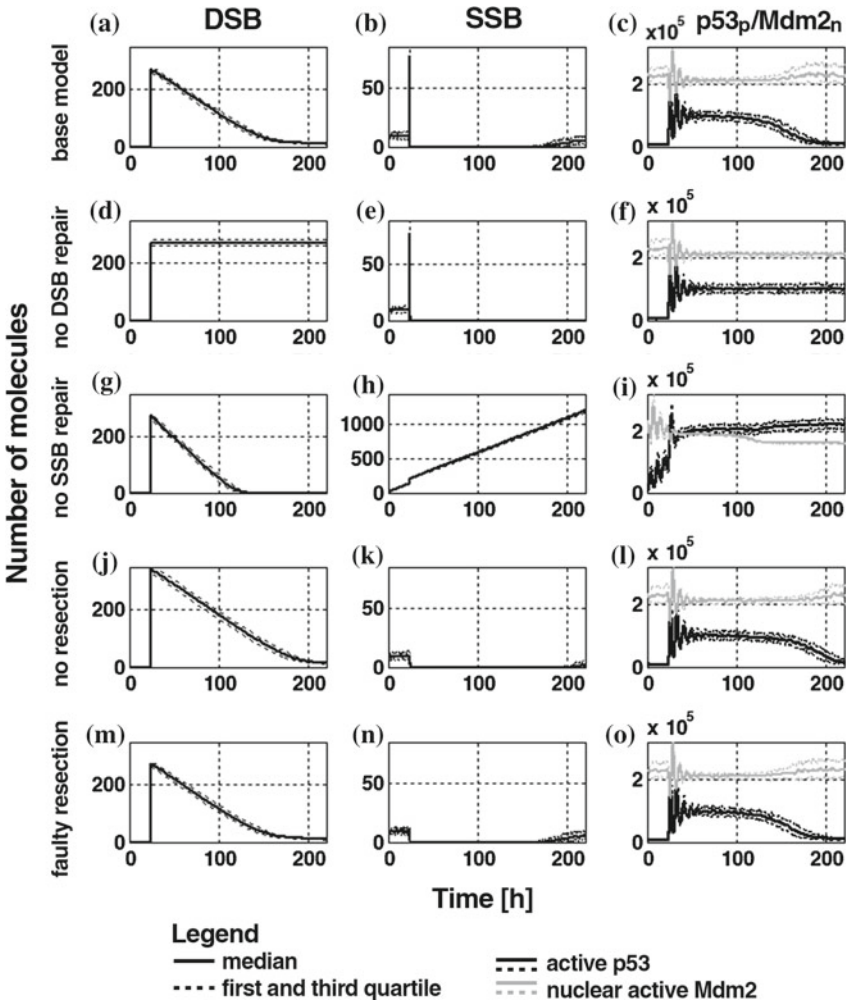


Fig. 3 Response of ATR-ATM-p53 to IR dose of 10 Gy in various mutations of DSBs repair. Result for 200 stochastic simulations. *Solid line*—median; *dashed line*—upper and lower quartile of the result. **a** number of DSB in the properly functioning pathway, **b** number of SSB in the properly functioning pathway, **c** level of active p53 and nuclear Mdm2 in the properly functioning pathway, **d** number of DSB in the case of faulty DSB repair, **e** number of SSB in the case of faulty DSB repair, **f** level of active p53 and nuclear Mdm2 in the case of faulty DSB repair, **g** number of DSB in the case of disorders in SSB repair, **h** number of SSB in the case of disorders in SSB repair, **i** level of active p53 and nuclear Mdm2 in the case of disorders in SSB repair, **j** number of DSB in the case of lack of resection mechanism, **k** number of SSB in the case of lack of resection mechanism, **l** level of active p53 and nuclear Mdm2 in the case of lack of resection mechanism, **m** number of DSB in the case of unproperly functioning resection mechanism, **n** number of SSB in the case of unproperly functioning resection mechanism, **o** level of active p53 and nuclear Mdm2 in the case of unproperly functioning resection mechanism

3.3 Despite Faulty DSBs Repair, Part of Lesions is Still Recognized and Repaired

We decided to check whether the IR-caused lesions will become repaired if there will be some abnormalities in DSBs repair pathways (for example disorders in NHEJ mechanism). Our results suggest that part of DSBs will be repaired engaging the ATR pathway to signal transduction (Fig. 3e). The level of p53 remains elevated (Fig. 3f) because the module is continuously activated by unrepaired genetic material (Fig. 3d).

3.4 Disorders in SSBs Repair Cause Increased Apoptosis Among the Cells

ATR module is essential in viability of organisms because it protects genomic stability during replication process which might be interrupted by the damage occurrence. Inability to repair the damage will result in directing cells to apoptosis, because large number of unrepaired SSB lesions (Fig. 3h) send strong signal which activates p53 and maintain it on really high level (Fig. 3i). The continuously increasing level of SSB is a result of assumed basal SSB formation. Obtained result clearly show that SSB repair mechanism is crucial for cell survival. Our results indicate that in the population of cells with faulty SSBs repair all of the cells die.

3.5 Faulty DSBs Resection Causes the Formation of Unrepaired DNA

We tried to examine what may occur if the resection mechanism does not work in the right way. For example, what happens if the resection is stuck on some stage so there is still DSB recognized by the system and it cannot be repaired? Our results suggests that both SSBs and DSBs are repaired in normal way, but the number of DSBs which may be repaired through resection (20 %) is not repaired by any pathway (Fig. 3m–n).

4 Conclusions

In this paper, we described our stochastic mathematical model of the ATR-ATM-p53 pathway. The main goal of this project was to check in the simplest possible way whether ATR activation occurs after IR and what can be the reason of this activation. We confirmed that the process activating ATR module might be the formation of ssDNA areas after resection of DSBs.

The results obtained from stochastic simulations performed on 200 cells confirmed that resection of DNA ends accelerates the DNA damage repair through amplifying the signal and making DNA chain more accessible for repair factors. Disorders in mechanisms of repair and resection cause decrease in viability of cells population.

We plan to expand the model developed in this work and build more accurate mathematical description of resection mechanisms. We consider perform biological experiments that will give us the rates of the parameters for the specific cell line and will be used to verifying our results.

Acknowledgments This project was funded by the Polish National Center for Science granted by decision number DEC-2012/05/D/ST7/02072.

References

1. Batchelor, E., Loewer, A., Mock, C., Lahav, G.: Stimulus-dependent dynamics of p53 in single cells. Mol. Syst. Biol. **7**(1), 488 (2011)
2. Ciccia, A., Elledge, S.J.: The DNA damage response: making it safe to play with knives. Mol. Cell **40**(2), 179–204 (2010)
3. Gillespie, D.T.: Exact stochastic simulation of coupled chemical reactions. J. Phys. Chem. **81**(25), 2340–2361 (1977)
4. Haseltine, E.L., Rawlings, J.B.: Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics. J. Chem. Phys. **117**(15), 6959–6969 (2002)
5. Jazayeri, A., Falck, J., Lukas, C., Bartek, J., Smith, G.: ATM- and cell cycle-dependent regulation of ATR in response to DNA double-strand breaks. Nat. Cell Biol. **8**(1), 37–45 (2006)
6. Jonak, K., Jedrasiak, K., Polanski, A., Puszynski, K.: Aplication of image processing in proteomics: automatic analysis of 2-D gel electrophoresis images from western blot assay. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) Computer Vision and Graphics. LNCS, vol. 7594, pp. 433–440. Springer, Berlin (2012)
7. Kim, D.H., Rho, K., Kim, S.: A theoretical model for p53 dynamics. Identifying optimal therapeutic strategy for its activation and stabilization. Cell Cycle **8**(22), 3707–3716 (2006)
8. Kracikova, M., Akiri, G., George, A., Sachidanandam, R., Aaronson, S.: A threshold mechanism mediates p53 cell fate decision between growth arrest and apoptosis. Cell Death Differ. **20**(4), 576–588 (2013)
9. Kurpas, M., Jonak, K., Puszynski, K.: Simulation analysis of the ATR module as a detector of UV-induced DNA damage. In: Piętka, E., Kawa, J., Wieclawek, W. (eds.) Information Technologies in Biomedicine, vol. 3, AISC, vol. 283, pp. 317–326. Springer, Switzerland (2014)
10. Lobrich, M., Rydberg, B., Cooper, P.: Repair of x-ray-induced DNA doublestrand breaks in specific Not I restriction fragments in human fibroblasts: joining of correct and incorrect ends. Proc. Natl. Acad. Sci. **92**(26), 12050–12054 (1995)
11. Mihara, M., Erster, S., Zaika, A., Petrenko, O., Chittenden, T.: p53 has a direct apoptogenic role at the mitochondria. Mol. Cell **11**(3), 577–590 (2003)
12. Mouri, K., Nacher, J., Akutsu, T.: A mathematical model for the detection mechanism of DNA double-strand breaks depending on autophosphorylation of ATM. PlosOne **4**(4), e5131 (2006)
13. Puszynski, K., Hat, B., Lipniacki, T.: Oscillations and bistability in the stochastic model of p53 regulation. J. Theor. Biol. **254**(2), 452–465 (2008)
14. Puszynski, K., Jonak, K., Kurpas, M., Janus, P., Szoltysek, K.: Analysis of ATM signaling pathway as an activator of p53 and NF-kB regulatory modules and the role of PPM1D. In: IWBBIO 2014, pp. 1471–1482. Granada, Spain (2014)

15. Rodriguez, A., Sosa, D., Torres, L., Molina, B., Frias, S., Mendoza, L.: A Boolean network model of the FA/BRCA pathway. Bioinformatics **28**(6), 858–866 (2012)
16. Rothkamm, K., Lobrich, M.: Evidence for a lack of DNA double-strand break repair in human cells exposed to very low x-ray doses. Proc. Natl. Acad. Sci. **100**(9), 5057–5062 (2003)
17. Sun, T., Yang, W., Liu, J., Shen, P.: Modeling the basal dynamics of P53 system. PlosOne **6**(11), e27882 (2011)
18. Zhang, H.P., Liu, F., Wang, W.: Two-phase dynamics of p53 in the DNA damage response. Proc. Natl. Acad. Sci. **108**(22), 8990–8995 (2011)
19. Zhou, Y., Caron, P., Legube, G., Paull, T.: Quantitation of DNA double-strand break resection intermediates in human cells. Nucleic Acids Res. **42**(3), e19 (2014)

Integrative Construction of Gene Signatures Based on Fusion of Expression and Ontology Information

Wojciech Łabaj and Andrzej Polanski

Abstract Gene signatures are lists of genes used for summarizing high-throughput gene expression profiling experiments. Various routines for obtaining and analyzing gene signatures in molecular biology studies exist, including statistical testing with false discovery corrections and annotations by gene ontology keywords. Despite the presence of well established routines there are still challenges in efficient application of gene signatures, which include gene signature instability, problems in defining optimal sizes and possible unreliability of inference results. Therefore there are continuous attempts towards improving algorithms for constructing meaningful gene signatures. In this paper we are introducing a methodology for constructing gene signatures, based on the fusion of information coming from statistical tests for differential gene expression analysis and resulting from statistical tests for GO terms enrichment analysis. On the basis of the DNA microarray datasets we are demonstrating that the proposed algorithm for fusion of expression and ontology information leads to improvement of the composition of gene signatures.

Keywords Gene expression · Gene signature · Gene ontology · Functional analysis

1 Introduction

DNA microarrays originally developed to study differential gene expression using diverse populations of RNA undergo continuous refinements and developments, such as disease diagnosis, gene discovery, drug discovery or toxicological research, with regard to gene expression profiling, comparative genomic hybridization, SNP detection and many others. DNA microarray technology also becomes an important element of integrative 'omics' studies, complementary to other methods of high throughput molecular measurements, e.g., sequencing [21, 27].

W. Łabaj (✉) · A. Polanski
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: wojciech.labaj@polsl.pl

A. Polanski
e-mail: andrzej.polanski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_20

DNA microarrays measure expression of tens of thousands of genes or gene products simultaneously. They are often summarized by gene signatures [9], which are lists of genes exhibiting certain patterns of expression across experiments. Gene signatures are frequently treated as a starting point for further downstream analysis, where biological conclusions from experimental results are drawn. Therefore, analyzes of gene signatures are an important area of research in bioinformatics. Searching for information concerning genes in signatures can be based annotations stored in dedicated databases, Gene Ontology (GO) [3], KEGG Pathways [17], motifs from InterPro database [15], keywords describing entries from the UniProt database [22] and many others [11].

Gene Ontology is a special source of gene annotation information, due to its hierarchical structure and controlled vocabulary, used by biologists as a standardized nomenclature for many specialized, large biological databases. GO terms are divided into three domains: biological process (BP), molecular function (MF) and cellular component (CC). Genes and gene products have been annotated to the categories of GO using the best currently available knowledge [7]. A hierarchical structure of GO terms, described by the directed acyclic GO graph (DAG) allows for representing biological knowledge at different levels of detail (GO edition from 30/11/2014 has 15 levels in the BP, 15 levels in MF and 13 levels in CC DAG).

A very common approach is that the functional analysis based on GO terms is performed for the domain of interest and level selected arbitrarily. However, two terms located at the same level in the GO graph can provide very different level of detail. Therefore, in the present article the ratio of Information Content (IC) [5] is used to characterize the accuracy of a GO term instead of the graph level.

There is a wide range of tools and algorithms to determine which GO terms are significantly overrepresented considering the given list of gene signatures. The basic approach implemented in many tools for GO terms enrichment/depletion evaluation is based on computation of frequencies, separately for each of the GO terms. In this approach, named *classic* [2] no dependencies related to the topology of the GO graph are taken into account. On the contrary, more advanced algorithms take into account dependencies resulting from the topology of the GO graph. These types of algorithms address the inheritance problem, related to the fact that annotations assigned to more general, ancestral GO terms are inherited from more specific, descendant GO terms, which can mislead the biological interpretation of GO signatures. To overcome this problem a new approach to GO signatures analyses includes additional steps of decorrelation of the GO DAG graph (named *elim, weight, weight01, lea, parentchild* [2, 12]).

All these algorithms for enrichment/depletion analysis of GO terms take into account exclusively binary information on gene selection (genes in the signature, all annotated genes, all genes of the considered organism etc.), but drop the full information from expression analysis from the previous steps. In this paper we want to point out that the gene signature construction can be stated as the problem of integrating data derived from gene expression analysis and GO terms assigned to genes.

In this paper we are presenting an approach, where information on the statistical significance of the differential gene expression is fused with the information concerning enrichment/depletion of GO terms assigned to genes. Such an approach does not overlook the valuable information that has been obtained during the differential expression analysis, namely the p-value for each gene. It also allows us to filter out GO terms of low Information Content (IC), which is crucial for many applications, e.g. tumors classification. Unbiased comparison of methods for determining the over-representation of GO Terms and assessing the quality of the results is a challenging task. Therefore, we have applied multiple complementary approaches. To this end biological consistency, the number of reduced GO terms and the characteristics of the IC ratio in the remaining GO terms were assessed. In addition to providing a robustness of our comparison we have focused on testing the accuracy of the classification and stability of gene signatures. A side effect of reducing the list of enriched GO terms is limiting the size of the gene signature. We have thus also assessed the impact of the used methods on the size of the gene signatures and its repeatability.

2 Data

Unbiased comparison of our method with the state-of-the-art algorithms for GO Terms enrichment analysis has been performed on two real datasets. We have used multiple DNA microarray experiments related to astrocyte cancer and leukemia.

2.1 Dataset I

Astrocytic brain tumors are cancers of the primary central nervous system (CNS), which develop from astrocytes and are most common glial tumors. They can be divided into two groups according to the way of growth, diffused or localized. In our study we are focusing on the astrocytic brain tumors with the diffused growth, which give poorer prognosis and are assigned to a higher grade according to the World Health Organization (WHO) [20]. We are further confining the research to the two most common tumors from this group, namely anaplastic astrocytoma (AA) and glioblastoma multiforme (GBM). Additionally, there are two different forms of the GBM, primary GBM arising de novo and secondary GBM arising from lower grade diffuse astrocytoma [23]. Primary and secondary glioblastomas are histologically indistinguishable, except the facts that the frequency of extensive necrosis is higher for the primary GBM and the frequency of oligodendroglioma components is higher for the secondary GBM [14]. Similar histopathology of glioblastomas may be due to similarity of genetic alterations behind their growths.

On the basis of the biological and clinical characteristics of astrocytoma and primary and secondary glioblastomas, it seems an interesting issue to design an experiment, aimed to highlight both the differences and similarities between these

cancers. We have explored the Gene Expression Omnibus (GEO) database [4] for experiments where gene expression profiling corresponding to the above-mentioned tumors and their comparisons to normal tissues was performed. The datasets found in GEO originate from three studies [6, 13, 19], which relied on comparisons of the three types of brain tumors, AA, primary GBM (GBM.P) and secondary GBM (GBM.S) with the normal brain tissues (NBT).

2.2 Dataset II

Although the first data set is very compelling, it lacks the sufficient number of samples for statistically significant classification procedure. In order to be able to perform such classification another data set has been compiled. Here we have focused on a more frequently studied disease (leukemia), so that we were able to collects sufficient number of microarray samples to perform the classification.

Leukemia is a group of cancers that usually begins in the bone marrow and results in a high numbers of abnormal white blood cells, which are called blasts. It is part of a broader group of neoplasms which affect the blood, bone marrow, and lymphoid system, known as tumors of the hematopoietic and lymphoid tissues [26]. But the exact cause of developing leukemia is unknown, it is even believed that each type of leukemia has a different cause [16].

There are four main types of leukemia: acute lymphoblastic leukemia (ALL), acute myeloid leukemia (AML), chronic lymphocytic leukemia (CLL) and chronic myeloid leukemia (CML), as well as a number of less common types [25]. In this article we have focused on studies where ALL and AML were investigated.

ALL is the most common type of leukemia in young children. This disease also affects adults, especially those aged 65 and older, whereas AML occurs more commonly in adults than in children.

We searched the GEO database for the experiments related to these two types of leukemia. The gene expression datasets originating from three studies were selected [10, 18]. We have chosen 84 samples from these experiments—42 samples for each type of leukemia respectively.

3 Methods

3.1 DATA Normalization and Differential Expression Analysis

All microarray analyses were performed with use of state of the art academic processing software. A custom CDF file for RefSeq annotation from BrainArray [8, 24] was used as it provides the latest genome and transcriptome information. GC Robust Multi-array Average (GCRMA) was used as background adjustment on Affymetrix

microarray probe-level data. In this method the sequence information is summarized by base types at each probe position, in a more complex way than the simple GC content. Variance Stabilization and Normalization (VSN) has been used for normalizing microarray intensity. It ensures that the variance remains nearly constant over the whole intensity spectrum. VSN has been used for normalization within experiments and normalization of all microarrays together (also to remove batch effect). As a summarization step the Affymetrix Probe-Level Modeling (affyPLM) was used. The next step was to create a model of comparison between case and control, after which data was fitted to the model. For this purpose Linear Models for Microarray Data (limma) were used, which provide the ability to analyze comparisons between many RNA targets simultaneously. Holm correction has been applied in order to correct statistical test results for multiple comparison, with threshold equal to 0.05. As a result, gene signatures were obtained, which allow for enrichment analysis of GO terms.

3.2 Enrichments Analysis of GO Terms

R package *'topGO'* [1] is an effective tool for semi-automatic enrichment analysis of GO terms. The package includes a set of ready-to-use functions for carrying out GO enrichments analysis. Not all combinations between algorithms and statistical tests currently supported by topGO are allowed [1].

The *elim* and *weight* algorithms were introduced by Alexa et al. [2]. Both methods investigate the nodes in the GO DAG from bottom to top and introduce weights for genes. In *elim* method the weighting process boils down to assigning 0 or 1 to genes. This means that the *elim* method eliminates the genes from the list of significant genes in ascendant GO terms, once they have been found to be associated with a statistically significant GO term. In *weight* method the process of assigning weights to genes is described as a function which always attains a value in the interval [0, 1]. The default algorithm used by the *topGO* package is *weight01*, which is a mixture of the *elim* and the *weight* algorithms. The *parentChild* algorithm was introduced by Grossmann et al. [12]. It measures overrepresentation of the GO term in the context of gene annotations to the parents of the term, not to the root.

During our analysis Fisher's Exact Test was used exclusively for simplicity of the comparison. It belongs to the class of exact tests, because the significance of the deviation from a null hypothesis can be calculated exactly, rather than relying on an approximation, as with many statistical tests.

3.3 Enrichment Analysis by Fusion

Our new heuristic approach for measuring overrepresentation of GO terms can be described as a method for filtering results obtained by the *classic* annotation

routine. The method requires as input data results of the *classic* algorithm and results of differentiation tests (in the form of a list of genes with corresponding p-values). Output data for each GO term are two indexes, namely enrichment index (EI) and differentiation index (DI). Fisher's combined probability test was used as a pattern of conduct to transform p-values from the enrichment test for the GO term and to fuse p-values from differentiation tests for genes annotated to this GO term. For standardization purposes indexes for each GO term are scaled to 0–1 range, which provides the method's applicability for any dataset. The methods of indexes' calculation are shown in formulae (1) and (2).

$$EI_{GOTerms_i} = -2ln\left(ET_{GOTerm_i}\right) \tag{1}$$

where,
ET—enrichment test (*classic* method).

$$DI_{GOTerm_i} = \frac{-2\sum_{j=1}^{n} ln\left(DT_{gene_j}\right)}{n} \tag{2}$$

where,
DT—differentiation test and
n—number of annotation genes to i-th GO term.

The filtering condition is the distance on two-dimensional plane of indexes between each GO term and the point (1,1). Exemplary result of our new approach are shown in the diagrams below (Fig. 1).

Our goal was to fuse information coming from the enrichment analysis of GO terms and differentiation analysis. Therefore, the cut-off was set as on arc of a circle of

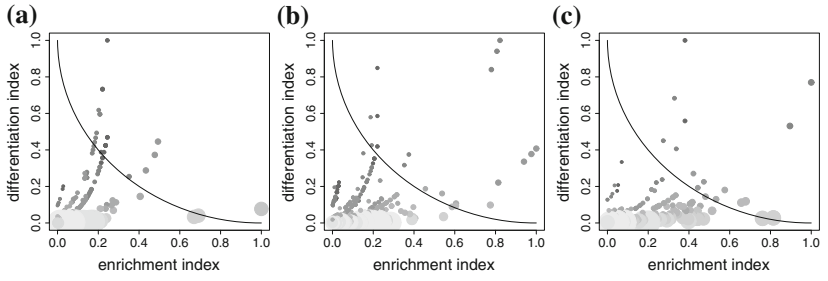


Fig. 1 Diagram presents an example of results for the new method. The x-axis is the enrichment index (EI) and y-axis is the differentiation index (DI). Color and size of the dots describe the information content (IC) ratio (*dark grey*—highest, *light grey*—lowest) and the number of significant genes associated with the GO term (the more genes the larger dot). Filtering condition is distance between each GO Term and the point (1,1) which correspond to perfect, maximal enrichment. Cut-off value has been chosen empirically and set to 1 (GO terms with higher distance are filtered out). The black line is the boundary cut-off. **a** AA, **b** GBM.P, **c** GBM.S

radius equal to 1. This contributes to the rejection of GO terms with low information content. However, in the analyzed cases, there were a few GO terms, which, despite low information content (IC) had a high rate of EI. They can be an integral part of the particular case and the method allows to retain them.

3.4 Criteria for Comparison of Algorithms

The Dataset I was used to investigate the characteristic of results of each method for enrichment analysis of GO Terms. First step was to check the cohesion of results with biological knowledge. More precisely, we assume that both tumors GBM.P and GBM.S should have a common part of the significant genes as well as have a common part of enriched GO Terms, which is expected to be larger than the common parts of GBM and AA.

Another step in the comparison is to check the quality of the results by examining the information content (IC) of enriched GO Terms. Dataset I will also provide us information on the number of reduced GO Terms, which entails a reduction in the size of the gene signature.

The Dataset II was used then for classification and stability analysis of gene signatures. As a validation method k-fold cross validation algorithm was selected. Therefore Dataset II was divided into 6 subsets, each of which contained 7 samples of both types of leukemia. We used this kind of division to maintain equal participation of both leukemia samples from different experiments. Training and validation sets were then subjected to data normalization, differential expression analysis, enrichment analysis of GO Terms and re-annotation of the significant GO Terms back to the genes to obtain gene signatures (features). K-nearest neighbors' algorithm was used as a classification method. Classification was carried out for different neighborhood sizes (from 1 to 6) and accuracy was the parameter determining the quality of a classifier.

The second metric calculated on the basis of Dataset II is gene signature stability. It assesses the reproducibility of the gene signature at all steps of classification. We chose two primary methods: the basic method of calculating this measure and the method, which introduces the adjustment related to the size of the reduced genes [9]. As this article presents new approaches for reduction of gene signatures based on GO and comparing it with other commonly used methods, therefore, the use of the second measure gene signature stability seems to be more appropriate.

4 Results

4.1 Reduction Assessment

Differential expression analysis (see. Methods) of Dataset I has provided a list of significant genes for each tumor. Not all of these genes are annotated to GO Terms therefore, these reduced gene signatures were used for comparison of *classic* routine of GO terms enrichment analysis with selected reduction methods for AA, GBM.P and GBM.S tumor experimental data.

The *classic* approach gives results in line with biological backgrounds of diseases, with numerous GO terms shared by GBM.P and GBM.S (Table 1) and few GO terms shared between AA and GBM. Our method completely removes common GO terms between AA and GBM, where the other methods with decorrelation step preserve a certain proportion of these terms. Some of the GO terms may be linked to different tumors, which is associated with low biological information. Removal of such GO terms can be useful in the tumor classification tasks, or for better understanding the characteristics of the specific tumor. As a result, the filtering performed by our new algorithm leaves only GO terms specific to the particular case.

The summarization of the major changes in resulted lists of GO terms for GBM tumors can be seen in Table 1. We present here what fraction of overrepresented GO terms is reduced for GBM.P, GBM.S and the common part. Additionally, it may be seen that our method radically reduced the common part of GBM, which can be helpful in tumour classification or retrieving important information related to the inspected diseases. In this way, it can be helpful eg. in drug discovery.

More details about influence of the reduction method can be seen on distribution of information content (IC, see Methods) for each of the used methods (Fig. 2). Table 2 presents complementary information with the percentage of enriched GO Terms which have IC ratio greater than 0.7. This threshold has been selected empirically

Table 1 Summary of size and reduced fraction of overrepresented GO terms for GBM.P, GBM.S and the common part

Method	GBM.P	Reduced GBM.P (%)	Common part of GBM	Reduced common part of GBM (%)	GBM.S	Reduced GBM.S (%)
Classic	409	–	160	–	251	–
Elim	154	62.3	41	74.4	69	72.5
Weight	101	75.3	26	83.8	41	83.7
Weight01	114	72.1	32	80	45	82.1
Lea	129	68.5	35	78.1	55	78.1
Parentchild	244	40.3	79	50.6	149	40.6
New	20	**95.1**	5	**96.9**	16	**93.6**

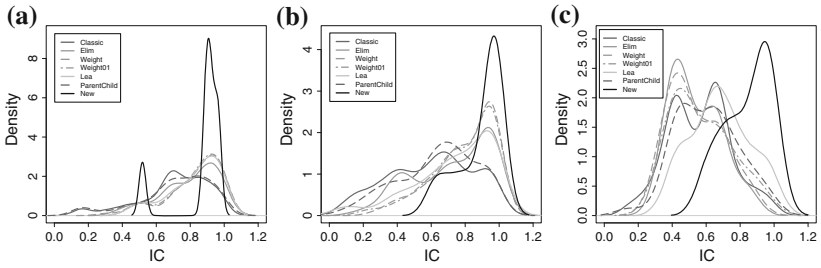


Fig. 2 Probability density function for all methods for each tumor. To estimate the probability density function of the random variable we used nonparametric methods, known as Kernel density estimation (KDE). **a** AA, **b** GBM.P, **c** GBM.S

Table 2 Table contains the percentage of enriched GO terms which have IC ratio greater than 0.7 for all methods and tumors

IC > 0.7	Classic (%)	Elim (%)	Weight (%)	Weight01 (%)	Lea (%)	Parentchild (%)	New (%)
AA	48.2	71.4	78.3	76.6	76.6	41.0	**85.7**
GBM.P	32.4	53.6	65.0	62.8	56.8	37.4	**87.5**
GBM.S	20.1	37.3	45.0	43.2	33.3	21.7	**86.7**
Mean	33.6	54.1	62.8	60.9	55.6	33.4	**86.6**

and reflects the separation of GO terms with high information content from ones with low information content.

It is clear that the shape of PDF for our method is distinct and the least similar to other methods. It has a clear shift of the main peak towards higher IC values, which reflects the reduction of the fraction of GO terms with low information content. This can be clearly seen in Table 2.

Reduction of GO Terms as a side effect leads to reduction of gene signature sizes, which is convenient as it reduces dimensionality of any follow-up analysis. Table 3 presents a summary of gene signature sizes and their reduction for GBM.P, GBM.S and the common part.

4.2 Classification Quality

Comparison based on Dataset II was performed on the reduced size of the gene signatures. Data after summarization steps was used as input features for classification and validation. (see. Methods) Accuracy of classification was presented in the form of a table (Table 4).

Analysis of classification accuracy has shown that the method hierarchy is highly dependent on the neighborhood size we have used (Table 4). Interestingly, our method

Table 3 Summary of size and reduced fraction of gene signatures for GBM.P, GBM.S and the common part

Method	GBM.P	Reduced GBM.P (%)	Common part of GBM	Reduced common part of GBM (%)	GBM.S	Reduced GBM.S (%)
Base	66	–	36	–	48	–
Classic	64	3.0	29	19.4	41	14.6
Elim	53	19.7	23	36.1	35	27.1
Weight	49	25.8	21	41.7	33	31.3
Weight01	51	22.7	22	38.9	34	29.2
Lea	60	9.1	23	36.1	35	27.1
Parentchild	62	6.1	25	30.6	37	22.9
New	27	**59.1**	9	**75.0**	25	**47.9**

Table 4 Summary of classification results of accuracy for each method and different neighborhood sizes

	Classic (%)	Elim (%)	Weight (%)	Weight01 (%)	Lea (%)	Parentchild (%)	New (%)
1	64.3	75.0	72.6	64.3	64.3	64.3	**85.7**
2	70.2	76.2	77.4	71.4	70.2	70.2	**83.3**
3	82.1	**84.5**	**84.5**	81.0	83.3	82.1	81.0
4	84.5	**85.7**	84.5	83.3	**85.7**	84.5	81.0
5	**85.7**	84.5	84.5	**85.7**	**85.7**	**85.7**	79.8
6	**91.7**	88.1	86.9	88.1	90.5	89.3	86.9

has the smallest variability of classification accuracy for all investigated sizes of the neighborhood. It is important to note that accuracy for the neighborhood equal to or greater than 3 for all methods is very similar. Therefore, based just on the accuracy metric, it is hard to determine which method gives the optimal signature for classification purposes.

For this reason, an additional approach was examined—stability of the gene signatures. This measure allows to assess reproducibility of gene signatures, taking into account all steps of the validation process. Results for all methods are shown in Table 5.

The highest score for adjusted stability (stabA) was obtained with our method. The best signature stability together with the lowest accuracy sensitivity to the neighborhood size and with not worse overall accuracy level, indicate that applying our new method provides an improvement toward the composition of the gene signatures.

Table 5 Summary for gene signature stability measures

	Classic (%)	Elim (%)	Weight (%)	Weight01 (%)	Lea (%)	Parentchild (%)	New (%)
StabA	26.18	30.21	30.70	28.16	27.41	26.58	**32.69**

5 Discussion

In the post genomic era, there is an enormity of publicly available molecular biology data, which originates from a variety of biological research projects. These data often include overlapping fields of science. The present state of molecular biology knowledge indicates the possibility of using comprehensive and integrative approaches to understand the biological processes. Therefore, methods for high-throughput data integration are a hot topic of research for molecular biology.

In this article, we have compared our new method with the reference, state-of-the-art algorithms for GO terms enrichment analysis. All reference methods take into account solely information concerning the size of sets (all investigated genes, significant genes and annotated number of genes from first and second group to each GO term). In contrast, our approach takes into account additional information originating from analysis of differential expression of genes. Thanks to this approach we do not omit the valuable information that has been obtained during the differential expression analysis, namely the p-value for each gene.

As previously mentioned there is no well-established procedure for comparison of methods for enrichment analysis of GO terms. Therefore, we have applied several possible approaches aiming at more robust conclusions. On Dataset I we have verified the characteristic of results for each method by means of the cohesion of the results with biological knowledge and quality of the results by examining the information content (IC) of enriched GO terms. Impact of GO terms reduction on gene signatures size reduction was also investigated. We have shown that the all of the methods for enrichment analysis of GO terms with decorrelation step both reduce number of enriched GO terms and maintain fraction of GO terms shared between AA and GBM. The common part of GO terms shared between AA and GBM is completely removed by our method, which in addition filters out GO terms of low Information Content (IC), leaving in only the GO terms which are specific to the particular case. In the end, it can help in tumor classification or can provide more information about characteristic of the tumors by reducing the dimensionality of the follow up analysis.

One of the major aims in molecular biology is to obtain a stable gene signature, which will be helpful in many areas of research. Still there are existing problems such as defining the sizes (numbers of genes) of gene signatures and possible unreliability of results of inference based on gene signatures. For this reason, in the last stage of our comparison we were investigating Dataset II for the classification accuracy as well as the stability of gene signatures. We have shown that our method is characterized by the best signature stability. Together with the lowest sensitivity of classification

accuracy to the used neighborhood size and with not worse overall accuracy level, it indicates that the proposed reduction approach provides an improvement in the composition of the gene signatures in comparison to other methods.

We have shown that our method reduces the number of enriched GO terms, focusing on the meaningful ones. Thus, it facilitates the analysis and biological interpretation. It also contributed to reducing the dimensionality of gene signatures, which ultimately improved accuracy of classification and stability of gene signatures.

Acknowledgments The authors are grateful to Pawel P. Labaj and Anna Papiez for helpful discussions. The work was financially supported by SUT - BKM/525/RAU-2/2014. Calculations were carried out using the infrastructure supported by POIG.02.03.01-24-099/13 grant: GeCONiI - Upper-Silesian Center for Scientific Computation.

References

1. Alexa, A., Rahnenführer, J.: Gene set enrichment analysis with topGO (2009). http://rgm.ogalab.net/RGM-files/R_BC/download/topGO/inst/doc/topGO.pdf
2. Alexa, A., Rahnenführer, J., Lengauer, T.: Improved scoring of functional groups from gene expression data by decorrelating GO graph structure. Bioinformatics **22**(13), 1600–1607 (2006)
3. Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., et al.: Gene Ontology: tool for the unification of biology. Nat. Genet. **25**(1), 25–29 (2000)
4. Barrett, T., Wilhite, S.E., Ledoux, P., Evangelista, C., Kim, I.F., Tomashevsky, M., Marshall, K.A., Phillippy, K.H., Sherman, P.M., Holko, M., et al.: NCBI GEO: archive for functional genomics data sets update. Nucleic Acids Res. **41**(D1), D991–D995 (2013)
5. Bérard, S., Tichit, L., Herrmann, C.: ClusterInspector: a tool to visualize ontology-based relationships between biological entities. Actes des Journées Ouvertes Biologie Informatique Mathématiques, pp. 447–457. Lyon (2005)
6. Bielen, A., Perryman, L., Box, G.M., Valenti, M.,de Haven Brandon, A., Martins, V., Jury, A., Popov, S., Gowan, S., Jeay, S., et al.: Enhanced efficacy of IGF1R inhibition in pediatric glioblastoma by combinatorial targeting of PDGFRα/β. Mol. Cancer Ther. **10**(8), 1407–1418 (2011)
7. Camon, E., Magrane, M., Barrell, D., Lee, V., Dimmer, E., Maslen, J., Binns, D., Harte, N., Lopez, R., Apweiler, R.: The gene ontology annotation (goa) database: sharing knowledge in uniprot with gene ontology. Nucleic Acids Res. **32**(suppl 1), D262–D266 (2004)
8. Dai, M., Wang, P., Boyd, A.D., Kostov, G., Athey, B., Jones, E.G., Bunney, W.E., Myers, R.M., Speed, T.P., Akil, H., et al.: Evolving gene/transcript definitions significantly alter the interpretation of GeneChip data. Nucleic Acids Res. **33**(20), e175–e175 (2005)
9. Davis, C.A., Gerick, F., Hintermair, V., Friedel, C.C., Fundel, K., Küffner, R., Zimmer, R.: Reliable gene signatures for microarray classification: assessment of stability and performance. Bioinformatics **22**(19), 2356–2363 (2006)
10. Figueroa, M.E., Wouters, B.J., Skrabanek, L., Glass, J., Li, Y., Erpelinck-Verschueren, C.A., Langerak, A.W., Lowenberg, B., Fazzari, M., Greally, J.M., et al.: Genome-wide epigenetic analysis delineates a biologically distinct immature acute leukemia with myeloid/T-lymphoid features. Blood **113**(12), 2795 (2009)
11. Galperin, M.Y., Rigden, D.J., Fernández-Suárez, X.M.: The 2015 nucleic acids research database issue and molecular biology database collection. Nucleic Acids Res. **43**(D1), D1–D5 (2015)

12. Grossmann, S., Bauer, S., Robinson, P.N., Vingron, M.: Improved detection of overrepresentation of Gene-Ontology annotations with parent-child analysis. Bioinformatics **23**(22), 3024–3031 (2007)
13. Grzmil, M., Morin, P., Lino, M.M., Merlo, A., Frank, S., Wang, Y., Moncayo, G., Hemmings, B.A.: MAP kinase-interacting kinase 1 regulates SMAD2-dependent TGF-β signaling pathway in human glioblastoma. Cancer Res. **71**(6), 2392–2402 (2011)
14. Homma, T., Fukushima, T., Vaccarella, S., Yonekawa, Y., Di Patre, P.L., Franceschi, S., Ohgaki, H.: Correlation among pathology, genotype, and patient outcomes in glioblastoma. J. Neuropathol. Exp. Neurol. **65**(9), 846–854 (2006)
15. Hunter, S., Apweiler, R., Attwood, T.K., Bairoch, A., Bateman, A., Binns, D., Bork, P., Das, U., Daugherty, L., Duquenne, L., et al.: InterPro: the integrative protein signature database. Nucleic Acids Res. **37**(suppl 1), D211–D215 (2009)
16. Hutter, J.J.: Childhood leukemia. Pediatr. Rev. **31**(6), 234–241 (2010)
17. Kanehisa, M., Araki, M., Goto, S., Hattori, M., Hirakawa, M., Itoh, M., Katayama, T., Kawashima, S., Okuda, S., Tokimatsu, T., et al.: KEGG for linking genomes to life and the environment. Nucleic Acids Res. **36**(suppl 1), D480–D484 (2008)
18. Krupp, M., Itzel, T., Maass, T., Hildebrandt, A., Galle, P.R., Teufel, A.: Cell LineNavigator: a workbench for cancer cell line analysis. Nucleic Acids Res. **41**(D1), D942–D948 (2013)
19. Liu, Z., Yao, Z., Li, C., Lu, Y., Gao, C.: Gene expression profiling in human high-grade astrocytomas. Comp. Funct. Genomics (2011)
20. Louis, D.N., Ohgaki, H., Wiestler, O.D., Cavenee, W.K., Burger, P.C., Jouvet, A., Scheithauer, B.W., Kleihues, P.: The 2007 WHO classification of tumours of the central nervous system. Acta Neuropathol. **114**(2), 97–109 (2007)
21. Nat. Biotechnol. A comprehensive assessment of RNA-seq accuracy, reproducibility and information content by the sequencing quality control consortium. **32**(9), 903–914 (2014)
22. Nucleic Acids Res. The universal protein resource (UniProt) in 2010. **38**(suppl 1), D142–D148 (2010)
23. Ohgaki, H., Kleihues, P.: The definition of primary and secondary glioblastoma. Clin. Cancer Res. **19**(4), 764–772 (2013)
24. Sandberg, R., Larsson, O.: Improved precision and accuracy for microarrays using updated probe set definitions. BMC Bioinform. **8**(1), 48 (2007)
25. Stewart, B.W., Wild, C.P.: World cancer report 2014. IARC Press, International Agency for Research on Cancer, Geneva (2008)
26. Vardiman, J.W., Thiele, J., Arber, D.A., Brunning, R.D., Borowitz, M.J., Porwit, A., Harris, N.L., Le Beau, M.M., Hellström-Lindberg, E., Tefferi, A., et al.: The 2008 revision of the World Health Organization (WHO) classification of myeloid neoplasms and acute leukemia: rationale and important changes. Blood **114**(5), 937–951 (2009)
27. Wang, C., Gong, B., Bushel, P.R., Thierry-Mieg, J., Thierry-Mieg, D., Xu, J., Fang, H., Hong, H., Shen, J., Su, Z., et al.: The concordance between RNA-seq and microarray data depends on chemical treatment and transcript abundance. Nat. Biotechnol. **32**(9), 926–932 (2014)

Automatic PDF Files Based Information Retrieval System with Section Selection and Key Terms Aggregation Rules

Rafal Lancucki and Andrzej Polanski

Abstract Standard approaches to knowledge extraction from biomedical literature focus on information retrieval from abstracts publicly available in medical databases like PubMed. To limit the number of the results initially, a suitable query against such databases can be constructed. However, for many research topics the pre-selection of small enough set of the documents can be very difficult or even impossible. Another problem stems from large variability of the retrieved lists of publications when changing keywords in search engines. In this paper we address both of these problems by proposing an algorithm and an implementation capable of working on the full text articles. We present an information retrieval system with selection of separate sections of full texts of papers and a rule-based search engine. We demonstrate that in some research our solution can provide much better results than finding documents only by keywords and abstracts.

Keywords Knowledge discovery · Information extraction · Natural language processing · Keyword search

1 Introduction

Motivation Important projects concerning developing and maintaining scientific databases involve information retrieval from scientific papers. A routine strategy consists in using a (dedicated) search engine for selecting a corpus of texts, whose contents are then studied in detail (manually curated). However, using such strategies may lead to difficulties and errors. There are two main sources of errors in information retrieval in such scenarios. The first one is related to the fact that specifying sets of keywords for searching through abstracts is typically done on an intuitive background and adding/removing a word from a set of keywords may lead to large differences

R. Lancucki (✉) · A. Polanski
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: lancucki@onet.pl

A. Polanski
e-mail: andrzej.polanski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_21

in the sizes and structures of corpora of documents to be analyzed. The second, more important source of errors, is that engines for searching through scientific documents typically arrange (rank) the retrieved documents according to their "level of importance". Such arrangement is, unfortunately, very imprecise. Due to a large number of retrieved papers, their analyses are limited only to a few documents at the top of the list. Ignoring documents ranked lower may lead to overlooking of important information.

Moreover, manual curation of scientific literature documents for information extraction is a tedious and expensive process, which can again cause a lot of hard to verify errors. Therefore, developing methodologies capable of performing deeper searches in texts of scientific documents can lead to substantial improvements in processes of literature studies in the scientific research.

Contribution We present a system for supporting information extraction from scientific pdf documents based on automatic selection of paragraphs, marking important fragments (words, phrases) and navigation through the selected corpus of papers.

Our contribution is a practical use of several known techniques in one working solution. The first step of our information retrieval strategy involves execution of a query against PubMed database, retrieval of the list of available articles and downloading full texts of these articles from the PubMed FTP servers to the local repository. Available texts are then processed by the rule-based engine at the level of the source PDF document. If interesting connections are found, a copy of the source document is created with annotated keywords (comments inside PDF document). During the process several statistics for each document are also created, so it is possible to rank the documents by their estimated importance. For the statistics we also created an additional viewer, where basic aggregation and arrangement of results is possible.

We provide evidence that the proposed method can be significantly better than traditional search engines. We illustrate the application of our system in relation to the literature study on the genetic background for type 1 diabetes by searching for paragraphs containing statements on single nucleotide polymorphisms (SNP) markers of type 1 diabetes [7] in the corpus of full texts of documents. The problem of characterization of the background of type 1 diabetes by lists of SNPs is complex because of limited data on some SNPs and demographic specificity. Therefore, all findings must be later verified manually. Available search engines are not constructed to fulfil such complex requirements.

Our application exhibits higher precision in retrieving corpora of documents of closest relation to a topic of interest than standard search engines. It also shows higher potential in providing lists of articles containing sets of keywords thanks to an "alternative" path of searching. Standard search engines have obvious limitations related to confining searches to abstracts only.

Another feature of the elaborated system is searching through scientific literature on the genetic background of diseases, which includes a large number of papers reporting results of many different experiments.

2 Related Work

There is an increasing number of papers and related software systems devoted to supporting the projects of searching through scientific literature [1–4, 6, 9, 10]. Available applications might not offer the possibility of interactive searching through documents on the basis of arbitrarily defined aggregation rules for key terms (phrases) combined with navigation and marking important fragments of the texts. Some of the systems works over pre-defined rules [5, 14] but they are focused on automatic text extraction rather than on supporting manual reviews of the scientific literature. Other systems could support the task we faced, but they were too general in approach to be used in this specific research [8, 13].

3 Description of the Solution—Rule-Based Knowledge Discovery

The requirements for the elaborated software system, specified with respect to the above-described applications, were to simplify the visual extraction process by pre-selecting paragraphs. An additional requirement was that the input data are provided in the pdf format.

The rule is defined by the user in the following way:

- divide source document into logical sections
- for each section find T1D or type 1 diabetes—if found, count occurrences
- if found, find rs[integer_number]—if found, add the document to the result set and annotate it.

3.1 Pre-selection

First step is to execute the query against the PubMed database, compare the results with the list of files available inside PubMed database and download them. Due to the size of source documents this is a lengthy process—for type 1 diabetes it looked as follows:

- query executed against the PMC database—"type 1 diabetes"
- number of results returned by query—32,249 documents
- number of available free text documents—14,283 documents
- size of downloaded document corpus—12.7 GB—downloading time 24 h (depends on connection speed)

3.2 Description of the Algorithm

The software is implemented using the pdf-box library [11, 12], as a low-level kernel to access pdf files. Pdf-box capabilities are also used to modify source files during creation of comments inside the output file. Inside the software an additional layer was created to solve the problem of word and paragraph breaks. The pre-analysis implies working over low level of pdf document trying to identify paragraphs by position of characters tracking and different fonts used in the document. If the defined rules are fulfilled the found keywords are highlighted and the document is copied into the results folder and the statistics are updated. It was clear from the beginning that fully automated process will be very hard to achieve. Thus, the approach aimed to automate the process as much as possible, by highlighting searched fragments. Statistics created during the process help in later sorting and displaying the results. The solution includes the following steps:

- In the first step, conversion of text to new object-model structure containing words, sentences and paragraphs is performed. After this conversion a plain text is available for the search engine, but every character still has a reference to its original position inside the PDF document
- In the second step, the configuration of the program is read out (rules definition). It is possible to define obligatory keywords and optional keywords inside the rules. This is a key feature of the program, which allows for its configuration for different purposes according to the current needs of the user.
- Found elements are highlighted inside the original document (copy of the original pdf file is created with added comments). For every paragraph inside the document one comment is created with highlighted keywords, while inside the comment—the defined text keywords are listed.
- Additionally, the statistics are updated inside the database. These statistics are used to display the results in the form of ranking, and to compare the achieved results with different queries executed against PubMed database.

Below we show an example of the sentence found by rules defined for type 1 diabetes (PMC article 3674006):

> Genotyping was done using Taqman allele discrimination (KBioscience, Hoddesdon, UK). The 31 selected **T1D** SNPs were: [24]: INS (**rs3842753** and **rs689**), PTPN22 (**rs2476601**), PTPN2 (**rs478582** and **rs1893217**), [...]

The process described above was executed on all previously downloaded documents. Because of the input corpus size, it took nearly 10 hours to parse and analyze all documents. As a result 476 output PDF documents were created with annotations inside (marked keywords) along with additional database with statistics/search results. Additional software to aggregate and present the results was created, making it possible to sort and view them grouped by SNPs. This presentation is very important during manual verification of the results—it is also possible to open the annotated source documents from this level.

4 Results

As an input for automated process 14,283 documents downloaded from the PubMed central PMC database were used. Those documents were retrieved by querying PubMed database with query "type 1 diabetes" and matching the list of results with the list of documents available on the PubMed Central FTP server. The results are presented from two different perspectives:

1. as document lists with statistics on how many keywords were found inside a document and with the possibility to compare it with PubMed results. In this view, as details for each document, the list of found SNPs is presented (See Table 2 on page 254).
2. as a list of found SNPs specifying the the number of publications, in which the results were found and the list of those publications. This view is particularly important since it allows for fast verification of each result. Here, 1618 single SNPs were found, but only 302 were confirmed inside more than 1 publication. (See Table 1 on page 253).

Thanks to this second view of the results, one might easily perform the manual verification of them. For each SNP, the reviewer/curator has a list of citing publications. In every publication, the occurrences of SNP and type 1 diabetes are highlighted,

Table 1 Top 20 results—SNPs sorted by number of citing publications in descending order

SNP	Number of publications
rs2476601	45
rs1990760	26
rs3184504	19
rs689	18
rs3087243	17
rs13266634	17
rs11594656	15
rs6679677	14
rs2104286	14
rs231775	13
rs6897932	13
rs9939609	13
rs12708716	12
rs6822844	12
rs7903146	12
rs2542151	12
rs2292239	10
rs7574865	10
rs1893217	9
rs763361	9

and thanks to native PDF comments navigation it is easy to jump into the fragment of interest. The speed of verifying and retrieving results is incomparable with any manual solution.

Results Verification To verify our results, the application can run a query against PubMed and compare the returned set of results with our results. For results verification, we used query "type 1 diabetes AND SNP", what should return similar results as we achieved using our solution. This query returns 3414 results. By default, the results are sorted by their relevance, so we expected some similarity between what we have as results and what was returned by PubMed database. But there are many examples where PubMed position within the results (sorted by relevance) is far from top (See Table 2 on page 240). The application rank was created by sorting the number of type 1 diabetes occurrences inside a document in descending order (we assumed that the more times a disease is mentioned inside a publication, the more interesting it is for us). We also performed cross validation in the other way round—for some of the PubMed results we verified why documents do not exists in our result set. We found generally 2 cases:

Table 2 List of top 20 from documents view—sorted by the number of occurrences type 1 diabetes descending and compared with the results of PubMed query "type 1 diabetes AND SNP"

Document ID	T1D occurrences	Number of found SNPs	PubMed result position
PMC4140826	187	58	418
PMC3900458	156	2	Not found by PubMed
PMC2889752	154	46	11
PMC3924830	151	19	138
PMC2661594	144	2	10
PMC3150451	137	80	30
PMC3219940	121	3	5
PMC3342174	112	20	162
PMC3789883	110	17	57
PMC2217539	107	31	66
PMC2602853	99	93	206
PMC3322139	94	1	67
PMC3363338	90	10	279
PMC3114708	85	65	377
PMC3075244	83	4	70
PMC3292339	81	53	40
PMC4038470	81	14	261
PMC2738846	79	26	52
PMC2858242	78	64	127
PMC3183083	77	8	54

- rs[integer_number] text does not exist inside full text article - this is a false positive result of PubMed query—it is good that it does not exist in our result set
- we found few cases where rs[integer_number] exists inside an article, but because it is not within one section with type 1 diabetes, it was not found. However, it is still important publication for the results. It is a true negative of our approach and in the future, we should consider how to eliminate this problem. But this problem did not affect our results in this particular research, while SNPs were found in different publication.

Unexpected Results Surprisingly, there are also many documents occupying very low positions in the PubMed results ranking, despite the fact that they directly focus on diagnosing type 1 diabetes and its genetic background. For example, in PMC3674006, the words 'T1D' and 'SNP' exist in one sentence, directly one after another. However, the document was placed by the PubMed search engine on the 322nd position—much lower than other papers much less related to the topic of the search. The document is on the 10th result page, in the standard view of 30 results per single page. Considering why such situation could happen we concluded the following possibilities:

- standard search engine does not work on full publication text, but only on abstract or keywords defined by authors
- standard search engine has some unclear rules, how the ranking is created from the results

In both cases our rule-based method with clear presentation of results could have potential in other research as well.

5 Conclusions

In this paper we presented an application aiding information retrieval from scientific pdf documents. The system is based on the partition of documents into paragraphs and on the automated selection of paragraphs, highlighting important fragments (words, phrases) with the navigation through the selected corpus of papers. An example of application of the elaborated system is searching through scientific literature on the genetic background of diseases, which includes a large number of papers reporting results of many different experiments.

We combined known so far techniques, but we combined them in a unique way developing practically working solution. Additionally our contribution is a way of presenting results, where it is very simple to decide, whether a result is proper or not during manual verification (highlighting keywords inside documents, view of a result by keywords with citing the publication). In this paper we proved that our approach is significantly better than relying on PubMed search engine and later manual verification of the results.

Future Work During our work we found out that the solution we proposed could be also used for different research as well. It shows great potential especially in case the subject of research is known, but it cannot be precisely defined (like rs[integer_number]). While some potential can be also explored in terms of standard queries against medical databases, our proposed result presentation seems to have more benefits than standard search engines methodology.

Acknowledgments This paper was partially financially supported by the NCN Opus grant UMO-2011/01/B/ST6/06868 to AP. Computations were performed with the use of the infrastructure provided by the NCBIR POIG.02.03.01-24-099/13 grant: GCONiI - Upper-Silesian Center for Scientific Computations.

References

1. Agarwal, S., Yu, H.: Figure summarizer browser extensions for pubmed central. Bioinformatics **27**(12), 1723–1724 (2011)
2. Bhattacharya, S., Ha-Thuc, V., Srinivasan, P.: MeSH: a window into full text for document summarization. Bioinformatics **27**(13), 120–128 (2011)
3. Chiang, J.H., Shin, J.W., Liu, H.H., Chin, C.L.: Genelibrarian: an effective gene-information summarization and visualization system. BMC Bioinform. **7**(1), 392–401 (2006)
4. Cohen, K.B., Johnson, H.L., Verspoor, K., Roeder, C., Hunter, L.E.: The structural and content aspects of abstracts versus bodies of full text journal articles are different. BMC Bioinform. **11**(1), 492–501 (2010)
5. Divoli, A., Attwood, T.: Bioie: extracting informative sentences from the biomedical literature. Bioinformatics **21**(9), 2138–2139 (2005)
6. Fundel, K., Küffner, R., Zimmer, R.: RelEx-Relation extraction using dependency parse trees. Bioinformatics **23**(3), 365–371 (2007)
7. Howson, J.: Analysis of 19 genes for association with type i diabetes in the type i diabetes genetics consortium families. Genes Immun. (2009)
8. Hur, J., Schuyler, A., States, D., Feldman, E.: Sciminer: web-based literature mining tool for target identification and functional enrichment analysis. Bioinformatics **25**(6), 838–40 (2009)
9. Krallinger, M., Valencia, A., Hirschman, L.: Linking genes to literature: text mining, information extraction, and retrieval applications for biology. Genome Biol. **9**(Suppl 2) (2008)
10. Krallinger, M., Leitner, F., Valencia, A.: Analysis of biological processes and diseases using text mining approaches. In: Matthiesen, R. (ed.) Bioinformatics Methods in Clinical Research, Methods in Molecular Biology, vol. 593, pp. 341–382. Humana Press, New York (2010)
11. Litchfield, B.: Making PDFs Portable: Integrating PDF and Java Technology (2014). http://java.sys-con.com/node/48543
12. McCandless, M., Hatcher, E., Gospodnetić, O.: Lucene in action. Manning Publications Co., Shelter Island (2010)
13. Papanikolaou, N., Pavlopoulos, G., Pafilis, E., Theodosiou, T., Schneider, R., Satagopam, V., Ouzounis, C., Eliopoulos, A., Promponas, V., Iliopoulos, I.: Biotextquest(+): a knowledge integration platform for literature mining and concept discovery. Bioinformatics **30**(22), 56–3249 (2014)
14. Šarić, J.: Extraction of regulatory gene/protein networks from medline. Bioinformatics **22**(6), 645–650 (2006)

Influence of Introduction of Mitosis-Like Processes into Mathematical-Simulation Model of Protocells in RNA World

Dariusz Myszor

Abstract In this article influence of operation of mitosis–like mechanisms, on hybrid simulation–mathematical model of protocells development in RNA world, was presented. Introduced processes are responsible for even distribution of genetic material, between progeny formations, during protocells division phase. Obtained results point out that such mechanisms might improve abilities of genetic information storage by population of protocells, thus they can support process of life emergence, however high fidelity of genes replication is required. Therefore, emergence of genes assortment process should follow improvement of RNA replicase abilities in the area of speed and replication reliability.

Keywords Branching processes · RNA world · Monte Carlo simulations

1 Introduction

Currently RNA World is the most popular hypothesis concerning emergence of life on Earth [11]. According to RNA world proponents, once there was the time when key processes related to sustainment of life, such as replication and information storage, were based on RNA molecules. Execution of these functionalities was possible because of RNA strands properties. Some RNA particles can exhibit catalytic abilities, that are essential for the process of replication of molecules, in addition single strand structure of RNA molecules facilitates information storage [1, 4]. There are many potential evidences of occurrence of such RNA based phase during evolution of life on Earth e.g. ribosomes which are probably direct descendants from RNA world era [19]. However, there are also certain unknowns, which must be solved in order to confirm existence or RNA world. The most important issue is the amount of information which can be stored by such formations. Replication of long strands requires specialized RNA molecules (RNA replicase) which, on the other hand, require storage of sufficient amount of information in order to be constructed. Huge problem is

D. Myszor (✉)
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: dariusz.myszor@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_22

antiquity of processes of life creation, direct signs were vanished by processes ongoing on Earth. Therefore, various computer and mathematical models were created in order to explain process of evolution from the simplest forms of RNA molecules to the more advanced system, in which molecules interplay which each other and obtain self organisation [9]. Utilization of these models could support laboratory works and, what is especially important, is time and cost efficient.

As mentioned before important issue of RNA world hypothesis is the amount of information that can be stored by primordial formations [5, 17]. RNA molecules were diluted in primordial soup, therefore obtainment of material organisation was difficult. On the other hand, contemporary researches point out that primitive cell-like formations (also knows as protocells or packages), that were composed of phospholipids membranes [13, 14] which enclosed genetic material [2], might play important role during emergence of life on Earth [10, 17]. Therefore, modifications were introduced into original simulation model of protocells development created by Niesert et. al. [3, 15, 18]. Limitations of original model, in the area of number of protocells which can be hold by a single population, were rule out through application of mathematical apparatus based on branching processes. The purpose of this work was to check whether introduction of mitosis–like mechanisms, in the form of genes segregation during protocell division process, and obtainment of even distribution of genetic material on the progeny cells, could support increase of the amount of information that can be stored by population of such entities.

2 Model Description

Model simulates growth and decay of populations of protocells. Each protocell contains set of genetic material—genes—which are enclosed by primitive membrane. Every gen is represented by a single RNA strand which contains specific sequence of nucleotides. Genes which posses the same sequence of nucleotides belong to the same type of gene class. In order to be viable, thus to be a member of the population, protocell must contain at least one representative of each different type of gene (abbreviated as DTOG in text and denoted as D in equations), from the predefined pool of required types of genes. In addition there might be multiple replicas of the same type of gene in a single package. Set of required types of genes is defined at the beginning of the simulation. Packages that are not viable are eliminated from the population and are no longer simulated.

RNA molecules, that are enclosed in protocells, are being replicated in a random process. Each gene located in the package can be replicated with equal probability, in addition multiple replications of the same gene are possible. It is assumed that after replication of certain number of molecules, internal pressure within the package causes division of the cell into two progeny formations and random distribution of genetic material between progeny entities (parameter which is responsible for determination of the number of genes that should be replicated between division of a cell is denoted as N in equations and abbreviated as NORM in text). Process of

replication of molecules is characterized by limited fidelity. There are two types of mutations, which are characterized by various effects, that are caused by mutated molecules, on the level of protocells:

- Lethal mutations—introduce molecules which cause direct death of the package e.g. as a result of protocell wall breach. Probability of occurrence of lethal mutation, during the process of molecules replication, is denoted as p_l.
- Parasite mutations—introduce molecules that do not realize their functionality. As long as there are other molecules which can fulfil required functionality they do not have direct negative effect on the package viability. However, parasite molecules (the same as regular genes) are subject of replication. As a result existence of parasites limits ability of replication of proper genes, thus it leads to the creation of not viable progeny packages during division process. Probability of occurrence of parasite mutation during the process of molecules replication, is denoted as p_p.

Protocells were subject to harmful events such as UV radiation, which can render them not viable. This process is taken into account by introduction of accident coefficient into the model. Accident coefficient introduces random elimination (with probability equal to p_a) of protocells form the population.

In foster conditions number of protocells in consecutive generations increases exponentially, it imposes huge computational demand and increases significantly results await time. Therefore, original model introduces limitation of three individuals per population. In current researches this value was significantly increased, up to 10000 protocells that can be hold by a single population. After simulation of every generation the number of viable protocells is determined and if there are more than 10000 individuals, prospective coefficient (V) is applied and the weakest progenies are being eliminated. Prospective coefficient is implemented in simulation part of the model and is defined by following equation:

$$V = D^D \left(\prod_{i=1}^{D} \frac{n_i}{S} \right) \left(\frac{S}{T} \right)^D \log_2(S - D + 2) \tag{1}$$

where D is the number of different gene types, n_i stands for the number of copies of i^{th} gene, n_p is the number of parasites in a package, $S = \sum_{i=1}^{D} n_i$ and finally $T = S + n_p$.

The purpose of the model is determination of maximum amount of information, expressed in maximum number of different types of genes (abbreviated as MDTOG) with respect to NORM, that can be stored by population of such formations. In order to obtain above mentioned data, series of computer simulations, followed by application of mathematical apparatus, were performed.

Initial state of the system is unknown, therefore every individual from the first population is initialized with arbitrary data. Number of RNA molecules in each package is equal to $D\lceil \frac{2N}{D} \rceil$, every DTOG has equal number of representatives in every cell. Initial population contains 25 protocells. In such a case, in order to obtain meaningful results from Monte Carlo simulations, warm up period has to be taken into

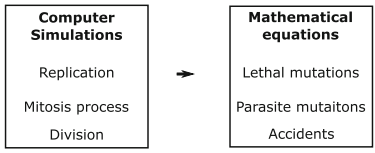


Fig. 1 Division of responsibility over modelled phenomenons between simulation and mathematical parts

account. Simulations conduced in previous work [15, 16] pointed out that minimum number of generations for which simulations shall be conducted, in order to obtain stabilisation and rule out influence of system initialization, is at the level of 1000. At the same time each simulation scenario should be repeated independently 40 times (Fig. 1).

3 Mitosis Process

Novelty in the presented system was introduction of mitosis–like processes, which are required for the purpose of even segregation of genes between progeny packages. In contemporary cells mitosis is one of the phase of cell cycle process, during which chromosomes located in cell nucleus are separated into two identical sets of DNA strands, each enclosed by cell nuclei. Contemporary cells cannot divide efficiently without this phase. The purpose of implemented modifications was to check whether introduction of similar processes at the level of primitive cell–like entities, would lead to improvement of abilities of these formations to store genetic information. Processes related to mitosis, realized in contemporary cells, require complex apparatus, however they provide high fidelity of division process. In case of prebiotic processes obtainment of such level of complexity, thus high level of fidelity, would not be possible. Therefore, mitosis fidelity coefficient was introduced into the model (M) which expresses fraction of the genetic material that is divided in an even way between daughter packages. As a result, when parental package possesses g_p representatives of given gene type, each daughter package obtains $g_d = \lfloor g_p \cdot M/2 \rfloor$ ($g_d \in \mathbb{N}$) molecules from mitosis based process for every type of gene. After an even genes assortment process, remaining part of gene pool from parental package is divided randomly between progeny cells.

4 Mathematical Analysis

Mathematical method, based on branching processes [6–8], was utilized in order to limit number of simulations that had to be performed and to significantly reduce results generation time. Exact description of equations presented in this section is provided in Myszor 2011 [16]. Branching processes can be employed because protocells simulated in the model are independent from each other and because parameters of

the simulation process do not change during simulation of a single scenario. In order to apply this method, series of 100 independent simulations were conducted for every triplet of NORM, DTOG and M values, from the ranges of $N \in \{1, 2, \ldots, 100\}$, $D \in \{1, 2, \ldots, 15\}$ and $M \in \{0, 0.1, \ldots, 1\}$. During simulation of subsequent scenarios accidents and mutations were disabled ($p_a = 0$, $p_p = 0$, $p_l = 0$). Each simulation was running for 10000 generations. At the end of each simulation information about mean values of probability that after the division process, parental protocell will generate none ($p_{0,N,D,M}$), one ($p_{1,N,D,M}$) or two ($p_{2,N,D,M}$) viable daughter packages were collected. Each protocell has exactly two daughter packages therefore, condition $p_{0,N,D,M} + p_{1,N,D,M} + p_{2,N,D,M} = 1$ is always fulfilled (for notation convenience D and M subscripts were omitted in the consecutive equations). Then equations responsible for inclusion of accidents as well as parasite mutations influence, were utilized

$$p_{0pa,N} = p_{0,h(N)} + p_{1,h(N)}p_a + p_{2,h(N)}p_a^2, \tag{2}$$

$$p_{1pa,N} = p_{1,h(N)} - p_{1,h(N)}p_a + 2p_{2,h(N)}p_a(1 - p_a), \tag{3}$$

$$p_{2pa,N} = p_{2,h(N)} - 2p_{2,h(N)}p_a(1 - p_a) - p_{2,h(N)}p_a^2, \tag{4}$$

where $p_{0pa,N}$, $p_{1pa,N}$, $p_{2pa,N}$ are probabilities that the parental package has none, one or two viable daughters after division process when accidents and parasite mutations are operating; $h(N)$ coefficient is effective NORM for given value of parasite mutation probability (for details see [16]). Above mentioned equations do not include lethal mutation, in order to introduce this process, additional set of formulas must be introduced, based on coefficient obtained through calculation of Eqs. 2–4.

$$p_{0pal,N} = p_{0pa,N} + 2p_{1pa,N}(1 - (1 - p_l)^N) + p_{2pa,N}(1 - (1 - p_l)^N)^2, \tag{5}$$

$$p_{1pal,N} = p_{1pa,N} - p_{1pa,N}(1-(1-p_l)^N) + 2p_{2pa,N}(1-(1-p_l)^N)(1-p_l)^N, \tag{6}$$

$$p_{2pal,N} = p_{2pa,N} - 2p_{2pa,N}(1-(1-p_l)^N)(1-p_l)^N - p_{2pa,N}(1-(1-p_l)^N)^2, \tag{7}$$

where $p_{0pal,N}$, $p_{1pal,N}$, $p_{2pal,N}$ are probabilities that the parental package has none, one or two viable descendants after division process, for the scenario in which accidents, parasite and lethal mutations can be activated. For convenience, following unified notation can be introduced that replaces factors $p_{0pal,N} - p_{2pal,N}$

$$p_{r,N}^F = p_{rpal,N} \tag{8}$$

where $p_{r,N}^F$, is the probability of possession of r alive descendants ($r \in \{0, 1, 2\}$), when all deleterious processes might be activated. Then probability generating functions are applied in order to calculate mean amount of viable packages obtained after protocell division process in given scenario

$$f_N(s) = p^F_{0,N} + p^F_{1,N}s + p^F_{2,N}s^2. \tag{9}$$

In the next step mean value of viable descendants (μ) is calculated with following equation

$$\mu = f_N(1)' = p^F_{1,N} + 2p^F_{2,N}. \tag{10}$$

Based on criticality property of branching processes, if $\mu > 1$ then the probability of extinction of the population is smaller than 1, on the other hand where $\mu \leq 1$, probability of population vanishing is equal to 1 [12]. As a result determination of the fate of the population, at the level of long term survival abilities, can be determined.

5 Results

For each triplet of NORM, MDTOG and M, 100 independent simulations were conducted, as a result 100 values of μ were obtained. T-test was utilized for the purpose of statistical analysis of results obtained from branching process based method. Null hypothesis (H_0) stated that mean number of viable descendants is equal to one and alternative hypothesis (H_1) stated that mean number of descendants is greater than one. On the graph for each analysed value of NORM parameter, MDTOG was marked for which H_0 was rejected at the significance level $\alpha = 0.05$.

Initial test of influence of introduction of mitosis-like processes was conducted for the scenario with disabled accidents and mutations. Obtained results point out that for low values of M, at the level of 0.1, there is no visible benefit in the increase of information storage capabilities for the analysed system (unpublished data because of space limitations). For higher values of M there is visible increase in the number of DTOG that can be hold by such a population (Fig. 2). With higher NORM influence

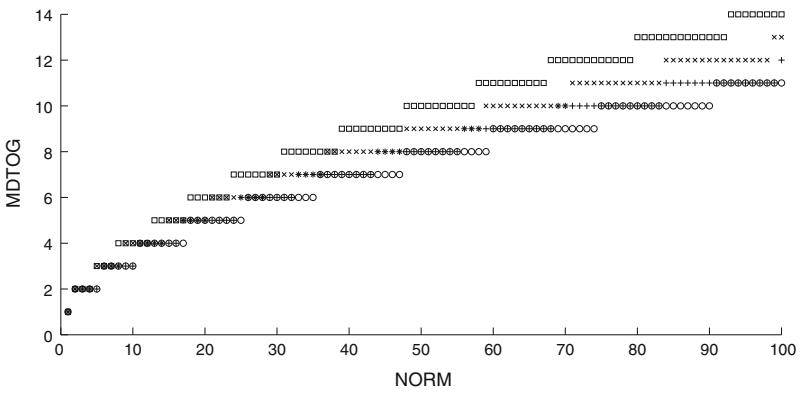


Fig. 2 Maximum number of different types of genes MDTOG as a function of number of replicated molecules NORM. Results obtained for mitosis coefficient M equal to 0 (◯), 0.3 (+), 0.6 (×) and 1 (□); $p_a = 0$, $p_p = 0$ and $p_l = 0$

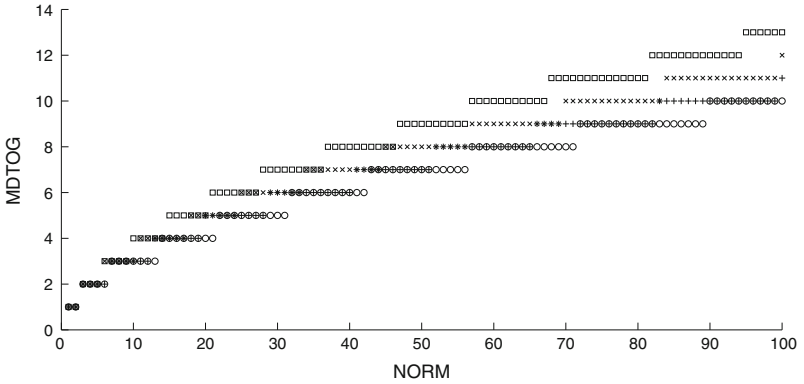


Fig. 3 Maximum number of different types of genes MDTOG as a function of number of replicated molecules NORM. Results obtained for mitosis coefficient M equal to 0 (◯), 0.3 (+), 0.6 (×) and 1 (□); $p_a = 0.1$, $p_p = 0$ and $p_l = 0$

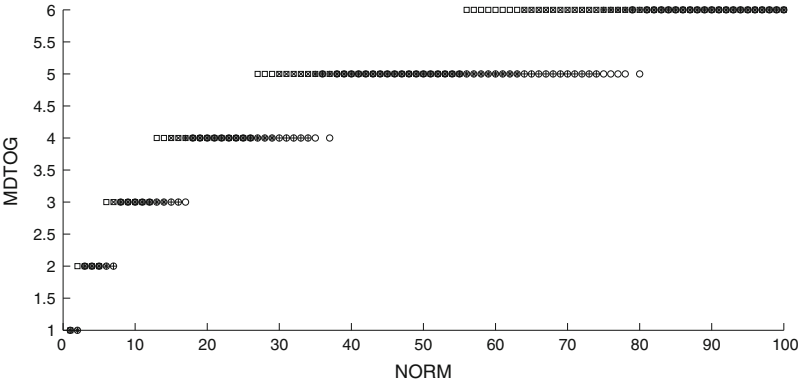


Fig. 4 Maximum number of different types of genes MDTOG as a function of number of replicated molecules NORM. Results obtained for mitosis coefficient M equal to 0 (◯), 0.3 (+), 0.6 (×) and 1 (□); $p_a = 0$, $p_p = 0.1$ and $p_l = 0$

of even division of genetic material between progeny cells is more visible. Similar effects are visible for case with accidents operating (Fig. 3) and lethal mutations operating (Fig. 5). On the other hand parasite mutations, operating at realistic levels [15], seem to limit influence of analysed processes of even division of genetic material (Fig. 4).

In the final step all deleterious processes were operating (Fig. 6), in such a case even for the value of M equal to 1, there is practically no gain achieved from operation of mitosis-like processes in the area of amount of information that can be stored by such populations.

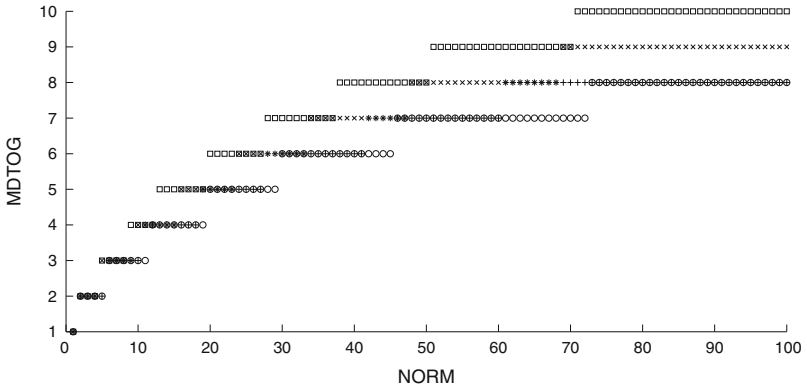


Fig. 5 Maximum number of different types of genes MDTOG as a function of number of replicated molecules NORM. Results obtained for mitosis coefficient M equal to 0 (◯), 0.3 (+), 0.6 (×) and 1 (□); $p_a = 0$, $p_p = 0$ and $p_l = 0.005$

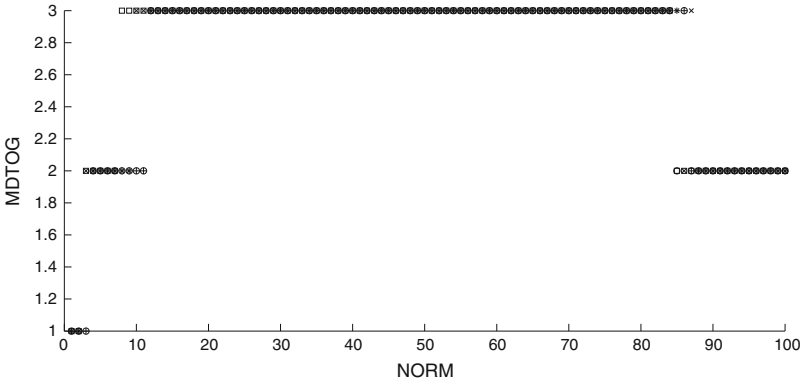


Fig. 6 Maximum number of different types of genes MDTOG as a function of number of replicated molecules NORM. Results obtained for mitosis coefficient M equal to 0 (◯), 0.3 (+), 0.6 (×) and 1 (□); $p_a = 0.1$, $p_p = 0.1$ and $p_l = 0.005$

It is worth to mention that, in order to validate the adopted mathematical approach, additional set of simulations was conducted for several scenarios and random values of DTOG, NORM, M triplets (from analysed ranges). Obtained results confirmed validity of applied methodology.

6 Discussion

Obtained results point out that in general, introduction of mitosis processes allows to obtain increase in the amount of information that can be stored by population of protocells. However, effects are somehow limited because even for the highest possible reliability of mitosis process ($M = 1$), and foster conditions (accidents and mutations disabled), increase in the number of different types of genes that can be stored by population of protocells is at the level between 1 DTOG, for NORM in the range between 4 and 6, up to 3 DTOG when NORM was greater than 94. Importantly increase in DTOG introduced by mitosis processes is also visible when accidents as well as mutations were operating individually at realistic levels (see Figs. 3, 4 and 5). Unfortunately activation of all deleterious processes in a single scenario blocked efficiently influence of even distribution of genetic material in descenting entities (Fig. 6).

Conclusions can be drawn that introduction of mitosis processes might lead to the increase of information storage abilities, however emergence of such a processes should be preceded by increase in fidelity of replication and inclusion of processes which can direct replication of RNA molecules. In other case benefits of possession of such a process would probably be mitigated by the complication in protocell structure which will lead to evolutionary disadvantage of such formations in relation to entities which do not realize such functionality. Therefore, in the next version of the model introduction of such RNA replication directing processes should be considered.

Acknowledgments The research leading to these results has received funding from BK/266/RAU2 /2014.

References

1. Benner, S.A., Kim, H.J., Yang, Z.: Setting the stage: the history, chemistry, and geobiology behind RNA. Cold Spring Harb. Perspect. Biol. **4**(1), a003541 (2010)
2. de Boer, F.K., Hogeweg, P.: Mutation rates and evolution of multiple coding in RNA-based protocells. J. Mol. Evol. **79**(5–6), 193–203 (2014)
3. Bresch, C., Niesert, U., Harnasch, D.: Hypercycles, parasites and packages. J. Theor. Biol. **85**(3), 399–405 (1980)
4. Cheng, L.K.L., Unrau, P.J.: Closing the circle: replicating RNA with RNA. Cold Spring Harb. Perspect. Biol. **2**(10), a002204 (2010)
5. Cyran, K.A.: Complexity threshold in RNA-world: computational modeling of criticality in galton-watson process. In: ACS 2008. pp. 290–295. Venice, Italy (2008)
6. Cyran, K.A., Kimmel, M.: Distribution of the coalescence time of a pair of alleles for stochastic population trajectories: comparison of fisher-wright and o'connell models. Am. J. Hum. Genet. **73**(5), 619–619 (2003)
7. Cyran, K.A., Kimmel, M.: Interactions of Neanderthals and modern humans: what can be inferred from mitochondrial DNA? Math. Biosci. Eng. **2**(3), 487–498 (2005)
8. Cyran, K.A., Kimmel, M.: Alternatives to the wright-fisher model: the robustness of mitochondrial eve dating. Theor. Popul. Biol. **78**(3), 165–172 (2010)

9. Eigen, M., Schuster, P.: The hypercycle: a principle of natural self-organisation, part A. Naturwissenschaften **64**(11), 541–565 (1977)
10. Higgs, P.G., Lehman, N.: The RNA world: molecular cooperation at the origins of life. Nat. Rev. Genet. **16**(1), 7–17 (2014)
11. Joyce, G., Orgel, L.: Progress toward understanding the origin of the RNA world. Cold Spring Harb. Monogr. Arch. **43**, 23–56 (2006)
12. Kimmel, M., Axelrod, D.E.: Branching Processes in Biology, Interdisciplinary Applied Mathematics, vol. 19. Springer, New York (2002)
13. Mozafari, M.R., Reed, C.J., Rostron, C.: Formation of the initial cell membranes under primordial earth conditions. Cell. Mol. Biol. Lett. **9**, 97–99 (2004)
14. Mulkidjanian, A.Y., Galperin, M.Y., Koonin, E.V.: Co-evolution of primordial membranes and membrane proteins. Trends Biochem. Sci. **34**(4), 206–215 (2009)
15. Myszor, D., Cyran, K.A.: Estimation of the number of primordial genes in a compartment model of RNA world. In: Cyran, K.A., Kozielski, S., Peters, J.F., Stańczyk, U., Wakulicz-Deja, A. (eds.) Man-Machine Interactions, pp. 151–161. Springer, Berlin (2009)
16. Myszor, D., Cyran, K.A.: Branching processes in the compartment model of RNA World. In: Czachórski, T., Kozielski, S., Stańczyk, U. (eds.) Man-Machine Interactions 2, AISC, vol. 103, pp. 153–160. Springer, Berlin (2011)
17. Myszor, D., Cyran, K.A.: Mathematical modelling of molecule evolution in protocells. Int. J. Appl. Math. Comput. Sci. **23**(1), 213–229 (2013)
18. Niesert, U., Harnasch, D., Bresch, C.: Origin of life between scylla and charybdis. J. Mol. Evol. **17**(6), 348–353 (1981)
19. Steitz, T.A., Moore, P.B.: RNA, the first macromolecular catalyst: the ribosome is a ribozyme. Trends Biochem. Sci. **28**(8), 411–418 (2003)

eVolutus: A Configurable Platform Designed for Ecological and Evolutionary Experiments Tested on Foraminifera

Paweł Topa, Maciej Komosinski, Maciej Bassara and Jarosław Tyszka

Abstract In this paper we present a new software platform called eVolutus, which is designed for modelling the ecological and evolutionary processes of living organisms. As a model organism we choose foraminifera—single-celled eukaryotes that mainly occupy marine benthic and pelagic zones. These organisms have lived on Earth for at least 500 million years and have an extraordinary fossil record. This makes them an ideal objects for testing general evolutionary hypotheses. We use a multiagent-based modelling platform called AgE. Our platform is designed to provide a highly configurable environment for conducting *in silico* experiments.

Keywords Ecology · Evolution · Multi-agent systems · Foraminifera

1 Introduction

Recent advances in evolutionary computation and genetic algorithms (GAs), although based on principles of genetics—including mutations, recombinations, and natural selection—are mostly motivated by the performance of these algorithms in the context of optimization. They are widely applied in modern science and technology, but their goal is not to simulate evolutionary processes within a realistic palaeoe-

P. Topa (✉) · M. Bassara
Department of Computer Science,
AGH University of Science and Technology, Krakow, Poland
e-mail: topa@agh.edu.pl

M. Bassara
e-mail: mbassara@gmail.com

M. Komosinski
Institute of Computing Science, Poznan University of Technology, Poznan, Poland
e-mail: maciej.komosinski@cs.put.poznan.pl

J. Tyszka · P. Topa
Research Centre in Cracow, Polish Academy of Sciences,
Institute of Geological Sciences, Warszawa, Poland
e-mail: ndtyszka@cyf-kr.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_23

cological and evolutionary "deep time" context. We propose to employ an *in-silico* artificial life approach [7] and reconstruct evolutionary patterns of foraminifera by implementing realistic principles into the computer environment inhabited by virtualized populations. Our aim is to construct a highly configurable modelling tool applied for running virtual ecological experiments and testing emerging dynamics of evolutionary patterns.

There are many computer simulations working either at the molecular level and/or the population genetic level (overview in [6]). Two categories of simulation algorithms exist, forward and backward, both suitable for addressing different questions. We aim to construct a forward-in-time simulator focused on individuals in their simulated populations. In this work we extend individuals (foraminifers in this case) into their iterative ontogenetic and morphogenetic growth stages, adequately controlled by their semi-genetic codes and interactively reacting to microhabitats. Random mutations of traits are introduced, shaping survival rates of individuals. This will lead to emergence of best-fit individuals that are continuously selected by the simulated environment.

As a model organism we use Foraminifera that belong to the class Globathalamea [14], single-celled eukaryotes that occupy marine benthic and pelagic zones throughout the world and have an extraordinary fossil record since Cambrian (500 Ma). This makes them an ideal microfossils often used for testing general evolutionary hypotheses [12, 15, 16].

We have previously introduced a new generation of morphogenetic models that can successfully predict the architecture of foraminiferal shells following the moving reference system [11, 17]. The reference system in this model is attached to the aperture (a hole) which provides communication between the surrounding environment and a foraminifera cell hidden inside the shell. We believe that this morphological component has crucial meaning for shell formation. The original model has now been extended by new components like size of the first chamber and thickness of the shell wall. They are introduced to simulate more realistic shell morphology, as well as to achieve proper behaviour in the microhabitat.

We assume that micropaleontologists and scientists from other related disciplines (i.e. palaeobiology) are skilled enough to define assumptions for similar experiments and analyse results produced by our model. Thus, we want to provide a modeling environment that can be used by a person which has limited skills in computer programming. This software will be freely available to the research community. In the following parts of this paper, the idea of the simulation model is presented along with its implementation and results.

1.1 Foraminifera

Foraminifera are single-celled marine eukaryotes that occupy benthic and pelagic habitats. Benthic foraminifera live either on the sea floor around the water/sedi-

ment interface, or within the top 10 cm of soft, usually fluidal, sediment. Planktonic foraminifera live mostly in the photic zone of the water column [2].

Foraminifera produce multichambered shells covering their soft cytoplasmic bodies. Shells are made from secreted $CaCO_3$ or from agglutinated grains of sand. Foraminifera build their shells through their whole life by adding successive chambers. In nature, we observe enormous variety of shell shapes and chambers, however for many species spheroidal chambers may be a close approximation. Communication between an internal part of a shell and the environment is provided by aperture. Foraminifera extend reticulopodia (network pseudopodia) through the aperture in order to gather food, move, and communicate [2, 13].

Foraminifera feed mostly on single-celled algae and their remains which makes them strongly dependent on the availability of algae in time and space. The most common temporal variability is reflected in seasonality associated with temperature and availability of nutrients. Variability in space is reflected in patchiness observed in various scales dependent on micro- and macro-scale environmental dynamics. All these factors have a direct impact on distribution, life history strategies, reproduction modes, and population dynamics of foraminifera [13].

Foraminifera are able to monitor their microenvironments around the cell thanks to the extension of large reticulopodial structures. We can assume that if food is available, benthic foraminifera are supposed to stay and feed and iteratively grow by adding chambers following certain portions of digested food. If there is a shortage of food, foraminifera can use at least two strategies: (1) wait for food and save energy or (2) move to another, better location. Planktonic foraminifera do not have any ability of active movement, so they are doomed to use the first strategy.

Foraminiferal life span ranges from a few weeks in some planktonic foraminifera up to a few years in larger benthic foraminifera [5]. A typical life cycle of benthic foraminifera is characterized by an alternation of two methods of reproduction: sexual (in haploid generation) and asexual (in diploid generation) [5]. A haploid generation is equipped with one set of chromosomes, while diploid generation has two sets of chromosomes. Planktonic foraminifera have only diploid generations and use a sexual method of reproduction. These complex life cycles help foraminifera in adjusting to variable (e.g. seasonal) conditions and allow them to create diverse and flexible life history strategies [13].

2 eVolutus Architecture

A general architecture of the eVolutus platform is presented in Fig. 1. The software presented in this paper is implemented in the Java language. For modeling habitat of foraminifera and their population dynamics, we use AgE, which is a framework for development and run-time execution of distributed agent-based simulations and computations, especially the ones utilizing the evolutionary paradigm [3, 4]. Agents equipped with genetic code and rules of growth and behaviour represent foraminifers placed in a virtual habitat. We plan the simulations to cover the area of up to hundreds

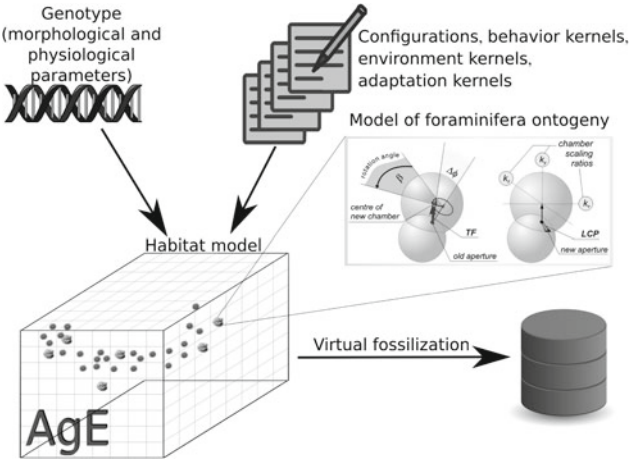


Fig. 1 The general architecture of the eVolutus simulation environment

or even thousands of square kilometres. The AgE framework which is equipped with tools for distributed computing can be helpful in this case.

The Java implementation described in this paper complements other existing foraminifera simulations, such as the one available in the Framsticks environment [8–10]. While Framsticks allows for high flexibility in defining experiments and setting up simulations thanks to a dedicated scripting language called *FramScript*, this scripting language is tailored for artificial life experiments. As such, it has less features than full-fledged programming languages like Java or Python.

In the experiments described in this work, the habitat is represented by a regular 3-dimensional grid of cells. Both types of foraminiferal habitats can be represented using this framework. Models of marine habitat cover the water volume up to 500 m deep. For modeling the benthic habitat, we need to only represent space that is 10 centimetres deep. Planktonic habitat is modelled with cell size of about 10–100 m. In case of the bentic habitat it will be a very thin block, i.e., 10 m × 10 m × 0.01 m.

In order to model the evolution of living organisms, thousands of generations have to be simulated. We assume that one step of simulation is mapped to 6–12 h of real time. These values were chosen upon observations of the time required by key physiological processes in foraminiferal ontogenesis, i.e., chamber forming and reproduction.

We assume that each block can be occupied by no more than thousands of agents—in order to save computational resources, we do not track their individual locations. Agents located in the same block use the same resources and experience the same environmental conditions. We also assume that during sexual reproduction, agents in the same block can exchange their genotypes with equal probability.

Agents in the model of planktonic habitat cannot move actively. They are moved using the random walk algorithm. The experimenter is able to define vectors of forces

which represent ocean currents. These forces will affect agents and move them. On the other hand, bentonic foraminifera have the ability to move actively. They slowly travel through their habitat gathering nutrients.

2.1 The Genotype and the Model of Foraminiferal Reproduction and Ontogenesis

The fundamental part of our platform is the existing model of foraminifera ontogenesis [10, 11, 17] shown in Fig. 2. The algorithm that generates morphology of foraminiferal shell is governed by several parameters listed in Table 1. They constitute a part of the genotype and cannot be modified or customized by experimenters.

In the model of foraminifera habitat, additional parameters that control physiology and behaviour have to be introduced (see Table 2). Some of them are included in the genotype and we assume that they will be more susceptible to customization.

The model of foraminifera reproduction is also built-in in our software and cannot be changed by users. At this moment, two reproduction methods are implemented: asexual and sexual. Planktonic foraminifers use only the sexual method, while bentonic foraminifera use both methods alternately in consecutive generations.

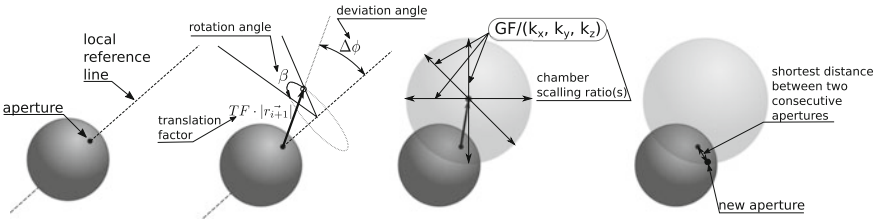


Fig. 2 The model of foraminifera ontogenesis [11]. The position of a new chamber is calculated with respect to the location of the previous aperture. The size of a new chamber is always greater or equal to the size of the previous chamber. The position of a new aperture is calculated by minimizing the distance between two consecutive apertures. The new aperture cannot be located in any of the previous chambers

Table 1 Parameters of the model of foraminifera morphology

Symbol	Name	Range
$GF\ (K_x, K_y, K_z)$	Scaling factors	$\geq$1.0
TF	Translation factor	0.0–1.0
$\Delta\phi$	Deflection angle	−180°–180°
$\Delta\beta$	Rotation angle	−180°–180°
R_1^*	Radius of first chamber	1–100 μm
W^*	Wall thickness factor	0.0–1.0

The asterisk (*) indicates that the parameter has been added in the current version of the model

Table 2 Parameters of the model of foraminifera physiology

Symbol	Name
D_{MAX}	Maximum amount of energy stored by an agent
D_{MIN}	Minimum amount of energy required for surviving
V_{Amin}	Minimum cytoplasm volume necessary for reproduction
J	The amount of cytoplasm for a gamete of new foraminifera
M	Metabolic efficiency

Asexual reproduction is implemented by generation of a new agents which inherit genotypes from the ancestor. These genotypes are modified by genetic operators. A more complex model is used to represent a sexual reproduction. In order to efficiently implement this complex and multistage process, we introduced the concept of reproduction events. These are procedures that generate new genotypes without laborious simulations of gamete creation, movement and coupling. The following steps are performed during the procedure of a reproduction event:

- An agent with the proper age and the sufficient level of energy spontaneously initiates reproduction and induces reproduction in other agents in the same box that fulfil the requirements (age, energy level).
- All agents generate tables with genotypes—they represent gametes that carry the genetic material.
- Randomly selected genotypes are removed reflecting the fact that most of the gametes are not coupled.
- Genotypes that remained are randomly coupled—the genetic recombination operator is applied.
- For each new genotype, a new agent is created.

Regardless of the type of reproduction, the ancestor agents are removed from the simulation and, if necessary, their genotypes and other parameters are saved for further analysis.

Although the general algorithms that handle reproduction (and especially reproduction events) cannot be easily modified by users, they can be adjusted and tuned by changing their parameters. Additionally, genetic operators can be defined by users.

2.2 Kernels

A high level of configurability and customization is provided by allowing users to define the behaviour of the environment and agents. These rules or procedures are often hidden in software code and cannot be easily modified. In eVolutus, we introduce the concept of "kernels" that roughly correspond to "events" in Framsticks [8, 10]. Kernels are relatively short functions that are created directly by the

experimenter. They contain information about environment configurations, procedures of its evolution, and agent behaviour rules. The name "kernel" used in CUDA or OpenCL GPU programming means a short function executed by many GPU processing units in parallel. Our kernels should also be short and will be executed many times for the population of simulated agents, so parallel execution would also be desired.

There are three types of kernels:

- **Environment kernels** are responsible for the evolution of the environment, i.e., changes in insolation and temperature.
- **Behaviour kernels** are responsible for the behaviour of individual agents.
- **Adaptation kernels** allows for calculating the level of foraminifera adaptation for local environmental conditions. The level of adaptation is used by behaviour kernels to calculate moments of growth, reproduction, death, and hibernation. By providing adaptation kernels, the experimenter defines specific conditions required by the evolutionary model.

Kernels have a strictly defined list of formal parameters and the returned value. Our goal is to define a new dedicated Domain Specific Language for programming eVolutus kernels. Its functionality should fit the experimenter needs and the eVolutus capabilities. Currently, Javascript is used as the programming language, as it can be easily invoked from the Java code. Three examples of kernel functions are provided below.

- The environment kernel for calculating the insolation level depending on spatial coordinates:

```
function insolation(x, y, z) {
    var surfaceInsolation = 1.0;
    var insolation = surfaceInsolation − 0.5 ∗ z;
    return Math.max(0.0, insolation);
}
```

  The function calculates the level of insolation at the location described by coordinates (x, y, z). The insolation at 0 meters depth is set to 1.0, and it decreases as depth increases.
- The adaptation kernel that calculates the probability of building a new chamber in the current step:

```
function growthProbability(x, y, z, age, energy) {
  return energy/Genome.DMAX;
}
```

  The probability is calculated depending on the current amount of energy stored by an agent and its maximum energetic capacity as defined by its genotype.
- The behaviour kernel checks whether the conditions required for initializing the reproduction event are fulfilled:

```
function startReproduction(x, y, z, age, energy) {
```

```
    if (age > 1000 && energy > Genome.VAMIN)
      return true;
    else
      return false;
}
```

The `age` is expressed in time-steps. In this example, the experimenter decided that after 1000 steps, agents are mature enough to reproduce.

2.3 Virtual Fossilization

The virtual fossilization module mimics the process of sedimentation of dead foraminifera shells. For millions of years, foraminifera shells are accumulated on the ocean floor in sedimentary rocks. Drilling and excavations uncover fossilized foraminifera and provide palaeoecological and evolutionary information. Virtual fossilization provides all the information that experimenter considers relevant, such as genetic and habitat information. It allows for tracking of genotype changes over consecutive generations under environmental pressure. Data stored in this module can be post-processed and classified using various analytic tools.

3 Demonstration of Platform Capabilities and Sample Results

Figure 3 presents a simulation snapshot. Parameters and rules for this simulation were adjusted to achieve the dynamics of the population that follows Lotka-Volterra model [1]. Two diagrams show changes in the number of foraminifers and the amount of food during simulation. In the upper diagram, the initial population of foraminifers was numerous, while in the lower diagram, the simulation started with a relatively small number of individuals. In both situations, the initial amount of nutrients was the same.

The results closely resemble the classical predator-prey models such as the Lotka-Volterra model. Regardless of the starting population of foraminifera, the system displays a similar behaviour—oscillations of foraminifera and algae populations shifted in phase. The amplitudes of these oscillations are gradually decreasing, and the average size of the population stabilizes around the carrying capacity which was similar in both simulations.

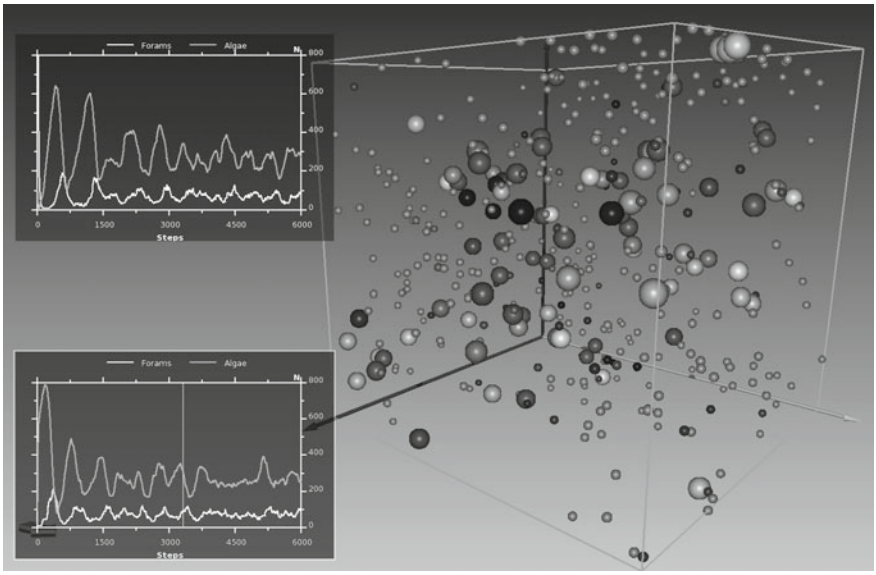


Fig. 3 A snapshot from the simulation: a population of virtual foraminifers (bigger spheres) and algae (small spheres) in an open marine environment visualized with the Amira software. Plots show changes in their populations in consecutive steps of simulation. *Upper plot* high level of initial foraminifers population, *lower plot* low level of initial foraminifers population

4 Conclusions and Further Work

The software platform presented here is under active development, and the initial tests of the model of habitat proved that the proposed approach is efficient. By using the concept of kernels, we are able to easily configure the environment so that it can simulate results similar to well-known theoretical models [1].

Our further work will focus on testing and validating algorithms for reproduction and exchanging genes. Another area of intensive works concerns large-scale simulations. Our goal is the ability to simulate the population size of 10^9–10^{12} individuals through a long period of time. The AgE framework has support for distributed computing, however for our purposes, additional mechanisms have to be implemented. This includes the need to map a regular grid of the habitat space into the available set of computational nodes, and providing methods of communication between nodes that process neighbouring blocks. It is desired that these mechanisms are implemented as an additional layer between AgE and the model of habitat. This way, the layer would be transparent to eVolutus developers and users. We anticipate that the distribution of agents will not be uniform, thus load-balancing mechanisms will also be necessary.

Acknowledgments The research presented in the paper received partial support from Polish National Science Center (DEC-2013/09/B/ST10/01734).

References

1. Berryman, A.: The origins and evolution of predator-prey theory. Ecology **73**(5), 1530–1535 (1992)
2. Brasier, M.: Microfossils. George Allen and Unwin, Cambridge (1980)
3. Byrski, A., Kisiel-Dorohinicki, M.: Agent-based model and computing environment facilitating the development of distributed computational intelligence systems. In: Allen, G., et al. (eds.) ICCS 2009. LNCS, vol. 5545, pp. 865–874. Springer, Berlin (2009)
4. Cetnarowicz, K., Kisiel-Dorohinicki, M., Nawarecki, E.: The application of evolution process in multi-agent world (MAW) to the prediction system. In: ICMAS 1996, pp. 26–32. Kyoto, Japan (1996)
5. Goldstein, S.: Foraminifera: a biological overview. In: Sen Gupta, B.K. (ed.) Modern Foraminifera, pp. 37–55. Springer, Netherlands (1999)
6. Hoban, S., Bertorelle, G., Gaggiotti, O.E.: Computer simulations: tools for population and evolutionary genetics. Nat. Rev. Genet. **13**(2), 110–122 (2012)
7. Komosinski, M., Adamatzky, A. (eds.): Artificial Life Models in Software, 2nd edn. Springer, London (2009)
8. Komosinski, M., Ulatowski, S.: Framsticks: creating and understanding complexity of life. In: Komosinski, M., Adamatzky, A. (eds.) Artificial Life Models in Software, chap. 5, pp. 107–148. Springer, London (2009)
9. Komosinski, M., Mensfelt, A., Topa, P., Tyszka, J.: Application of a morphological similarity measure to the analysis of shell morphogenesis in Foraminifera. In: Man-Machine Interactions 4. AISC, Springer (in press)
10. Komosinski, M., Mensfelt, A., Topa, P., Tyszka, J., Ulatowski, S.: Foraminifera: genetics, morphology, simulation, evolution (2014). http://www.framsticks.com/foraminifera
11. Labaj, P., Topa, P., Tyszka, J., Alda, W.: 2D and 3D numerical models of the growth of foraminiferal shells. In: Sloot, P.M.A., Abramson, D., Bogdanov, A.V., Dongarra, J.J., Zomaya, A.Y., Gorbachev, Y.E. (eds.) Computational Science–ICCS 2003. LNCS, vol. 2657, pp. 669–678. Springer, Berlin (2003)
12. Lazarus, D., Hilbrecht, H., Spencer-Cervato, C., Therstein, H.: Sympatric speciation and phyletic change in Globorotalia truncatuloides. Paleobiology **21**(1), 28–51 (1995)
13. Murray, J.: Ecology and Palaeoecology of Benthic Foraminifera. Longman Scientific and Technical, New York (1991)
14. Pawlowski, J., Holzmann, M., Tyszka, J.: New supraordinal classification of foraminifera: molecules meet morphology. Mar. Micropaleontol. **100**, 1–10 (2013)
15. Pearson, P., Shackleton, N., Hall, M.: Stable isotopic evidence for the sympatric divergence of Globigerinoides trilobus and Orbulina universa (planktonic foraminifera). J. Geol. Soc. **154**(2), 295–302 (1997)
16. Strotz, L.C., Allen, A.P.: Assessing the role of cladogenesis in macroevolution by integrating fossil and molecular evidence. Proc. Nat. Acad. Sci. **110**(8), 2904–2909 (2013)
17. Tyszka, J., Topa, P.: A new approach to modeling of foraminiferal shells. Paleobiology **31**(3), 526–541 (2005)

Part V
Biomedical Signal Processing

Real-Time Detection and Filtering of Eye Blink Related Artifacts for Brain-Computer Interface Applications

Bartosz Binias, Henryk Palus and Krzysztof Jaskot

Abstract Artifacts related with eye movements are the most significant source of noise in EEG signals. Although there are many methods of their filtering available, most of them are not suitable for real-time applications, such as Brain-Computer Interfaces. In addition, most of those methods require an additional recording of noise signal to be provided. Applying filtering to the recorded EEG signal may unintentionally distort its uncontaminated segments. To reduce that effect filtering should be applied only to those parts of signal that were marked as artifacts. In this paper it was proven that it is possible to detect and filter those artifacts in real-time, without the need of providing an additional recording of noise signal.

Keywords Real-time artifact detection · EEG · BCI · Ocular artifacts · Brain computer interfaces · Artifact detection · Artifact filtering

1 Introduction

Currents produced within dendrites during brain cells activation can be measured as the nonstationary electrical signal with the help of the device known as electroencephalograph [2]. Recording of such bioelectrical activity is called electroencephalogram (EEG). It was proven that certain characteristics of such signal can be changed by many kinds of mental activity [8]. Systems such as Brain-Computer Interfaces (BCI) are capable of detecting and interpreting those changes. As a result a communication channel between human mind and machine can be established [6]. Because EEG signals are characterized by very low amplitudes and bandwidth mostly located below 100 Hz, they are highly susceptible to contamination by other sources

B. Binias (✉) · H. Palus · K. Jaskot
Institute of Automatic Control, Silesian University of Technology, Gliwice, Poland
e-mail: bartosz.binias@polsl.pl

H. Palus
e-mail: henryk.palus@polsl.pl

K. Jaskot
e-mail: krzysztof.jaskot@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_24

of electrical activity which can occur during their recording. Among most problematic ones are artifacts related with eye movements, muscle noise and electrical line interferences [3]. In general, limiting of their influence proves to be a great difficulty both in clinical use of EEG and in BCI applications [5]. It must be noted that because ocular artifacts tend to mix with EEG rhythms and activity, they can make analysis of those signals not only less effective but, in many cases, impossible [3, 4]. Most commonly implemented approach for dealing with contaminated segments of signal is simply removing them from further analysis [4, 5]. However, such approach leads to a significant loss of data which, is unacceptable in e.g. real-time analysis of EEG signal for BCI applications. Over the years many approaches of filtering out ocular artifacts were proposed and successfully implemented. Most commonly a regression methods are performed in time or frequency domain [5]. Although highly effective, these methods posses a potential weakness for BCI applications which is a requirement of providing at least one regressing channel. Source of the best information about noise is an electrooculogram (EOG) recording, which can serve as reference for regression-based algorithms [5]. Applied in time domain, those methods can be implemented for real-time applications and are capable of producing a very good results in attenuation of ocular artifacts. Among other techniques of detecting and filtering eye movements and blinks are methods based on blind source separation. Those include Principal Component Analysis (PCA) which rely on recorded EEG and EOG signals for calibration [1]. Another approach is to apply the Independent Component Analysis (ICA) for the task of detection and correction of ocular artifacts in EEG [5]. This method, when applied to large number of data recorded over many channels, can produce some highly satisfying results. Additionally, in theory it is possible to implement this approach for online processing but because of its complexity it may not be an effective approach [5].

One very common assumption made for algorithms used for eye movement artifacts detection or filtering is that their causality is not required. That means the output of an algorithm in any given time does not need to depend only on the current and past samples. In addition, many of those methods are not suited for online data processing. Another presumption is that the noise reference is always available. Fulfilling aforementioned assumptions can be very problematic in BCI applications. These systems are created in order to provide a communication channel between human and machines, so that they can be controlled with mental activity [6]. Such action can be considered successful only if the control command can be delivered and executed in a time which is reasonably close to the moment of its transmission. An ideal situation would be to provide a real-time control for the user. In most cases classification of mental activities requires use of data recorded from many channels over different scalp locations. Number of electrodes used for experiment has a high impact on the comfort of BCI systems. Reducing that number should be taken into consideration by any researcher and BCI system designer. Such approach will significantly improve the comfort, practicability and portability of BCI.

In this article a special approach to the correction of ocular artifacts in EEG is proposed. The general idea is to apply filtering methods only to those segments of signal which are marked as contaminated. That way modification and corruption of

uncontaminated signal can be avoided. Similar approach was proposed in [7] but the requirement of causality was not fulfilled. In order to achieve said result a method of real-time eye blink and eye movement artifacts detection is presented. Special emphasis is placed on ensuring that proposed algorithms can be applied for real-time applications and that their performance is based only on recorded EEG signal, thus eliminating the need for additional EOG recording.

2 Data Description

Data used in this study was Dataset IIa provided for the BCI competition IV [9]. Database consists of recordings taken from 9 subjects. All available signals were recorded with Ag/AgCl electrodes from twenty-two different locations on scalp. Measurements were performed in monopolar setting with left mastoid serving as reference while right one was used as ground. Recordings were sampled with frequency of 250 Hz and were band pass filtered in range between 0.5 and 100 Hz. For removal of energy network interferences a notch filter at 50 Hz was enabled (for more details, see [9]). Additionally, three electrooculogram channels recordings were provided. These were referenced against left mastoid.

In this study there was used a 40 min long recording taken from subject 2. For a better evaluation of proposed method, signal from electrode position F_z of standard 10–20 system was taken. The reason for that is the fact that ocular artifacts are most present in frontal and prefrontal regions of a brain [9]. All 303 eye movement related artifacts which were present in chosen signal were manually marked and later used as reference for method's accuracy evaluation. Exemplary EOG time course with marked ocular artifacts is presented in Fig. 1. To compare efficiency of detection performed on EOG signal with one based on EEG recording, provided central EOG channel was used as a reference [6].

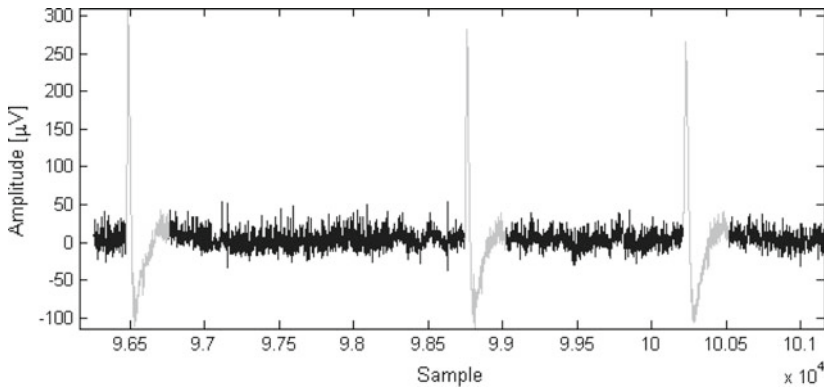


Fig. 1 Exemplary EOG time course with marked ocular artifacts

3 Artifact Detection Algorithm

Proposed in this paper method of eye movement related artifacts detection can be described as a thresholding with a varying cut-off level which adapts to changes in characteristics of signal. Conception that laid a basis for this approach is based on an assumption that there is no need for applying filtering to long intervals of EEG signal to eliminate their influence. Ocular artifacts occur randomly throughout the signal for short segments of time. Accurate detection of those time moments would allow to apply filtering only to contaminated parts of signal. That way, loss of important information can be avoided, because there would be no modification of clear recording. Such detection can be qualified as real-time, only if its output is causal, or delayed by a negligible amount of time. Rapid nature of amplitude changes, that accompany EOG artifacts, makes it hard to perfectly detect their beginning without any information about future values of the signal.

This method allows the possibility of setting a small delay δ for the output y_{det}, so it is needed to substitute time index n with $m = n + \delta$. For a given observation of y at time moment n, proposed detection algorithm can be described with following steps:

1. Calculate a squared value of each sample of recorded y signal: $y_{pow}(m) = y^2(m)$
2. Calculate value of detection function as mean of m_1 previous values of $y_{pow}(m)$: $d_1(m) = \frac{1}{m_1+1} \sum_{i=m-m_1}^{m} y_{pow}(i)$.
3. If previous sample was marked as an artifact $y_{det}(m-1) = 1$ do not change the value of threshold: $\theta(m) = \theta(m-1)$ and go to point 5.
4. If local maximum was detected change the cut-off level $\theta(m)$ to its value. Otherwise do not change the threshold: $\theta(m) = \theta(m-1)$
5. If $d_1(m) > B\theta(m)$ mark mth time index as an artifact $y_{det}(m) = 1$. Otherwise $y_{det}(m) = 0$. B is a input parameter of described method.
6. Output $y_{det}(m) = y_{det}(n+\delta)$ applies for nth time index of signal y.

Detection based on the square of analyzed signal, makes proposed algorithm insusceptible to polarization of eye movement artifacts. Threshold level used in described algorithm is being adjusted on the basis of value of local maxima that occur in smoothed, squared signal. Any method that provides a real-time result can be applied here. In this paper for each sample m it was tested whether the following condition was satisfied:

$$d_1(m) < d_1(m-1) \bigwedge d_1(m-2) < d_1(m-1) \tag{1}$$

For evaluation of described method results of detection based on EOG and EEG signal were compared with manually marked artifacts. Quality of proposed artifact detection algorithm was evaluated on the basis of two coefficients Δ_1 and Δ_2 (formulas (2) and (3)).

$$\Delta_1 = n_{start}^{real} - n_{start}^{detected} \tag{2}$$

Table 1 Evaluation of artifact detection algorithm

Detection signal	Undetected artifacts (%)	False positives	Δ_1 (ms)	Δ_2 (ms)
EOG	0	1	-3.75 ± 3.3	251.89 ± 9.16
EEG	14.19	0	-86.72 ± 189.68	-485.28 ± 532.56
Filtered EEG	2.31	7	-113.2 ± 22.72	-230.98 ± 57.33
Filtered EEG ($\delta = 210$ ms)	0.99	13	57.36 ± 55.2	160.04 ± 122

$$\Delta_2 = n_{end}^{detected} - n_{end}^{real} \tag{3}$$

In this paper it was assumed that proposed algorithm should be able to detect artifact's beginning $n_{start}^{detected}$ and ending $n_{end}^{detected}$ time moments with some margin. Although a precise detection, where $\Delta_1 = \Delta_2 = 0$ may seem as the best result, in real cases there is always some margin of detection added [4, 5]. Additionally, in order to avoid situation where an interference would not be fully detected, thus resulting in high amplitude artifact around points n_{start}^{real} or n_{end}^{real}, it was assumed that both Δ_1 and Δ_2 should be greater than 0. For better evaluation of method's accuracy a number of undetected artifacts, as well as number algorithm's False Positive detections were taken into consideration. Results of tests performed on both EOG and EEG signal can be found in Table 1. For testing the following parameters were selected:

- Length of buffer for previous samples $m_1 = 1000$ ms
- Delay introduced by method $\delta = 37$ ms
- Input parameter of method $B = 3$.

Analysis of achieved results shows that detection of ocular artifacts in EOG signal is very accurate. All 303 artifacts were correctly marked with only one false detection. Additionally proposed method provided a safe margin of detection. On contrary, use of EEG signal produced a significantly less accurate results. With over 14 % of undetected trials it cannot be applied for any real life solutions. The reason for such weak performance is the fact, that in EEG recordings a signal-to-noise (SNR) ratio is much higher than in EOG recording. Ocular artifacts, which are here considered to be a noise, are much more mixed into EEG signal, making them very difficult to separate. One solution to that problem would be to artificially decrease SNR of EEG signal. A time-frequency analysis of EEG segment (Fig. 2) shows that ocular artifacts are very strong in frequency band below 10 Hz. Applying a lowpass filtering with cut-off frequency at that level should significantly increase the difference between artifacts and clear EEG. For filtering of EEG signal a simple elliptic Infinite Response Filter (IIR) was used. Because of nonlinear phase characteristic such filters are not common in biomedical measurements. However, because purpose of lowpass filtering in described method is only to ensure a better separability between clear and contaminated time segments, such choice of filter type is acceptable. Amplitude and

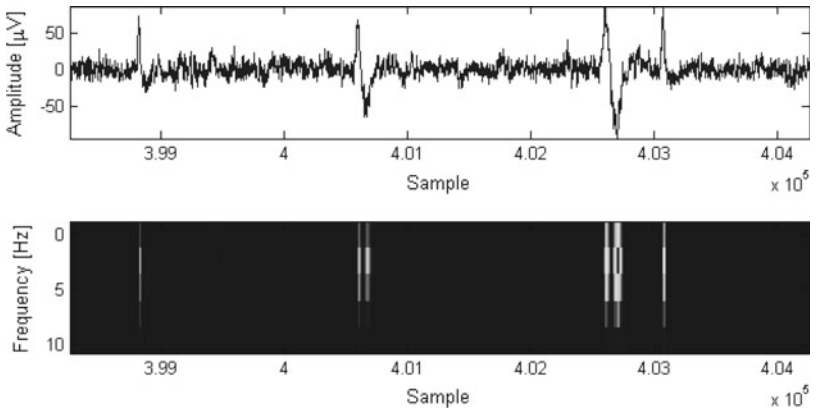


Fig. 2 Time-frequency analysis of an EEG segment contaminated with ocular artifacts

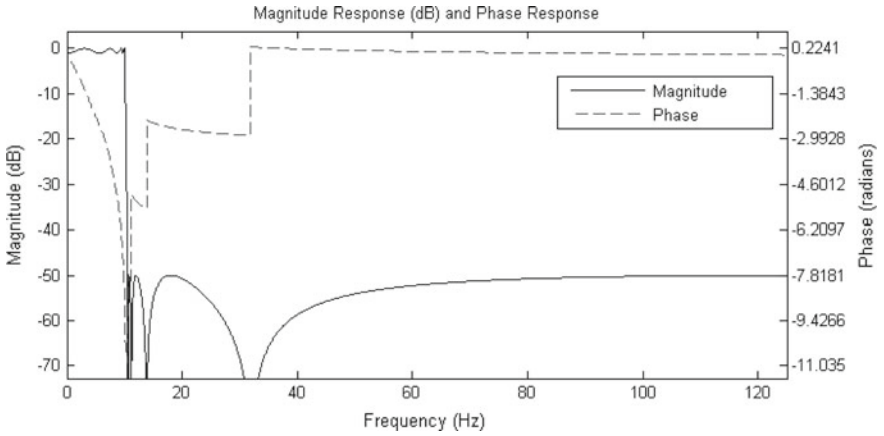


Fig. 3 Amplitude and phase characteristics of used IIR lowpass filter

phase characteristics of used IIR lowpass filter are presented in Fig. 3. Exemplary time courses of EEG signal before and after increasing of ocular artifact's separability are shown in Fig. 4.

Even the visual analysis of Fig. 4 confirms that proposed approach led to an improvement of artifact's visibility. Results of detection performed with mentioned parameters are presented in Table 1. It should be noted that the number of undetected artifacts was significantly lowered to 2.31 % with only 7 False Positives and improved detection margin. Performed tests used settings that provided best possible results while introducing the lowest possible delay to the output of algorithm. However it should be noted, that for longer buffer and delay the method is capable of almost perfect detection of ocular artifacts in EEG signal. Exemplary parameters that allow such performance are presented below:

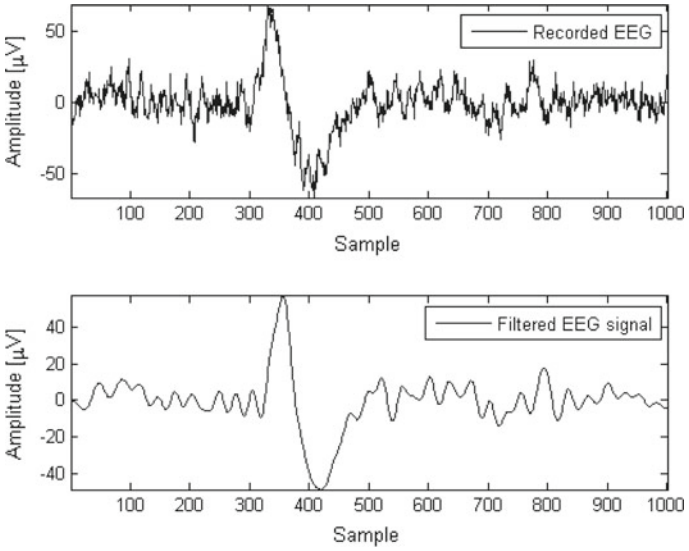


Fig. 4 Result of lowpass filtering

- Length of buffer for previous samples $m_1 = 1500$ ms
- Delay introduced by method $\delta = 210$ ms
- Input parameter of method $B = 2.2$.

Results of detection performed with mentioned parameters presented in Table 1 show a very accurate detection witch only about 1 % of undetected artifacts and margins Δ_1 and Δ_2 that are positive for every detection.

4 Artifact Filtering

In this paper a method of filtering eye movement related artifacts from EEG signal was proposed. Its performance was compared to classical approach based on adaptive filtering with EOG signal used as noise reference. A structure of an adaptive filter based on a Recursive Least Squares (RLS) algorithm is shown in Fig. 5. This algorithm requires a primary signal $y(n)$ and secondary signal $z(n)$ to be provided. In this research $y(n)$ was a contaminated EEG recording and $z(n)$ (reference signal) carried the information about unwanted noise. For typical EEG filtering applications an additional EOG channel is used for that purpose. The goal of an adaptive filter is to change coefficients of linear filter so that the reference $z(n)$ will be transformed to resemble primary signal. It is assumed that the corrupted signal $y(n) = s(n) + z_0(n)$ can be described with two components: desired $s(n)$ and uncorrelated with it noise $z_0(n)$. Because of said uncorrelation linear filter's

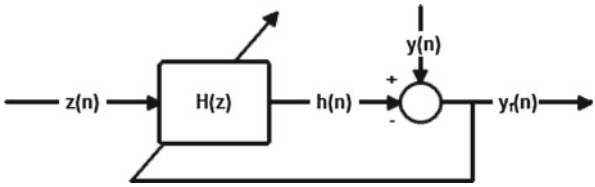


Fig. 5 Block diagram of filtration method based on adaptive filtering

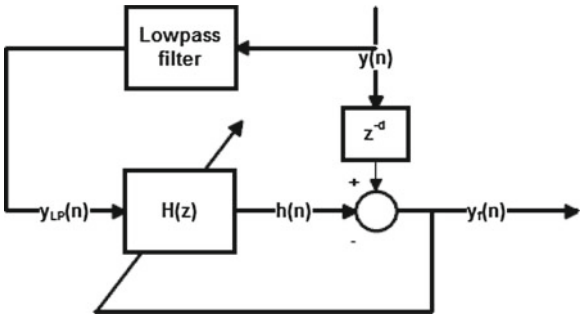


Fig. 6 Block diagram of filtration method based on adaptive filtering with filtered signal as reference

output $h(n)$ will adapt to the noise $z_0(n)$ present in $y(n)$, so that $h(n) \simeq z_0(n)$. Subtracting $h(n)$ from $y(n)$ allows to achieve the estimate of uncorrupted signal $y_f(n) = y(n) - h(n) = [s(n) + z_0(n] - h(n) \simeq s(n)$.

Proposed method of ocular artifact filtration is based on adaptive filtering algorithm described above. However, a crucial innovation lays behind a fact that an external EOG reference signal is replaced by lowpass filtered EEG. In Fig. 6 there is a general structure of proposed algorithm. It was shown in this paper that information about ocular artifacts is strongest in bandwidth below 10 Hz. Because in proposed approach artifact filtering is applied only to time segments of signal marked as contaminated, it can be assumed that reference $y_{LP}(n)$ is not correlated with desired component $s(n)$ of recorded $y(n)$. It must be noted that use of lowpass filter will introduce some latency to the signal. To ensure a reliable filtering signal $y(n)$ should thereby be delayed for d samples. This delay can be, however, covered in δ parameter of detection algorithm described in this paper.

Performance of proposed methods was tested on modified data used for detector's evaluation. Firstly, all manually marked artifacts were removed from EEG and EOG signals. That way a two 30 min long time series of uncontaminated signals were acquired, as well as, 303 single ocular artifacts. Then, for each artifact a randomly selected 20 s long segment of clear EEG and EOG was selected. For EOG channel, artifact samples replaced original signal. Artifacts inserted to EEG were first downscaled 3.3 times and then added to the original data. Effectiveness of described filtering algorithms was evaluated using Mean Squared Error (MSE) and Pearson's Correlation (ρ) coefficients. Because idea behind proposed approach was to apply filtering only to parts of signal marked as artifacts, both MSE and ρ were calculated only for those time moments. Comparison of proposed algorithm with classical approach is presented in Table 2.

Table 2 Evaluation of artifact filtering algorithms

Reference signal	ρ	MSE
EOG	0.93 ± 0.06	11.67 ± 11.29
Filtered EEG	0.65 ± 0.09	52.4 ± 21.26

Analysis of achieved results shows that both methods are capable of reconstructing EEG signal satisfyingly. It should be noted that use of EOG recording as reference signal allowed almost perfect artifact correction. However, although noticeably weaker in performance, proposed approach with filtered EEG signal used as reference produced a satisfactory results while, in the same time, eliminating the need for providing an additional EOG recording.

5 Conclusions

Method of detecting artifacts proposed in this research allowed to achieve a highly satisfying results. It was shown that the detector is capable of detecting ocular artifacts both in EOG and EEG signal. Achieved results are characterized by a small number of undetected artifacts and False Positives. Additionally, for EOG and filtered EEG signals, algorithm ensured a safe margin of detection. That way avoided is situation were artifact is not fully detected, resulting in some high-amplitude peaks being still present in signal. Proposed approach of filtering eye movement related artifacts and reconstructing EEG signal performed satisfactorily during evaluation. It must be noted that in general MSE and Pearson's Correlation scores of this method were lower than for approach with EOG channel used as reference. However, based on the experience of this paper's authors, use of as few as even one additional electrode required for EOG measurements may highly decrease the comfort of the BCI system usage. Taking that into consideration, steps of eliminating that problem should be taken by all BCI system designers. Solutions proposed in this paper allowed to deal with that inconvenience while, at the same time, maintaining a very satisfactory accuracy of artifact detection. Although methods described in this research focus on a single channel applications, it is possible to apply them to a multichannel EEG recordings. However, as the influence of eye movement related signals tend to decrease with the increase of the distance from their source, it is suggested to perform a detection on a recordings acquired from locations closer to a frontal and prefrontal regions of brain such as F_z, F_{p1} or F_{p2} of standard 10–20 system. Because those signals contain more information about noise than others, such approach will ensure a more accurate performance of proposed method. Results of such detection will apply to all EEG signals, recorded from different scalp locations. In order to provide a more efficient detection of ocular artifacts beginning, proposed method needed to introduce a delay to signal's output. Importantly, as that delay does not exceed 37 ms

(apart from one proposed setup) it can be considered as negligible. Considering that most of the commonly used detection algorithms rely on an offline processing, this feature is a huge improvement and holds a significant advantage for real time BCI applications.

Acknowledgments This work was supported by Polish Ministry for Science and Higher Education under internal grant BK-227/RAu1/2015/t.4 for Institute of Automatic Control, Silesian University of Technology, Gliwice, Poland.

References

1. Berg, P., Scherg, M.: Dipole models of eye activity and its application to the removal of eye artifacts from the EEG ad MEG. Clin. Phys. Physiol. Meas. **12**(Suppl A), 49–54 (1991)
2. Chambayil, B., Singla, R., Jha, R.: EEG eye blink classification using neural network. In: WCE 2010, vol. I, pp. 63–66. London, UK (2010)
3. Correa, A.G., Laciar, E., Patiño, H.D., Valentinuzzi, M.E.: Artifact removal from EEG signals using adaptive filters in cascade. J. Phy. Conf. Ser. **90**, 1–10 (2007)
4. Croft, R.J., Barry, R.J.: Removal of ocular artifact from the EEG: a review. Clin. Neurophysiol. **30**(1), 5–19 (2000)
5. Jung, T. P., Makeig, S., Humphries, C., Lee, T.W., McKeown, M.J., Iragui, V., Sejnowski, T.J.: Removing electroencephalographic artifacts by blind source separation. Psychophysiology **37**(2), 163–178 (2000)
6. Leeb, R., Lee, F., Keinrath, C., Scherer, R., Bischof, H., Pfurtscheller, G.: Brain-computer communication: motivation, aim and impact of exploring a virtual apartment. IEEE Trans. Neural Syst. Rehabil. Eng. **15**(4), 473–482 (2007)
7. Melia, U., Clariá, F., Vallverdú, M., Caminal, P.: Filtering and thresholding the analytic signal envelope in order to improve peak and spike noise reduction in EEG signals. Med. Eng. Phys. **36**(4), 547–553 (2014)
8. Pfurtscheller, G., Aranibar, A.: Evaluation of event-related desynchronization (ERD) preceding and following voluntary self-paced movement. Electroencephalogr. Clin. Neurophysiol. **46**(2), 138–146 (1979)
9. Tangermann, M., Müller, K.R., Aertsen, A., et al.: Review of the BCI competition IV. Front. Neurosci. **6**(55), 1–31 (2012)

Application of Dimensionality Reduction Methods for Eye Movement Data Classification

Aleksandra Gruca, Katarzyna Harezlak and Pawel Kasprowski

Abstract In this paper we apply two data dimensionality reduction methods to eye movement dataset and analyse how the feature reduction method improves classification accuracy. Due to the specificity of the recording process, eye movement datasets are characterized by both big size and high-dimensionality that make them difficult to analyse and classify using standard classification approaches. Here, we analyse eye movement data from BioEye 2015 competition and to deal with the problem of high dimensionality we apply SVM combined with PCA feature extraction and random forests wrapper variable selection. Our results show that the reduction of the number of variables improves classification results. We also show that some of classes (participants) can be classified (recognised) with high accuracy while others are very difficult to be correctly identified.

Keywords Eye movement data analysis · DTW · Dimensionality reduction · Classification · PCA · SVM · Random forest

1 Introduction

Recent advances in computer science technology, development of new technologies and data processing algorithms provided new tools and methods that are used to control access to numerous resources. Some of them are widely available while others should be protected against an unauthorized access. For the latter case various security methods have been developed like PINs, passwords, tokens, however biometrics solutions such as gait, voice, mouse stroke or eye movement are becoming more and more popular due to their convenience.

A. Gruca (✉) · K. Harezlak · P. Kasprowski
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: aleksandra.gruca@polsl.pl

K. Harezlak
e-mail: katarzyna.harezlak@polsl.pl

P. Kasprowski
e-mail: pawel.kasprowski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_25

In the field of eye movement research the biometric identification plays an important role as the information included in such signal is difficult to imitate or forge. Acquisition of such data is realized with the usage of various types of cameras and can provide, depending on recording frequency of an eye tracker used, from 25 to 2000 samples per second. A total amount of obtained samples depends on a time of registration—it can be only during a login phase or continuously during the whole session. In the latter case, obtained dataset is characterized by both big size and high dimensionality of features describing it. Analysis of such big dataset is therefore a challenging task.

It may seem that the more information we include into our analysis, the better decision we can make, however it is possible to reach a point beyond which data analysis is very difficult and sometimes impossible. In numerous cases objects in dataset are characterized by many features, which are redundant or have no impact on a final result. Taking them into consideration may not improve quality of results but even make it worse. Additionally, two problems—a complexity of a classifier used and so-called the curse of dimensionality—may arise. In the latter case we need to deal with exponential growth of data size to ensure the same quality of classification process when a dimensionality grows up. Moreover, collecting such amount of data may be expensive and difficult to perform. To solve this problem, additional methods are necessary that allow finding relationships existing in data, filter redundant information and select only these features which are relevant to a studied area, and classification outcome. Data dimensionality reduction methods has been successfully applied in machine learning in many different fields such as industrial data analysis [3], computer vision [10], geospatial [17] or biomedical data analysis [20]. Removing unimportant features has the following advantages:

- reduces bias unimportant from a classifier point of view,
- simplifies calculation saving resources usage,
- improves learning performance,
- reveals real relationships among data,
- improves classifier accuracy.

In the research described in the paper the problem of dimensionality reduction has been applied into analysis of eye movement data for the purpose of biometric classification [19]. Two methods have been considered: PCA [6] method combined with SVM classifier and random-forest based procedure [5]. Data from eye movement sessions were transformed into the form suitable to be analysed by a classifier using Dynamic Time Wrapping (DTW) distance measure method. The main contribution of this paper is application of DTW metrics to eye movement data analysis and performance comparison of two different data reduction dimensionality methods: feature selection and feature extraction on classification results.

The paper is organized as follows: two first sections provide the description of data used in the research and its pre-processing phase. Next section includes description of two dimensionality reduction methods. Then, results of analysis and final conclusions are presented.

2 Description of Data

Data used in the presented research is a part of the dataset available for the BioEye competition (www.bioeye.info). The purpose of the competition was to establish methods enabling human identification using eye movement modality. Eye movement is known to reveal a lot of interesting information about a human being and eye movement based identification is yet another biometric possibility which was initially proposed about 10 years ago [13]. Since then, many research have been done in that field and the BioEye competition follows the previously announced EMVIC2012 [12] and EMVIC2014 [11] competitions.

The dataset used in this research consisted of eye movement recordings of 37 participants. During each session the participant had to follow with eyes a point displayed on a screen. As the point changed its position by leaps and bounds, eye movement data consisted of fixations on stable point and sudden saccades to the subsequent location. The point position changed every 1 s and there were 100 random point locations during each session so the whole session lasted 1 min and 40 s. Eye movement data was initially recorded with frequency 1000 Hz and then downsampled to 250 Hz with the usage of noise removal filter. Finally, there were 25,000 recordings available for every session. Each recording was additionally described by a "Validity" flag. Validity equal to 0 meant that the eye tracker lost eye position and data recorded is not valid.

There were two sessions available for every participant referred later as the first session and the second session. The task was to build a classification model using data from the first session as training samples and then use it to classify the second session for every subject.

3 Data Preprocessing

There were 37 first (training) sessions and 37 second (testing) sessions available. Initially, every session was divided into segments when displayed point location was stable. It gave 100 segments for each session. The first segment of each session was removed. Every segment consisted of 250 recordings but some of these recordings were invalid (with validity flag set to 0). Segments with less than 200 valid recordings were removed from the set. It resulted in 6885 segments. Every segment consisted of 200–250 eye movement recordings. 3425 segments were extracted from the first sessions and were used as training samples and 3460 segments from second sessions were used as test samples. The segments were divided into four groups: NW, NE, SE and SW based on the direction of points location change. Finally, there were 823 training segments in NE direction, 869 in SW direction, 925 in SE and 808 in NW accordingly.

In the next step pairwise distances among all training segments were calculated. As the length of the segments was varying and we were interested more in

shape comparison than point-to-point comparison, we used Dynamic Time Warping to calculate distances among samples [4]. The distance calculation was done for each of the nine different signal features: velocity, acceleration and jerk in vertical direction, velocity, acceleration and jerk in horizontal direction and absolute velocity, acceleration and jerk values. The distances were calculated separately for every group (NW, NE, SE and SW). The results of that calculations were $9 \times 4 = 36$ matrices containing distances among training samples.

These distances were treated as features in a way similar to [18]. For every test sample there were DTW distances of this sample to every training sample of the same direction calculated and these distances were treated as features describing this sample.

Finally, the full dataset consisted of 36 sets. Nine NE sets had 1689 samples (including 823 training samples) with 823 features, nine SW sets had 1769 samples (incl. 869 training) with 869 features, nine NW sets had 1597 samples (incl. 808 training) with 808 features, and finally nine SE sets consisted of 1830 samples (incl. 925 training) with 925 features.

4 Dimensionality Reduction Methods

Methods for decreasing dimensionality may be divided into two main groups—feature extraction or feature selection. In the first group of methods original features are transformed to obtain their linear or non-linear combination. As a result data are represented in another feature space. The second technique relies on such choice of features that discriminate analysed data best. The task of a feature selection is to reduce redundancy while maximizing quality of the final classification outcome. The extension of a feature selection method is wrapper variable selection where during feature selection process the learning algorithm and the training set interact.

In this section we present application of two data dimensionality reduction methods: PCA for feature extraction and random forest for wrapper variable selection into BioEye2015 dataset.

4.1 Feature Extraction with Principal Component Analysis

One of the methods utilized in the research was PCA (Principal Component Analysis) which is an example of the feature extraction method. It was successfully used in many classification problems (pattern recognition, bioinformatics), in these, in field of eye movement data processing as well [2, 19].

The PCA task is to reveal a covariance structure in data dimensions to find differences and similarities between them. As a result transformation of correlated variables into uncorrelated is possible. These uncorrelated variables are called principal components. They are constructed in a way ensuring that the first of components

accounts for the most possible variability in the data. The same regards each succeeding component, which explains as much of the remaining variability as possible.

In the presented research the feature extraction was done with usage of *prcomp()* function available in R language from the default stats package. As a function input, a matrix representing DTW distances calculated based on one of previously-described features was provided. Data from this matrix was limited only to the first sessions of recordings. Center and Scale parameters of *prcomp()* function have been used to (1) shift the data to be zero centered and (2) scale it to have unit variance. Data transformed this way served as a training set for SVM classifier [1, 24], which has been successfully used in the field of machine learning and pattern recognition. SVM performs classification tasks by constructing hyperplanes in a multidimensional space that separates objects of different class labels. It uses a set of mathematical functions called kernels to map original data from one feature space to another one. The method is very popular because it solves a variety of problems and was proved to provide a good classification accuracy even for a relatively small data set. For this reason it seems to be suitable for an analysis of an eye movement signal, which is often gathered during short sessions.

There are different types of kernel mappings such as the polynomial kernel and the Radial Basis Function (RBF) kernel. The latter one was applied in the presented research with usage of *svm()* R function from e1071 package and $C = 2^{15}$ and gamma $= 2^{9}$ settings. The classification model was verified using of *predict()* function. Its input parameter was a test set constructed on the basis of PCA model. It was applied to this part of samples in a form of the distance matrix, which was obtained from the second recording session. Because prediction probabilities were evaluated for each sample in the distance matrix, they were subsequently summed up and normalized in regard to samples related to one user. As a result one probability vector for each user was provided for one distance matrix. This procedure has been repeated for all 36 distance matrices thus 36 user probability vectors were achieved, which were finally averaged for the second time.

4.2 Wrapper Variable Selection with Random Forest

Another data dimensionality reduction method used was random-forest based procedure for wrapper variable selection [14]. Unlike feature extraction, feature selection methods allow improving classifier accuracy by selecting the most important attributes. Therefore, resulting subset of attributes may be further used not only for classification purposes but also for data description and interpretation [15, 21].

Wrapper variable selection approach can be used on any machine learning algorithm, however we decided to choose random forest due to the fact that this method is particularly suitable for the high-dimensionality problems and it is known to be hard to over-train, relatively robust to outliers and noise, and fast to train [23]. Wrapper variable selection method is based on the idea of measure of importance, which ranks variables from the most to the least important. Then, in several iterations, less

important variables are removed, the random forest is trained on remaining set of values and its performance is analysed.

Random forest method [5] is based on ensemble learning idea and it combines number of decision trees in such a way that each tree is learned (grown) based on a bootstrap sample drawn from the original data. Therefore, during the learning process the ensemble (forest) of decision trees is generated. Final classification result is obtained based on a simple voting strategy. Typically one-third of the cases are left out of the bootstrap sample and not used to generate the tree. The objects that are left are later used to estimate so-called out-of-bag (OOB) error.

Additional feature of random forest method is a possibility of obtaining a measure of importance of the predictor variables. In the literature one can find various methods to compute importance measures and these methods typically differs in two ways: how the error is estimated and how the importance of variables is updated during learning process [8]. Here, we focus on so-called permutation importance that is estimated in such a way that for a particular variable its values are permuted in OOB cases and then it is checked how much prediction error increased. The more error increases, the more important is the value or, in other words, if the variable is not important, then rearranging the values of that variable will not decrease prediction accuracy. The final importance value for an attribute is computed as an average over all trees.

There are two backward strategies that can be applied when using importance ranking. First one is called Non Recursive Feature Elimination (NRFE) [7, 22] and in this approach the variable ranking is computed only once at the beginning of the learning process. Next, less important variables are removed from the ranking and the random forest is learned based on the remaining set of values. This step is repeated in several iterations until no further variables remain. Second approach is called Recursive Feature Elimination (RFE) [9] and it differs from NRFE method is such a way that the importance ranking is updated (recomputed) at each iteration. Then, similarly to NRFE, the less important variables are removed and random forest is learned. In the work of Gregorutti et al. [8] extensive simulation study was performed comparing these two approaches. Based on their analysis we decided to choose RFE approach as it might be more reliable than NRFE since the ranking by the permutation importance measure is likely to change at each step and by recomputing the permutation importance measure we assure the ranking to be consistent with the current forest [8].

The final procedure used to learn random forest classifier was as follows:

1. Train the random forest.
2. Compute permutation measure of importance for each attribute.
3. Remove half of the less relevant variables.
4. Repeat steps 1–3 until there is less than 10 variables in the remaining attribute set.

As in the case of PCA analysis, the learning procedure has been repeated for all 36 distance matrices which resulted in obtaining 36 random forests.

5 Results

5.1 Combined SVM and PCA Results

To obtain the best possible prediction result, several cases concerning various cumulative proportion of explained variance—95, 97, 99 and 99.9 %—have been analysed. The most interesting issue on the first step of analysis was to check how dimensionality reduction influenced accuracy of data classification and what degree of reduction could provide the best possible data classification.

Results were compared to the classification based on the all dimensions used. Please note that for one user recording there were 36 sets of samples. Each set included DTW distance matrix calculated for all users taking one signal feature into consideration. The number of dimensions related to each set, dependent on eye movement direction, varied from 808 to 925 elements. The performance of the classification was assessed using two quality indices:

- Accuracy—the ratio of the number of correctly assigned attempts to the number of all genuine identification attempts.
- FAR—the ratio calculated by dividing the number of false acceptances by the number of identification attempts.

First, we classified the whole dataset using SVM method and the classification accuracy obtained with the usage of all dimensions was 24 %, and the FAR ratio was 4 %. Then we applied PCA method; the Table 1 presents classification results for all levels of explained variability considered in the research. They are complemented by the information about the number of principal components required to account for a given variability. Due to the fact, that this number was calculated independently for each set of samples (36 times), the final result is presented in a form of average, minimal and maximal number of components utilized.

These outcomes clearly indicate that applying PCA dimensionality reduction method has had significant influence on classification accuracy. It is visible for both explained variance percentages 99 and 99.9 %. While in the former case the accuracy is comparable with the full dimensionality calculations, in the latter one it was improved more than twice. It is worth emphasizing that both results were obtained for

Table 1 Classification results for various levels of explained variability

Proportion of explained variance (%)	Accuracy (%)	FAR (%)	Average number of components	Maximal number of components	Minimal number of components
95	11	5	1.9	5	1
97	11	5	3.08	8	1
99	22	4	8.81	18	1
99.9	54	2	91.81	203	19

remarkable smaller number of dimensions (on average 8.81 and 91.81 components respectively comparing to about 900 features in the primary sets).

Analysing the classification results we noticed that there were some samples that obtained very similar probability for two or more analysed classes. To deal with this similarity results appropriate *acceptance threshold* was defined and another step to data analysis was introduced.

If we denote by:

- p_i a probability that sample s belongs to class i,
- p_j a maximal probability obtained for sample s indicating that this sample belongs to class j,
- and ($j \neq i$ and $i, j \in 1 \ldots 37$),

and if:

$$p_i - p_j \leq acceptance\_threshold,$$

then the probability of sample s belonging to both classes i and j is treated as equally likely.

Four values of the threshold defined as the difference between calculated probabilities 0.000, 0.001, 0.0025 and 0.005 were studied. As it was expected, the bigger threshold value the higher accuracy was obtained. However, increasing threshold values resulted in increasing of FAR ratio as well (Table 2 and Fig. 1). The last column of the table presents the ratio of accuracy and FAR for a particular threshold. It can be noticed that for the two first proportions of explained variability the best ratio was obtained for threshold equal 0.001, while in the third case both thresholds 0.000 and 0.001 provided similar ratio values. In the last of the variabilities, proportion threshold 0.000 significantly surpassed the others. The ratio of accuracy and FAR for all parameters was presented in Fig. 2.

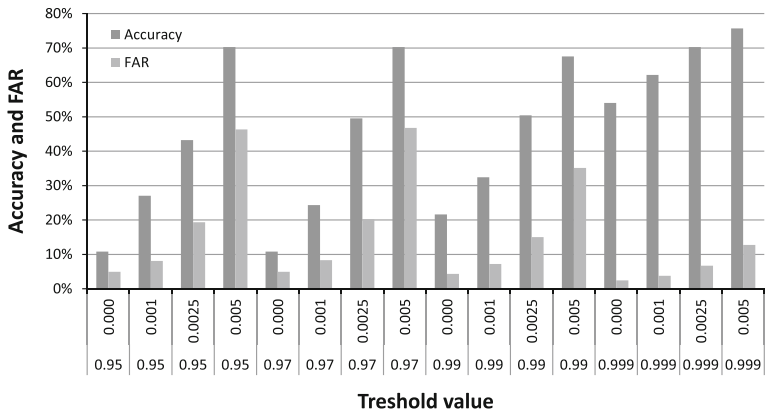


Fig. 1 Classification results for various threshold values

Table 2 Classification results for various threshold values

Explained variability proportion	Similarity threshold	Accuracy (%)	FAR (%)	Ratio of Accuracy and FAR
0.95	0.000	11	5	2.18
0.95	0.001	27	8	3.33
0.95	0.0025	43	19	2.23
0.95	0.005	70	46	1.52
0.97	0.000	11	5	2.18
0.97	0.001	24	8	2.92
0.97	0.0025	50	20	2.45
0.97	0.005	70	47	1.5
0.99	0.000	22	4	4.97
0.99	0.001	32	7	4.5
0.99	0.0025	50	15	3.35
0.99	0.005	68	35	1.92
0.999	0.000	54	2	21.76
0.999	0.001	62	4	16.37
0.999	0.0025	70	7	10.46
0.999	0.005	76	13	5.92

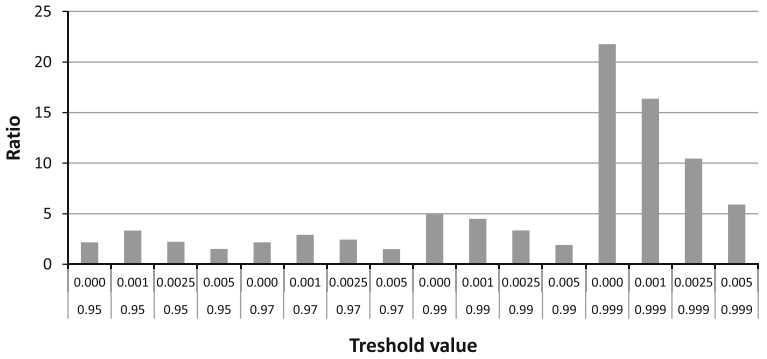


Fig. 2 The ratio of accuracy and FAR for all thresholds

5.2 Random Forest Results

Recursive feature elimination procedure described in Sect. 4.2 was repeated for all 36 datasets. Therefore, finally we obtained 36 different random forests that were used to classify examples from the test dataset (presented results do not include OOB error). During classification, objects from the test dataset were presented to each of the 36 classifiers and the final decision was made based on voting strategy.

Table 3 Classification accuracy obtained for selected number of important features

Number of attributes				
NE	NW	SE	SW	Accuracy (%)
825	810	927	871	42
413	405	464	436	46
207	203	232	218	42
104	102	116	109	46
52	51	58	55	49
26	26	29	28	40
13	13	15	14	31
7	7	8	7	31

Analyses were performed using R Project random forest implementation from *randomForest* package [16]. The number of trees in each forest was set empirically to 1500 (ntree = 1500) and the importance measure was computed as a mean decrease of accuracy (importance parameter type = 1). As we expect that it might exists a correlation among variables, the values of importance were not scaled, that is were not divided by their standard errors (importance parameter scale = FALSE). Classification accuracy obtained for the selected number of important features is presented in Table 3. As each direction (NE, NE, SE and SW) was characterized by different number of attributes, the number of selected features is presented separately for each direction.

Analysis of the results presented in Table 3 shows that reduction of the number of attributes improves classification accuracy. The best results were obtained for the number of attributes around 50. Further reduction of the attribute number decreased the performance of the classifier as too much of the important information was removed from the data description. In addition, we can notice that with the reasonably reduced number of attributes (more than 50), the classification accuracy is around 40–46 %. This is different than in case of the SVM analysis where in case of the full set of attributes, the classification results were very poor.

Finally, for random forest classifier, we have analysed classification accuracy for each of 37 class separately. The results were quite surprising, as we noticed that some of the classes, such as 7, 15, 30 and 35 are classified with quite high accuracy and for others we were able to correctly classify only several objects. This is something that we did not expect and it requires further investigation. The classification accuracy obtained separately for each class is presented in Fig. 3.

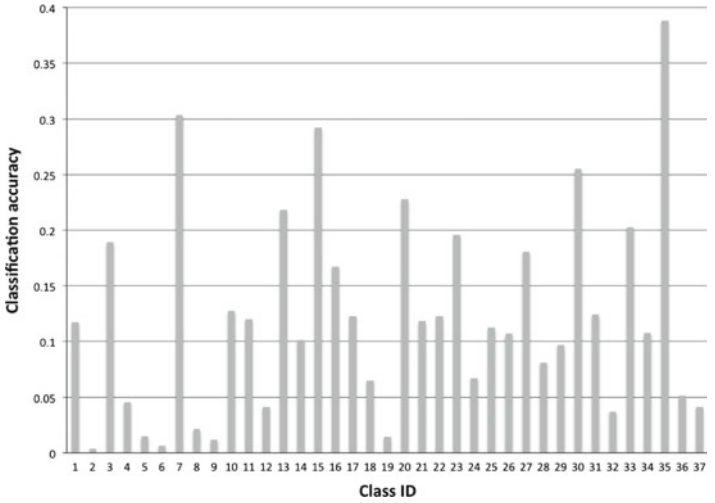


Fig. 3 Classification accuracy computed for separated classes

6 Conclusions

The aim of the research presented in this paper was to elaborate a procedure for classification of individuals based on data obtained from their eye movement signal. The data for the studies was acquired from the public accessed competition, which makes the results obtained in the research comparable with other prospective explorations.

To prepare data for the classification, the set of features was built based on dissimilarities among training samples. The dissimilarities were calculated with DWT metrics. The drawback of such approach for data preprocessing is that as the result it produces a dataset in which each object is described by a huge number of attributes. High dimensionality of obtained dataset makes it difficult to analyse, therefore some additional preprocessing steps are required before selected classification method is applied. The obtained results show that combining DTW data preprocessing method with a dimensionality reduction approach provides the better classification accuracy.

Due to different philosophy of feature extraction versus feature selection it is difficult to directly compare both methods. In case of the combined SVM and PCA method different ranges of data size reduction and their influence on the final result were studied. This outcome confirmed that it is possible to decrease a data size meaningfully without decreasing classification accuracy, even improving it. Applying a threshold parameter allows obtaining better classification results, however it is a trade-off between accuracy of the classifier and a security system false acceptance rate. The second data dimensionality reduction method used in our analysis was random forest procedure for wrapper variable selection. Comparing random forest with SVM method we can see that for full set of features, random forest classifier gives the better results than SVM method. This is something that is expected as the

random forest method is known to be suitable for the high-dimensionality problems. However, by reducing the number of attributes we can still improve accuracy of our random forest classifier.

Another interesting result that we observed during our analyses is that the classification accuracy highly differs among classes. Currently, we are not able to say if it is due to the differences among examined individuals or there was some bias introduced during data acquisition phase.

Summarizing the results, it must be emphasized that their quality is not sufficient to apply in a real authentication process, yet indicate promising directions of the future work.

Acknowledgments The work was partially supported by National Science Centre (decision DEC-2011/01/D/ST6/07007) (A.G). Computations were performed with the use of the infrastructure provided by the NCBIR POIG.02.03.01-24-099/13 grant: GCONiI - Upper-Silesian Center for Scientific Computations.

References

1. Aggarwal, C.C.: Data Classification: Algorithms and Applications. Data Mining and Knowledge Discovery. Hapman and Hall CRC, Boca Raton (2014)
2. Bednarik, R., Kinnunen, T., Mihaila, A., Fränti, P.: Eye-movements as a biometric. In: Kalviainen, H., Parkkinen, J., Kaarna, A. (eds.) Image Analysis. LNCS, vol. 3540, pp. 780–789. Springer, Berlin (2005)
3. Bensch, M., Schroder, M., Bogdan, M., Rosenstiel, W.: Feature selection for high-dimensional industrial data. In: ESANN 2005. pp. 375–380. Bruges, Belgium (2005)
4. Berndt, D.J., Clifford, J.: Using dynamic time warping to find patterns in time series. In: KDD workshop 1994. vol. 10, pp. 359–370. Seattle, USA (1994)
5. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
6. Burges, C.J.C.: Dimension reduction: a guided tour. Found. Trends Mach. Learn. **2**(4), 275–365 (2010)
7. Díaz-Uriarte, R., Alvarez de Andrés, S.: Gene selection and classification of microarray data using random forest. BMC Bioinform. **7**(1), 3 (2006)
8. Gregorutti, B., Michel, B., Saint Pierre, P.: Correlation and variable importance in random forests. arXiv:1310.5726. (2015)
9. Guyon, I., Weston, J., Barnhill, S., Vapnik, V.: Gene selection for cancer classification using support vector machines. Mach. Learn. **46**(1–3), 389–422 (2002)
10. Holzer, S., Ilic, S., Tan, D., Navab, N.: Efficient learning of linear predictors using dimensionality reduction. In: Lee, K., Matsushita, Y., Rehg, J., Hu, Z. (eds.) Computer Vision–ACCV 2012, LNCS, vol. 7726, pp. 15–28. Springer, Berlin (2013)
11. Kasprowski, P., Harezlak, K.: The second eye movements verification and identification competition. In: IJCB 2014. pp. 1–6. Clearwater, USA (2014)
12. Kasprowski, P., Komogortsev, O.V., Karpov, A.: First eye movement verification and identification competition at BTAS 2012. In: BTAS 2012. pp. 195–202. Arlington, USA (2012)
13. Kasprowski, P., Ober, J.: Eye movements in biometrics. In: Maltoni, D., Jain, A.K. (eds.) Biometric Authentication. LNCS, vol. 3087, pp. 248–258. Springer, Berlin (2004)
14. Kohavi, R., Johnb, G.: Wrappers for feature subset selection. Artif. Intell. **97**(1–2), 273–324 (1997)
15. Kursa, M., Jankowski, A., Rudnicki, W.: Boruta—a system for feature selection. J. Am. Water Work. Assoc. Fundamenta Informaticae **101**(4), 271–285 (2010)

16. Liaw, A., Wiener, M.: Classification and regression by randomforest. R News **2**(3), 18–22 (2002)
17. Miller, R., Chen, C., Eick, C., Bagherjeiran, A.: A framework for spatial feature selection and scoping and its application to geo-targeting. In: ICSDM 2011. pp. 26–31. Detroit, USA (2011)
18. Pekalska, E., Duin, R.P., Paclik, P.: Prototype selection for dissimilarity-based classifiers. Pattern Recognit. **39**(2), 189–208 (2006)
19. Saeed, U.: A survey of automatic person recognition using eye movements. Int. J. Pattern Recognit. Artif. Intell. **28**(08), 1456015 (2014)
20. Sikora, M.: Redefinition of decision rules based on the importance of elementary conditions evaluation. Fundamenta Informaticae **123**(2), 171–197 (2013)
21. Sikora, M., Gruca, A.: Quality improvement of rule-based gene group descriptions using information about go terms importance occurring in premises of determined rules. Appl. Math. Comput. Sci. **20**(3), 555–570 (2010)
22. Svetnik, V., Liaw, A., Tong, C., Wang, T.: Application of breiman's random forest to modeling structure-activity relationships of pharmaceutical molecules. In: Roli, F., Kittler, J., Windeatt, T. (eds.) Multiple Classifier Systems. LNCS, vol. 3077, pp. 334–343. Springer, Berlin (2004)
23. Touw, W., Bayjanov, J., Overmars, L., Backus, L., Boekhorst, J., Wels, M., van Hijum, S.: Data mining in the life sciences with random forest: a walk in the park or lost in the jungle. Brief. Bioinform. **14**(3), 315–326 (2013)
24. Vapnik, V., Golowich, S.E., Smola, A.: Support vector method for function approximation, regression estimation, and signal processing. In: NIPS 1996. pp. 281–287. Denver, USA (1996)

Dynamic Time Warping Based on Modified Alignment Costs for Evoked Potentials Averaging

Marian Kotas, Jacek M. Leski and Tomasz Moroń

Abstract Averaging of time-warped signal cycles is an important tool for suppressing noise of quasi-periodic or event related signals. However, in the paper we show that the operation of time warping introduces unfavorable correlation among the noise components of the summed cycles. Such correlation violates the requirements necessary for effective averaging and results in poor suppression of noise. To limit these effects, we redefine the matrix of the alignment costs. The proposed modifications result in significant increase of the noise reduction factor in the experiments on different types and levels of noise.

Keywords Evoked potentials · Dynamic time warping · Nonlinear alignment

1 Introduction

Electroencephalographic evoked potentials respond to a stimulus with a series of positive and negative deflections from the baseline. The time between a stimulus and a particular deflection is called as latency. The latencies convey important information on physiological mechanisms evolving in the brain. This information can be exploited for diagnostic objectives as well as for operating physical devices through the so-called brain-computer interfaces (BCI) [14]. Unfortunately, the evoked potentials are mixed with spontaneous EEG signal and can be of low signal-to-noise ratio. Different types of filtering techniques can be applied to raise this ratio [1, 2, 11, 17]; in cases of a very high level of noise, however, they can rarely produce signals of sufficient quality to detect the important deflections and measure their amplitudes and latencies. Thus of high importance are the methods of averaging [3, 12]. Unfortunately usually

M. Kotas (✉) · J.M. Leski · T. Moroń
Institute of Electronics, Silesian University of Technology, Gliwice, Poland
e-mail: marian.kotas@polsl.pl

J.M. Leski
e-mail: jacek.leski@polsl.pl

T. Moroń
e-mail: tomasz.moron@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_26

the evoked potentials and particularly their latencies differ from stimulus to stimulus and application of classical averaging leads to a limited success.

To overcome these difficulties different modifications of the classical averaging have been developed. In [18] a cross-correlation averaging technique was proposed. In this approach, assuming that the entire responses are of different shifts with respect to their stimuli, the cross-correlation based synchronization precedes their averaging. To overcome more difficult problem, when the individual components of the evoked potentials vary independently in latencies, a technique called as latency corrected averaging was proposed [8], aligning and averaging the individual components of the respective evoked potentials. In experiments during which the information on the total duration of the responses (the so-called response time) is acquired, a technique called as response time corrected averaging [4] can be applied. In this technique the entire responses are adjusted (squeezed or stretched) according to the assumed linear or nonlinear function, to be of the same length.

Application of nonlinear alignment of the evoked potentials, prior to their averaging, performed with the use of the technique of dynamic time warping [13], was proposed in [5, 10]. This method can be applied in all the above-mentioned types of signals variability and does not require any particular, additional information on the processed signals (as e.g. the times of individual responses). Actually, this method meets well not only the demands necessary for effective processing of the evoked potentials but also of other biomedical signals [6, 7]. However, the procedure of dynamic time warping does not distinguish the signal and the noise components of the aligned signals and therefore noise can cause their erroneous alignment [10]. To prevent this effect, in [15] trilinear modeling of evoked potentials was applied to filter single responses before aligning and averaging them. In this study, we present more details showing how unfavorable can be the influence of noise on the results of time warping and averaging.

The objective of this work is to introduce and investigate an approach that can limit these undesired effects.

2 Averaging of Time-Warped Evoked Potentials

2.1 An Outline of Dynamic Time Warping

To align nonlinearly two signals of possibly different length: $x(n), n = 1, 2, \ldots, N_x$ and $y(n), n = 1, 2, \ldots, N_y$, we determine a so-called warping path that consists of the ordered pairs of time indices: $\{(i_k, j_k)|k = 1, 2, \ldots, K\}$, which indicate the elements of the successive aligned pairs: $y(i_k), x(j_k)$. Classically a cost of aligning $y(i)$ with $x(j)$ is defined as [13]:

$$d_{i,j} = (y(i) - x(j))^2 . \tag{1}$$

The warping path minimizes the total cost of the alignment [13]:

$$Q = \sum_{k=1}^{K} d_{i_k, j_k}, \tag{2}$$

preserving a set of constraints. The first of the classical constraints: $i_1 = 1$, $j_1 = 1$, $i_K = N_y$, $j_K = N_x$, force the warping path to start with the pair $x(1)$, $y(1)$ and to end with $x(N_x)$, $y(N_y)$; the second constraints: $i_{k+1} - i_k \leq 1$, $j_{k+1} - j_k \leq 1$, prevent the elements of both signals from being omitted in the warping path, and the last of these classical constraints: $i_{k+1} \geq i_k$, $j_{k+1} \geq j_k$, force the elements of both signals to occur in the warping path in a non-decreasing order.

2.2 Averaging Algorithm

In this study we applied the algorithm proposed in [9]. Let's denote the evoked potentials to be averaged as

$$x_l(n),\ n = 1, 2, \ldots, N_l,\ l = 1, 2, \ldots, L, \tag{3}$$

where L denotes the number of evoked potentials; the lth one is of length N_l.

In the algorithm chosen, one of the evoked potentials is selected as the initial form of the constructed template $t(n)$. Then all successive potentials are aligned with this template, and a set of warping paths is obtained: $\{(i_k^{(l)}, j_k^{(l)}), k = 1, 2, \ldots, K_l\}, l = 1, 2, \ldots, L$. Then the operation called by the authors [9] as the DTW Barycenter Averaging (DBA) is performed: the average of the samples of the respective cycles that were aligned with a template sample $t(n)$ is calculated to update $t(n)$:

$$t(n) = \frac{\sum_{l=1}^{L} \sum_{k \in \Gamma_l(n)} x_l(j_k^{(l)})}{\sum_{l=1}^{L} |\Gamma_l(n)|}, \tag{4}$$

where $\Gamma_l(n) = \{k | i_k^{(l)} = n\}$.

The operation is repeated the assumed number of times and then the individual cycles are reconstructed on the basis of $t(n)$ and the final warping paths.

2.3 Vector Norm as a Cost of Alignment

Averaging of time aligned signals is a classical technique used to suppress noise disturbing the quasi-periodic signals [12]. Various conditions are specified that should be satisfied to achieve suppression of noise and enhancement of the desired compo-

nent as a result of averaging. Among the most important is the lack of correlation among the noise components of the respective signals.

Dynamic time warping can violate this requirement. The ability to repeat any sample of the time-warped signals the requisite number of times, to obtain good matching between two nonlinearly aligned signal cycles, unavoidably introduces increase of the correlation between the noise components of these cycles.

To prevent this unfavorable effect of time warping, we propose to apply the following definition of alignment costs

$$d'_{i,j} = \left\| \mathbf{x}^{(i)} - \mathbf{y}^{(j)} \right\|, \tag{5}$$

where $\|\cdot\|$ denotes the Euclidean norm, and vectors $\mathbf{x}^{(n)}$ and $\mathbf{y}^{(n)}$ are defined as follows

$$\mathbf{x}^{(n)} = [x(n-v), x(n-v+1), \ldots, x(n), \ldots, x(n+v)]^T, \tag{6}$$

which means that they are composed of $2v+1$ successive signal samples (with the nth sample being the central one).

By such a redefinition of a cost matrix, we assure that longer signal intervals are matched to each other instead of single samples of warped signals. An inconvenience, that is caused by this modification of alignment costs, is that to align $x(n), n = 1, 2, \ldots, N_x$ and $y(n), n = 1, 2, \ldots, N_y$ we must have an access to longer signal segments: $x(n), -v < n \leq N_x + v$ and $y(n), -v < n \leq N_y + v$. Fortunately, this requirement is usually satisfied for evoked potentials and other biomedical signals.

3 Influence of Noise on Dynamic Time Warping

Let's investigate a simple task of nonlinear alignment of a sine wave and the same wave with time axis deformed. To deform the time axis, we apply the second order polynomial function proposed in [16]:

$$t_b(z) = z + bz - bz^2 \tag{7}$$

which satisfies the conditions $t(0) = 0$ and $t(1) = 1$; thus, by using different values of parameter b, we can obtain more or less severe deformation of a time axis in the interval $\langle 0, 1 \rangle$, preserving its border values. The desired components of the generated signals are given by

$$s(n) = \sin\left(2\pi\, t_b\left(\frac{n}{N}\right)\right), -v < n \leq N + v. \tag{8}$$

For $y(n)$ we choose $b = 0$ and for $x(n)$, $b = -0.5$; for both signals $N = 400$. Both signals are contaminated with a colored autoregressive noise simulating the spontaneous EEG signal. To show how sensitive to noise is the operation of time

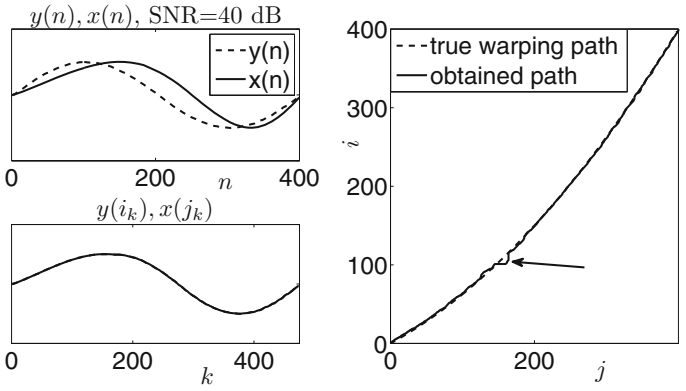


Fig. 1 Nonlinear alignment of two slightly contaminated sine waves (one of them with a deformed time axis): costs of alignment defined by (1) have been applied. The *arrow* indicates the place where the obtained warping path differs from the true one

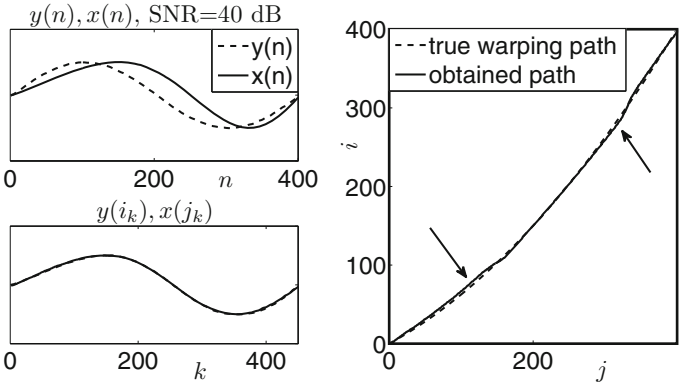


Fig. 2 Nonlinear alignment of two slightly contaminated sine waves (one of them with a deformed time axis): costs of alignment defined by (5) have been applied. The *arrows* indicate the places where the obtained warping path slightly differs from the true one

warping based on the classical definition of a cost matrix, we created the signals of very high quality, with SNR = 40 dB. Results of the signals nonlinear alignment with the cost matrix defined by (1) are presented in Fig. 1. We can see that the noise is almost completely invisible. After the alignment both signals overlapped and are unrecognizable. However, even for such a neglectful level of noise, the obtained warping path differed from the true one in the place indicated by the arrow.

When definition (5) of a cost matrix was applied, slightly different results of nonlinear alignment were obtained (see Fig. 2). The deformation of the warping path caused by noise vanished, but instead a slight lack of precision of the path determination occurred (in the places indicated by the arrows).

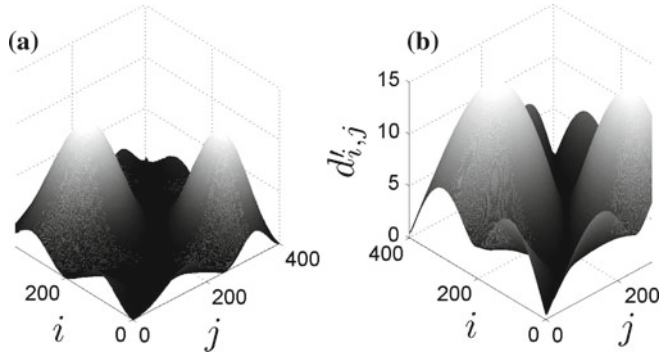


Fig. 3 Visualization of the matrices of costs corresponding to: **a** the nonlinear alignment in Fig. 1, **b** the nonlinear alignment in Fig. 2

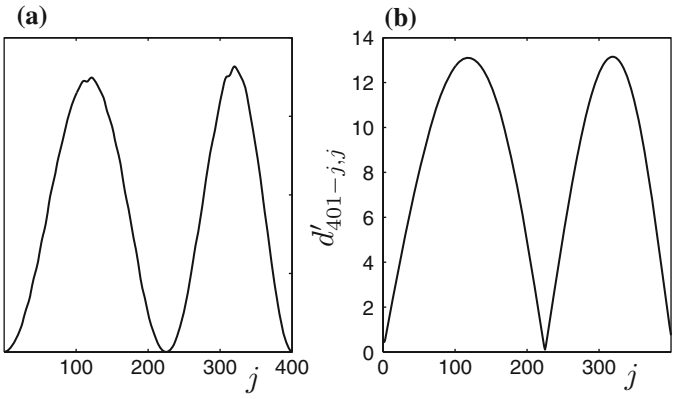


Fig. 4 Cuts across the matrices of costs from Fig. 3

To find the reasons of such differences in these results of time warping, we presented visually (Fig. 3) the cost matrices obtained in the two cases considered. In both cases the cost matrices contain two easily noticeable mountainous regions with high costs corresponding to the wrong alignment of the signals (high values of $y(i)$ at the beginning of this signal with low values of $x(j)$ at the end of that one, and vice verse). However, apart from this similarity, we can distinguish some differences. The mountainous regions in Fig. 3b are wider. This difference can be explained if we compare the applied definitions of costs. Cost (5) can be obtained from costs (1) with the use of a kind of filtering $d'_{i,j} = \left\|\mathbf{y}^{(i)} - \mathbf{x}^{(j)}\right\| = \sqrt{\sum_{z=-v}^{z=v} (y(i+z) - x(j+z))^2} = \sqrt{\sum_{z=-v}^{z=v} d_{i+z,j+z}}$, i.e. by adding $2v+1$ elements of a cost matrix based on (1) and finally taking the square root of the sum.

This filtering causes not only widening of the mountainous regions of the cost matrix, but also narrowing of its valleys, through which the warping path goes. To make it more apparent, in Fig. 4 we have presented the cuts across the both

cost matrices. We can notice that the modified matrix has steeper slopes of these valleys, indeed. With such steep valleys, even slight changes of the warping path can cause significant growth of the costs, while only minor growth would be incurred by such changes in the case of the classical definition of costs. This explains why the modification of the definition of costs makes the time warping more immune to noise, but at the same time slightly less accurate (compare Figs. 1 and 2).

4 Results of Time-Warped Evoked Potentials Averaging

To test the methods described, we formed 5 sets of simulated evoked potentials. Each set consisted of 16 responses. To simulate desired components of the constructed signals, we used the shape function proposed in [16]:

$$f(z) = 15e^{-20(z-0.7)^2} - 0.5e^{-50(z-0.45)^2} + 0.6e^{-100(z-0.3)^2} - 0.6e^{-150(z-0.2)^2} + 0.5e^{-200(z-0.15)^2}. \tag{9}$$

For independent variable t in the range [0, 1] it resembles visual evoked potentials following the target stimuli [11], with the largest peak similar to the so-called P300 wave. Beyond interval [0, 1] the function quickly descends to zero. To deform the time axis of these signals, we used function (7) with different values of parameter b. The desired components of the signals were generated according to the formula

$$s(n) = f\left(t_b\left(\frac{n}{a}\right)\right), n = -99, -98, \ldots, N_s + 100, \tag{10}$$

where parameter a specifies the number of time samples within the interval (0, 1] and $N_s > a$ is the number of samples of the main part of the signal (which undergoes time warping). With the assumed sampling rate of 500 Hz and $N_s = 400$, we obtained signals of 0.8 s length (plus the preceding and succeeding parts of 0.2 s). One of the generated signal sets is presented in Fig. 5. As we can see, for different values of parameters a and b different deformations of the time axis were caused and different final forms of these signals were obtained. By adding white or colored autoregressive noise, we obtained sets of noisy evoked potentials (two selected examples of individual responses are presented at the top of Fig. 6a, b). Each set of 16 simulated noisy responses was processed with the use of DTW barycenter averaging, based on the classical (DBA_C) or modified (DBA_M) alignment costs. For DBA_M 4 different values of parameter v were applied: 25, 50, 75 and 100. Results of signal enhancement were evaluated with the use of the following noise reduction factor:

$$NRF = \sqrt{\frac{\sum_n (x(n) - s(n))^2}{\sum_n (x'(n) - s(n))^2}}, \tag{11}$$

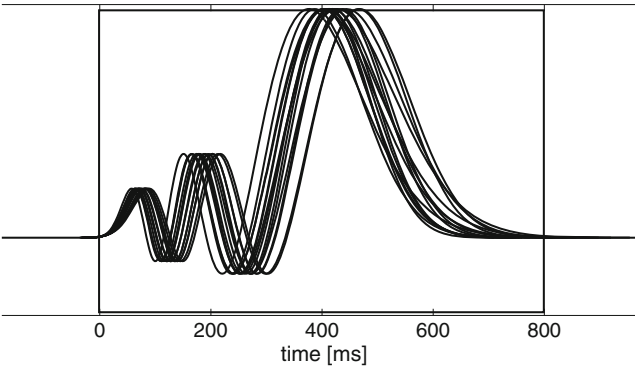


Fig. 5 A set of the simulated evoked potentials

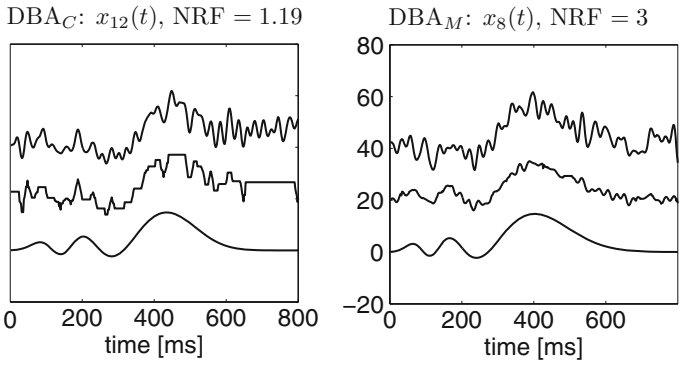


Fig. 6 Results of evoked potentials enhancement with the use of the classical costs based method (DBA$_C$, on the *left*) and the modified costs based one (DBA$_M$, on the *right*). In each subplot there are, from the *top*: the noisy EP, the enhanced EP and the desired component (vertically shifted for better presentation). The evoked potentials for which the highest NRFs were obtained with either DBA$_C$ or DBA$_M$ are presented

where $s(n)$ is the desired simulated EP, $x(n) - s(n)$, the noise added, $x'(n)$, the result of reconstruction and $x'(n) - s(n)$, the residual noise. NRF was calculated for each individual evoked potential and also for the whole sets (to this end, individual responses were regarded as segments of one longer signal).

For both types and three different levels of noise, and for all methods tested, we calculated the average values of NRF, obtained in the test on 5 sets of evoked potentials. The results are presented in Table 1.

We can notice that averaging based on the classical costs of alignment appeared rather ineffective. The obtained values of NRF hardly exceed 1.0. Although the method was described [9] as effective when clustering of different, not very noisy time series was the main objective, it appears less effective with the growth of the noise level. However, application of the modified alignment costs improved the method

Table 1 The average NRF in the test on 5 sets of evoked potentials, contaminated with either white or colored noise of the assumed level (SNR)

Method	White noise			Colored noise		
	5 dB	10 dB	20 dB	5 dB	10 dB	20 dB
DBA_C	1.24	1.24	1.25	1.10	1.10	1.07
DBA_M, $v = 25$	2.36	2.42	2.37	1.63	1.66	1.67
DBA_M, $v = 50$	2.62	2.65	2.12	1.94	1.93	1.78
DBA_M, $v = 75$	2.65	2.56	1.77	2.19	2.16	1.65
DBA_M, $v = 100$	2.78	2.44	1.34	2.34	2.22	1.30

operation significantly. In white noise environment, for SNR $= 5$ dB the best NRF achieved is even close to 3.0. In Table 1 we can see that for high level of noise (SNR $= 5$ dB) the increase of v causes the increase of NRF. For SNR $= 20$ dB, however, too great value of this parameter was unfavorable. Generally, good results for both types and all levels of noise were achieved for $v = 75$.

In Fig. 6 we presented visually results of individual evoked potentials enhancement in one of the processed sets of evoked responses. For both methods presented (DBA_C and DBA_M, with $v = 75$) we selected the evoked potential for which the highest NRF was achieved. We can see that the operation of DBA_C is hardly acceptable (NRF $= 1.19$) but application of the modified costs of alignment made the method much more effective (NRF $= 3.55$). This confirms the quantitative results presented in Table 1.

5 Conclusions

We have applied the classical method of DTW barycenter averaging to the averaging and enhancement of the simulated evoked potentials. The obtained results appeared unacceptable. The noise reduction factor used to evaluate the method operation hardly exceeded the value of 1.0. When however we applied the modified costs of alignment, the method improved its performance significantly. It allowed to obtain quite acceptable evoked potentials even if their original signal-to-noise ratio was rather low.

Acknowledgments This research was partially supported by statutory funds (BK-2015, BKM-2015) of the Institute of Electronics, Silesian University of Technology and GeCONiI project (T. Moroń). The work was performed using the infrastructure supported by POIG.02.03.01-24-099/13 grant: GeCONiI–Upper Silesian Center for Computational Science and Engineering.

References

1. Bartnik, E., Blinowska, K., Durka, P.: Single evoked potential reconstruction by means of wavelet transform. Biol. Cybern. **67**(2), 175–181 (1992)
2. Cerutti, S., Baselli, G., Liberati, D., Avesi, G.: Single sweep analysis of visual evoked potentials through a model of parametric identification. Biol. Cybern. **56**(2–3), 111–120 (1987)
3. Dawson, G.: A summation technique for the detection of small evoked potentials. Electroencephalogr. Clin. Neurophysiol. **6**, 65–84 (1954)
4. Gibbons, H., Stahl, J.: Response-time corrected averaging of event-related potentials. Clin. Neurophysiol. **118**(1), 197–208 (2007)
5. Gupta, L., Molfese, D., Tammana, R., Simos, P.: Nonlinear alignment and averaging for estimating the evoked potential. Trans. Biomed. Eng. **43**(4), 348–356 (1996)
6. Kotas, M.: Projective filtering of time-warped ECG beats. Comput. Biol. Med. **38**(1), 127–137 (2008)
7. Kotas, M.: Robust projective filtering of time-warped ECG beats. Comput. Methods Programs Biomed. **92**(2), 161–172 (2008)
8. McGillem, C., Aunon, J.: Measurements of signal components in single visually evoked brain response. Trans. Biomed. Eng. **24**(3), 232–241 (1977)
9. Petitjean, F., Ketterlin, A., Gancarski, P.: A global averaging method for dynamic time warping, with applications to clustering. Pattern Recognit. **44**(3), 678–693 (2011)
10. Picton, T., Lins, O., Scherg, M.: Dynamic programming algorithm optimization for spoken word recognition. Trans. Signal Process. **26**(1), 43–49 (1978)
11. Quiroga, R.Q.: Obtaining single stimulus evoked potentials with wavelet denoising. Physica **145**, 278–292 (2000)
12. Rompelmann, O.R.H.: Coherent averaging technique: a tutorial review. J. Biomed. Eng. **8**(1), 24–35 (1986)
13. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Trans. Acoust. Speech Signal Process. **26**(1), 43–49 (1978)
14. Schalk, G., McFarland, D., Hinterberger, T., Birbaumer, N., Wolpa, J.: BCI 2000: a general-purpose brain-computer interface (BCI) system. Trans. Biomed. Eng. **51**(6), 1034–1043 (2004)
15. Wang, K., Begleiter, H., Porjesz, B.: Warp-averaging event-related potentials. Clin. Neurophysiol. **112**(10), 1917–1924 (2001)
16. Wang, K., Gasser, T.: Synchronising sample curves nonparametrically. Ann. Stat. **27**(2), 439–460 (1999)
17. Wang, T., Lin, L., Zhang, A., Peng, X., Zhan, C.: EMD-based EEG signal enhancement for auditory evoked potential recovery under high stimulus-rate paradigm. Biomed. Signal Process. Control **8**(6), 858–868 (2013)
18. Woody, C.: Characterisation of an adaptive filter for the analysis of variable latency neuroelectric signals. Med. Biol. Eng. Comput. **5**(6), 539–553 (1967)

Principal Component Analysis and Dynamic Time-Warping in Subbands for ECG Reconstruction

Tomasz Moroń, Marian Kotas and Jacek M. Leski

Abstract The aim of this study was to combine methods from different fields of scientific research, such as dynamic programming, pattern recognition and signal processing to solve a very demanding problem of ECG signal reconstruction in extremely noisy environment. A fast method of signal decomposition into frequency subbands was developed. Its application, and processing the respective subbands with the use of either principal component analysis or dynamic time warping based methods allowed us to achieve a significant progress in suppression of highly intractable electric motion artifacts. The investigations performed showed the proposed method prevalence over the well known nonlinear state-space projections developed in the field of nonlinear dynamics.

Keywords PCA · DTW · Subband decomposition · ECG reconstruction

1 Introduction

Principal component analysis (PCA) is a classical method used to reduce the dimensionality of multivariate data. It has numerous fields of application, including data compression, image analysis, pattern recognition, regression and time series prediction. Its application to ECG processing was described among others in [4–6].

Dynamic time warping (DTW) allows to determine the best alignment of two time series (or sequences of vectors) and to provide a measure of their morphological similarity. This technique was developed for spoken words recognition [11], and later applied in a variety of pattern recognition problems, e.g. to clustering of different biomedical time series data [14]. In this study, the DTW technique is applied to noise

T. Moroń (✉) · M. Kotas · J.M. Leski
Institute of Electronics, Silesian University of Technology, Gliwice, Poland
e-mail: tomasz.moron@polsl.pl

M. Kotas
e-mail: marian.kotas@polsl.pl

J.M. Leski
e-mail: jacek.leski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_27

suppression. This task is extremely important since analysis of biomedical signals is often infeasible without its accomplishment. Suppression of noise is especially difficult when the noise component spectrum overlaps that of the desired component. When, however, the processed signals are of repetitive shape, a relatively simple method of time-aligned (coherent) averaging can be applied [10]. Already in the early years of computerized electrocardiography, this method was successfully applied to ECG signals enhancement [9]. Since then it has gradually been modified and improved [7, 8]. However, its intrinsic constraint of forming an average template of an ECG cycle and loosing the information on its variability was unchanged. Thus new methods, preserving the shapes of the individual cycles were needed. Advantageous results in ECG signal enhancement [5] and even fetal ECG extraction [4] were obtained with the use of the PCA based method. Unfortunately, PCA is less effective in cases of high QT interval variability, associated with a changing position within an ECG cycle of a relatively low T wave. Reconstruction of such signals appeared much more precise when the operation of dynamic time warping preceded the PCA based projections [6]. However, the very troublesome electrode motion artifacts, resulting from patients movements can disturb the operation of dynamic time warping and as a result negatively affect reconstruction of ECG signals.

The goal of this paper is to combine the classical methods: principal component analysis and dynamic time-warping in a novel way for noise immune reconstruction of ECG signals.

2 Methods

2.1 Principal Component Analysis

Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] = [x_{i,j}]_{i=1,j=1}^{i=d,j=n}$ be a matrix containing n observed d-dimensional variables. The measured variables compose a cloud of points in a d-dimensional measurement space. PCA objective is to project the points into a lower dimensional principal subspace that retains the maximum amount of information (as measured by variance). This operation is accomplished according to the following equation [3]

$$\mathbf{y}_j = \mathbf{E}^\top (\mathbf{x}_j - \bar{\mathbf{x}}), \tag{1}$$

where $\mathbf{E} = [\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_q]$.

The orthonormal principal directions (axes) $\mathbf{e}_i$ ($i = 1, 2, \ldots, q$) have the following properties: the first principal axis $\mathbf{e}_1$ assures the maximal variance of the points projected on it, the second principal axis $\mathbf{e}_2$ assures the greatest variance of the points projected on it in the subspace perpendicular to $\mathbf{e}_1$, similarly the third principal axis $\mathbf{e}_3$ assures the greatest variance of the points projected on it in the subspace perpendicular to $\mathbf{e}_1$ and $\mathbf{e}_2$, etc. The directions can be determined by eigendecomposition

of the $\mathbf{X}\mathbf{X}^T$ matrix: $\mathbf{e}_i$ is the eigenvector corresponding to the ith eigenvalue (sorted in a decreasing order).

Reconstruction of the observed d-dimensional variable is given by [3]

$$\mathbf{x}'_j = \mathbf{E}\mathbf{y}_j + \bar{\mathbf{x}} = \mathbf{E}\mathbf{E}^\top(\mathbf{x}_j - \bar{\mathbf{x}}) + \bar{\mathbf{x}}. \tag{2}$$

By this operation, we obtain the points that lie on a q-dimensional linear manifold that are nearest to the projected ones. Assuming that the desired components of the projected points are confined to this manifold, and that the deviations from this manifold result from the presence of noise, we can expect the projections to suppress noise.

2.2 Dynamic Time-Warping

Lets consider two signals of possibly different length: $v(n), n = 1, 2, \ldots, N_v$ and $u(n), n = 1, 2, \ldots, N_u$. To perform their nonlinear alignment, we calculate the alignment costs

$$d_{i,j} = (u(i) - v(j))^2 . \tag{3}$$

Each $d_{i,j}$ corresponds to the alignment of $u(i)$ and $v(j)$. The warping path that relates signals $v(n)$ and $u(n)$ consists of the ordered pairs of time indices: $\{(i_k, j_k)|k = 1, 2, \ldots, K\}$, which indicate the elements of the successive aligned pairs: $u(i_k)$, $v(j_k)$. In the classical approach, the warping path is subject to the following constraints [11]:

- Boundary conditions: $i_1 = j_1 = 1,\ i_K = N_u,\ j_K = N_v$, which force the warping path to start with the pair $u(1)$, $v(1)$ and to end with $u(N_u)$, $v(N_v)$.
- Continuity conditions: $i_{k+1} - i_k \leq 1,\ j_{k+1} - j_k \leq 1$, which prevent the elements of both signals from being omitted in the warping path.
- Monotonicity conditions: $i_{k+1} \geq i_k,\ j_{k+1} \geq j_k$, which force the elements of both signals to occur in the warping path in a non-decreasing order.

The warping path that minimizes the total cost of the alignment [11]:

$$Q = \sum\nolimits_{k=1}^{K} d_{i_k, j_k} \tag{4}$$

is searched for. It can be found with the use of dynamic programming.

First a matrix of cumulative costs $\mathbf{G} = [g_{i,j}]_{i,j=1}^{i=N_u, j=N_v}$ is constructed, according to the recursive definition [11]:

$$g_{i,j} = d_{i,j} + \min\left\{g_{i-1,j-1},\ g_{i-1,j},\ g_{i,j-1}\right\}, \tag{5}$$

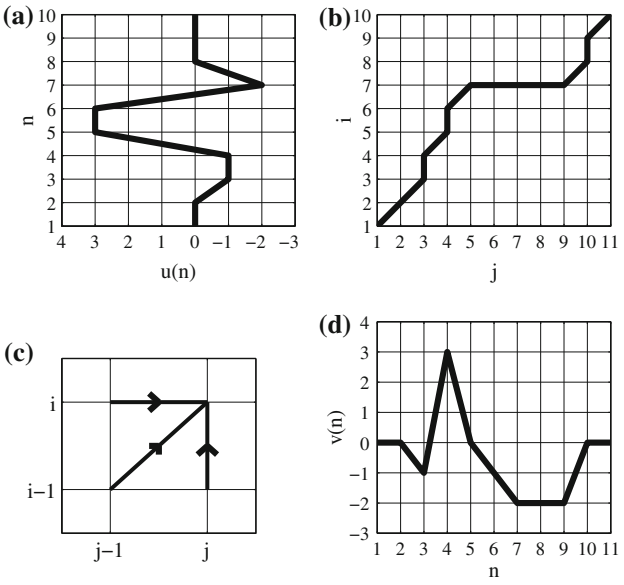


Fig. 1 Graphical representation **c** of the allowed step directions corresponding to Eq. (5) and an example of a warping path **b**: (i, j) = {(1, 1), (2, 2), (3, 3), (4, 3), (5, 4), (6, 4), (7, 5), (7, 6), (7, 7), (7, 8), (7, 9), (8, 10), (9, 10), (10, 11)} relating two signals: $u(n)$ **a** and $v(n)$ **d**

where $g_{1,1} = d_{1,1}$, and $g_{0,j} = \infty$, $g_{i,0} = \infty$ for all i, j (the pictorial representation of this definition, which shows the allowed step-directions of the warping path, is presented in Fig. 1c).

Then, starting from the upper right corner of **G** ($i = N_u,\ j = N_v$), the warping path is searched for by backtracking along the allowed step-directions through the minima of the cumulative costs in **G**, until $i = 1$ and $j = 1$. While backtracking, the allowed step-directions are opposite to those presented in Fig. 1c.

Applied to time warping of the signals $v(n)$ and $u(n)$, this classical approach produced the warping path presented in Fig. 1b. We can notice that to achieve nonlinear alignment of both signals, some elements of $u(n)$ and $v(n)$ had to occur more that ones in the warping path.

2.3 Averaging of Time-Warped Signal Segments

In this study, we applied the method proposed in [1] for evoked potentials reconstruction. It was called by the authors as NonLinear Alignment Averaging Filter (NLAAF).

The method is applied to process L signal segments of possibly different length: $x_l(n),\ n = 1, 2, \ldots, N_l,\ l = 1, 2, \ldots, L$, where N_l denotes the lth segment length.

We take the first segment $x_1(n)$ as the first template $t_1(n)$ and we align the template nonlinearly with the subsequent segments; each time after the alignment, the template is updated.

By warping the current $(l-1)$th template with the next lth segment, we obtain the lth warping path consisting of the successive pairs: $\{(i_k^{(l)}, j_k^{(l)})|\ k = 1, 2, \ldots, K_l\}$, whose number K_l (the length of the warping path) becomes the length of the new updated template

$$t_l(k) = \frac{(l-1)t_{l-1}(i_k^{(l)}) + x_l(j_k^{(l)})}{l},\ k = 1, 2, \ldots, K_l. \tag{6}$$

By adding the successive warped segments to the template, we gradually increase its length. The warping paths relating the respective segments with the template change after each time-alignment. After adding the last segment, we obtain the final template, $t_L(k)$. Then, knowing the warping paths relating this template with the individual segments, we reconstruct these segments.

2.4 Moving Average Based Subband Decomposition

Moving average (MA) filter and its modifications are often applied to ECG signals processing because of the filter linear phase response and its low computational costs. The filter calculates the mean of the signal samples that appear in a moving time window of the assumed length. Different forms of its system function

$$MA_K(z) = \frac{1}{K}\sum_{k=0}^{K-1} z^{-k} = \frac{1}{K}\frac{1-z^{-K}}{1-z^{-1}} \tag{7}$$

show that the filter can be applied in either nonrecursive or recursive way. Particularly favorable is the latter, very fast solution.

The MA low-pass filter can be used to form a high-pass filter with the following system function

$$H_K(z) = z^{-(K-1)} - MA_K^2(z) \tag{8}$$

By the proper choice of parameter K, we can obtain a filter appropriate for baseline wander suppression. The cut-off frequency of such a filter should not exceed about 0.8 Hz [13] to avoid suppression of diagnostically important components of the ECG signal. Unfortunately a filter with such a low cut-off frequency can not suppress the very troublesome low frequency noise caused by patients movements. This type of noise, called as electrode motion artifacts, can reach up to about 5 Hz and is extremely difficult to be dealt with. It spoils the action of most modern sophisticated methods of ECG noise suppression. However, although we can not remove this part of ECG spectral components permanently, we can split the signal into subbands to process each subband in a specific manner and then combine the obtained results together.

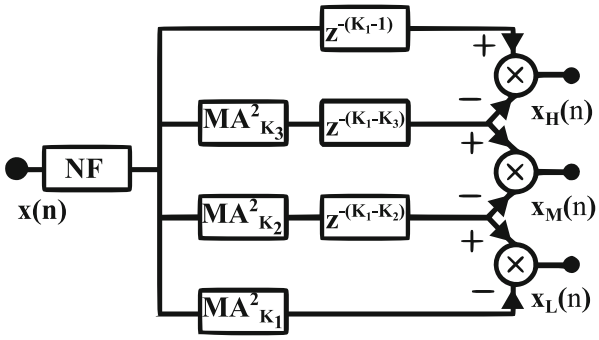


Fig. 2 Moving average based decomposition of signals into frequency subbands. NF is a notch filter for powerline suppression

The scheme showing how the ECG signal can be decomposed is presented in Fig. 2. First we apply a notch filter to suppress the powerline interference. Then we proceed with the subband decomposition. The lowest branch of the scheme contains the MA filter used to suppress the baseline wander. It is the moving average used in Eq. (8) to form a high pass filter of the appropriate cut-off frequency ($\leq$0.8 Hz). The upper branches contain the MA filters of shorter impulse responses and, consequently, of higher cut-off frequencies. Since these filters cause shorter delay, they are purposefully delayed to assure the same delay in each branch. The uppermost branch contains simply an appropriate delay. Owing to the same delay in all branches of the scheme, we can subtract the outputs of the respective filters to extract the required frequency components not introducing any phase distortions. The subtractions proposed allow us to obtain:

- signal $x_H(n)$ containing the highest frequency components,
- signal $x_M(n)$ containing the medium frequency components and
- signal $x_L(n)$ containing the lowest frequency components of the processed ECG.

Our goal is to form the $x_H(n)$ signal that contains components of the fast changing QRS complex only, and $x_M(n) + x_L(n)$ signal that embraces all other lower frequency components; in $x_M(n)$ we aim to preserve the medium frequency components of the ECG with electrode motion artifacts suppressed as much as possible. Of course, adding more branches to the scheme, we could decompose the signal into more subbands.

2.5 ECG Signal Reconstruction in Subbands

In the first stage of ECG signal processing, we perform QRS complexes detection. This way we obtain so-called fiducial points r_l, $l = 1, 2, \ldots, L$, corresponding to the central position within the respective detected L complexes. Since after the described

decomposition, the highest frequency component $x_H(n)$ contains the desired signal in places of QRS complexes only, we can discard noise elsewhere, be setting $x'_H(n) = 0$ for $r_l + \Delta \leq n < r_{l+1} - \Delta,\ l = 1, 2, \ldots, L-1$ (where Δ denotes half of the assumed width of a QRS complex, $\Delta \in \mathcal{N}$). By contrast, the segments containing successive complexes: $x_H(n), r_l - \Delta \leq n < r_l + \Delta$, are stored in columns of an auxiliary matrix $\mathbf{X}$. Then we perform PCA and we reconstruct the respective complexes according to Eq. (2) with the assumed dimension q of the principal subspace. Then the recovered complexes are written to the reconstructed signal $x'_H(n)$.

The sum $x_{ML}(n) = x_M(n) + x_L(n)$ containing all medium and low frequency components of the processed signal is cut into the following $L-1$ time segments: $r_l \leq n < r_{l+1},\ l = 1, 2, \ldots, L-1$. These signal segments undergo time-warping and averaging according to the NLAAF method. Afterwards, the recovered segments are written to the reconstructed signal $x'_{ML}(n)$. Finally, we reconstruct the ECG signal as a sum: $x'(n) = x'_H(n) + x'_{ML}(n)$. The described method will be denoted as ESRS_1 which stands for **E**cg **S**ignal **R**econstruction in **S**ubbands, version number one.

In the second version of the method, we process the highest frequency band as in ESRS_1 but, before applying NLAAF to medium and low frequency band, we cut into segments not only the sum $x_{ML}(n)$, but also the medium frequency subband $x_M(n)$. Then we calculate the warping paths by minimizing Eq. (4) with the alignment costs defined by

$$d'_{i,j} = (u_M(i) - v_M(j))^2 \tag{9}$$

where as before subscript M denotes the medium frequency subband of the aligned signal segments. This way we limit the influence of the troublesome electric motion artifacts on the results of time-warping. However, after determination of warping paths, we perform construction of two templates, both in the $x_M(n)$ and $x_{ML}(n)$ subbands. Template $t_M(n)$ is constructed only to enable its alignment with the successive segments. Template $t_{ML}(n)$ is used to recover the desired component of the $t_{ML}(n)$ subband. Finally, like in ESRS_1, in ESRS_2 (version number 2) we reconstruct the ECG signal as a sum: $x'(n) = x'_H(n) + x'_{ML}(n)$.

3 Results

To test the proposed methods performance, we applied them to processing the high SNR signals selected from the MIT-BIH database and used to simulate the desired ECG. The MIT-BIH database also contains the records of the electrode motion artifacts ('em.dat') and muscular noise ('ma.dat'). We added these two records to obtain an example of noise that can be encountered in real ECG signals. This noise was used to contaminate the selected 'desired' ECGs. We formed and then processed signal segments of 20 s length. For the sampling frequency of 360 Hz, we chose the following parameters of the decomposing filters: $K_1 = 250$, $K_2 = 40$ and $K_3 = 20$. Knowing the simulated desired components, we can evaluate the results of reconstruction with the use of the following noise reduction factor:

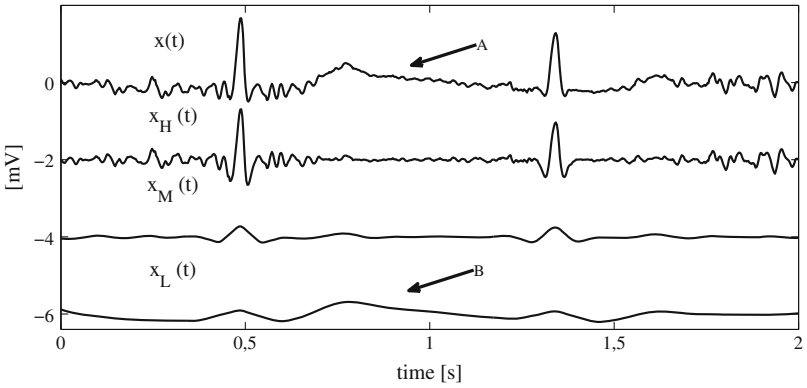


Fig. 3 ECG signal decomposition into frequency subbands. *Arrow* A indicates a local increase of the $x(t)$ baseline, caused by electrode motion artifacts. This increase was almost exclusively preserved in the lowest frequency subband (see *arrow* B)

$$NRF = \sqrt{\frac{\sum_n (x(n) - s(n))^2}{\sum_n (x'(n) - s(n))^2}}, \tag{10}$$

where $s(n)$ is the desired ECG, $x(n) - s(n)$, the noise added, $x'(n)$, the result of reconstruction and $x'(n) - s(n)$, the residual noise. An example of the filters operation is presented in Fig. 3 (for better presentation the respective signals are shifted in the vertical direction by a multitude of 2 mV, as well as the signals in Figs. 4 and 5). We can notice that the lowest frequency subband: $x_L(t)$, contains significant part of the desired ECG and, therefore, its rejection during the final ECG reconstruction would distort this signal. However, arrow B in the figure shows that this subband

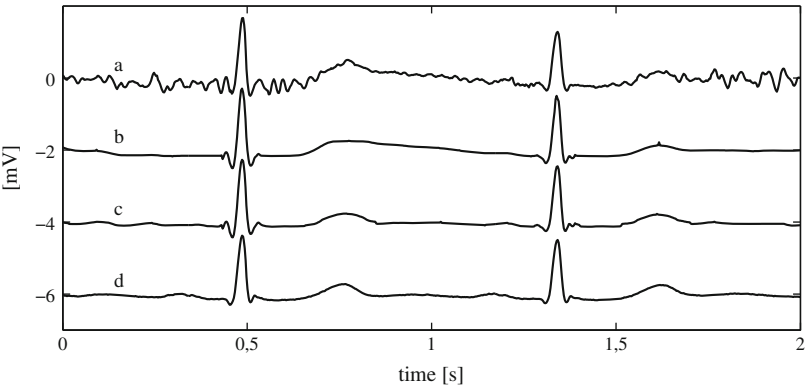


Fig. 4 ECG signal reconstruction in subbands: **a** the processed signal, **b** results of $ESRS_1$, **c** results of $ESRS_2$, **d** the desired ECG

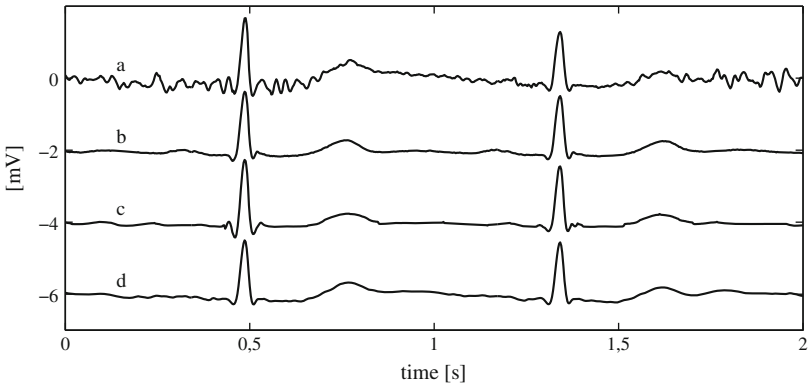


Fig. 5 ECG reconstruction in subbands versus nonlinear state-space projections: **a** the noisy ECG, **b** its desired component, **c** results of $ESRS_2$, **d** results of NSSP

$(x_L(t))$ contains also the troublesome electrode motion artifacts. These low frequency disturbances can utterly spoil the results of ECG segments time warping and make the NLAAF method operation ineffective. This is visible in Fig. 4b where the results of $ESRS_1$ are presented. As we can see, although the wide spectrum muscular noise was suppressed, the obtained signal does not resemble its desired component (D). The low amplitude waves between the both QRS complexes on the figure are utterly distorted. Fortunately, although we can not reject the lowest frequency subband during the final ECG reconstruction, we can diminish its unfavorable influence on this reconstruction. To this end, in $ESRS_2$ the operation of warping paths determination is based on the medium frequency subband $(x_M(t))$. The results are indisputable. In Fig. 4c we can see the reconstructed ECG signal which resembles its desired component (D) much better than the signal obtained with the use of $ESRS_1$.

Similar experiments with ECG enhancement by application of a few modern, rather sophisticated methods were presented in [2]. The most effective among the methods tested appeared the method of nonlinear state-space projections, developed in the field of nonlinear dynamics and applied to ECG processing in [12]. In Fig. 5 we compare this classical method to the method proposed ($ESRS_2$). We can observe that the NSSP method enhanced the signal quite well but, contrary to the method proposed, it did not manage to suppress the low frequency electric motion artifacts well (compare trace D to trace B). These visual results are confirmed by the quantitative ones, presented in Table 1. For medium quality signals (SNR $= 10$ dB) the $ESRS_2$ and NSSP methods obtained similar values of the noise reduction factor. For more noisy signals, $ESRS_2$ prevailed NSSP significantly.

Table 1 The average NRF in the test on 5 desired ECGs contaminated with a sum of muscular and electrode motion artifacts of the assumed level

Method/SNR:	0 dB	5 dB	10 dB
$ESRS_1$	1.59	1.50	1.32
$ESRS_2$	2.79	2.38	1.82
NSSP	1.55	1.78	1.81

4 Conclusions

We have combined two methods from the field of pattern recognition for the purpose of ECG signals reconstruction and enhancement. The first of them, principal component analysis, is used to reconstruct highly variable QRS complexes. The second method, of dynamic time warping, to align nonlinearly low frequency components of the signal for the purpose of their averaging. To employ both methods effectively, we have developed a method of ECG signals decomposition into three frequency subbands. PCA is applied for the highest frequency subband processing, while the two lowest frequency subbands undergo time warping and averaging. Decomposition into three subbands allowed us to modify the operation of dynamic time warping to make the method be more immune to the very troublesome electrode motion artifacts. It helped to improve reconstruction of the desired ECG significantly. The developed method ($ESRS_2$) appeared more immune to this type of noise than the widely acknowledged method of nonlinear state-space projections.

Acknowledgments This research was partially supported by statutory funds (BK-2015, BKM-2015) of the Institute of Electronics, Silesian University of Technology and GeCONiI project (T. Moroń). The work was performed using the infrastructure supported by POIG.02.03.01-24-099/13 grant: GeCONiI–Upper Silesian Center for Computational Science and Engineering.

References

1. Gupta, L., Molfese, D., Tammana, R., Simos, P.: Nonlinear alignment and averaging for estimating the evoked potential. IEEE Trans. Biomed. Eng **43**(4), 348–356 (1996)
2. Hu, X., Nenov, V.: A single-lead ecg enhancement algorithm using a regularized data-driven filter. IEEE Trans. Biomed. Eng. **53**(2), 347–351 (2006)
3. Jollife, I.: Principal component analysis. Springer, New York (2002)
4. Kotas, M.: Projective filtering of time-aligned beats for fetal ecg extraction. Bull. Pol. Acad. Sci. Tech. Sci. **55**(4), 331–339 (2007)
5. Kotas, M.: Projective filtering of time-aligned ecg beats for repolarization duration measurement. Comput. Methods Programs Biomed. **85**(2), 115–123 (2007)
6. Kotas, M.: Robust projective filtering of time-warped ecg beats. Comput. Methods Programs Biomed. **92**(2), 161–172 (2008)
7. Leski, J.M.: Robust weighted averaging. Trans. Biomed. Eng. **49**(8), 796–804 (2002)
8. Momot, A.: Methods of weighted averaging of ecg signals using bayesian inference and criterion function minimization. Biomed. Signal Process. Control. **4**(2), 162–169 (2009)

9. Rautaharju, P., Blackburn, H.: The exercise electrocardiogram: experience in analysis of noisy cardiograms with a small computer. Am. Heart J. **69**(4), 515–520 (1965)
10. Ros, O., Rompelmann, H.: Coherent averaging technique: a tutorial review. J. Biomed. Eng. **8**(1), 24–35 (1986)
11. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. Trans. Acoust. Speech Signal Process. **26**(1), 43–49 (1978)
12. Schreiber, T., Kaplan, D.: Nonlinear noise reduction for electrocardiograms. Chaos **6**(1), 87–92 (1996)
13. Van Alste, J.A., Van Eck, W., Herrmann, O.E.: Bag-of-words representation for biomedical time series classification. Comput. Biomed. Res. **19**(5), 417–427 (1986)
14. Wang, J., Liu, P., She, M.: Bag-of-words representation for biomedical time series classification. Biomed. Signal Process. Control. **8**(6), 634–644 (2013)

Part VI
Image and Motion Data Processing

Evaluating of Selected Systems for Colorimetric Calibration of LCD Monitors

Artur Bal, Andrzej Kordecki, Henryk Palus and Mariusz Frąckiewicz

Abstract In many applications there is a need for exact colour reproduction. The common solution of this problem is the usage of an open Colour Management System and device profiles. The profiles are determined in a colorimetric calibration process, so the quality of this process influences colour reproduction quality. The aim of this work was to evaluate the colour reproduction quality of monitors depending on used types of monitor and calibration system. Obtained results show that colour reproduction strongly depends on those devices and good results can be achieved even on a monitor not designed for colour critical works. However, in case where exact colour reproduction is crucial the use of a professional-grade graphical monitor is necessary.

Keywords Colour reproduction · Colour management · Monitor calibration · LCD monitor

1 Introduction

Nowadays visual information plays an important role. With the use of visual information, especially when it is analysed by a human being, problem of exact colour reproduction is related [3]. In some applications accuracy of colour reproduction is critical. Good examples of such applications are: diagnostic radiology, DTP and CAD systems, e.g. [1, 2, 9]. Additionally, there are many applications in which correct colour reproduction is required, e.g.: telemedicine systems, multi-screen work-

A. Bal (✉) · A. Kordecki · H. Palus · M. Frąckiewicz
Institute of Automatic Control, Silesian University of Technology, Gliwice, Poland
e-mail: artur.bal@polsl.pl

A. Kordecki
e-mail: andrzej.kordecki@polsl.pl

H. Palus
e-mail: henryk.palus@polsl.pl

M. Frąckiewicz
e-mail: mariusz.frackiewicz@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_28

ing environments, digital signage systems and distributed systems, e.g. [7, 8, 10]. Because of technical reasons, in practice, with no additional solutions the same colour—understood as an ordered vector of three colour components (R, G, B)—on different systems (even if those systems have the same hardware and software components) will be reproduced differently in terms of colour sensations induced in an observer.

For solving this problem a *Colour Management System* (CMS) can be used and its most popular type is *open CMS*. Examples of such CMS type are ColorSync introduced by Apple Inc. and ICC Color Management developed by the International Color Consortium (ICC). A characteristic feature of an open CMS is usage of a standard colour space called *Profile Connection Space* (PCS), which is a common colour space for all colour transformations executed in CMS between devices or software dependent colour spaces. Each colour conversion between PCS (which is device-independent) and a device-dependent colour space is realized using profiles. A common implementation of such concept of colour management is the standard developed by the ICC and approved by the International Organization for Standardization as the ISO 15076-1 standard [4, 5].

Profiles are determined in so-called *colorimetric calibration process*—henceforth in this paper this process will be referred simply as *calibration*. In order to determine a profile of an image reproducing device, it is necessary to measure the colours reproduced by this device. Calibration results may vary and depend on many factors e.g. type of a calibrated monitor, type and quality of a measuring device (i.e. colorimeter or spectroradiometer), type and settings of a software used for calibration.

The aim of the research presented in this study was objective evaluation of colour reproduction quality in dependency on a type of calibrated monitor, as well as hardware and software used for calibration. This issue is presented in the following order. In the Sect. 2 the purpose of the research and the necessary assumptions are detailed described. The Sect. 3 presents the methodology and the measurement conditions. The Sect. 4 contains the results of the measurements. The last section provides a short summary of the paper.

2 Aim of the Study and the Necessary Assumptions

The aim of this study was to examine the influence of used monitors and calibration systems on the quality of colour reproduction of the monitors. The used evaluation procedure is based on measurements of colours reproduced by a calibrated monitor and its comparison with required colours defined in a PCS. This procedure was repeated for each monitor-calibration set-up (i.e. a pair of monitor and calibration method or system which was used for its calibration).

This research's goal is important from the practical point of view. It was assumed that the study should take into account only the typical working conditions (e.g. monitor settings, ambient light) that are common for monitors used in applications like CAD/CAM and web pages design. The applications associated with a professional

Table 1 Main parameters of the tested monitors

Manufacturer	BenQ	Dell	Dell	NEC
Model	XL2420T	UltraSharp U2412M	UltraSharp U2410	SpectraView 241
Diagonal screen	24″	24″ (60.96 cm)	24″ (60.96 cm)	24″ (61.1 cm)
Resolution and aspect ratio	1920 × 1080 16 : 9	1920 × 1200 16 : 10	1920 × 1200 16 : 10	1920 × 1200 16 : 10
Panel technology	TN	e-IPS	H-IPS	H-IPS
Panel manufacturer and model	AU Optronics M24HW01 V8	LG Display LM240WU8-SLA2	LG Display LM240WU8-SLA2	LG Display LM240WU4-SLB1
Panel backlighting	W-LED	W-LED	CCFL	CCFL
Panel colour depth	6-bit + H-FRC	6-bit + A-FRC	8-bit + A-FRC	8-bit + Hi-FRC
sRGB and AdobeRGB gamut coverage	93.6 %	95 %	100 %	100 %
	72.4 %	74 %	98.1 %	98.1 %
Monitor static contrast	1000 : 1	1000 : 1	1000 : 1	1000 : 1
Monitor colour depth	8-bit/channel	8-bit/channel	10-bit/channel	10-bit/channel

graphic design (e.g. DTP, photo editing) have not been considered because of high diversity of its required working conditions.

The experiments were performed for four different monitors i.e. from the good-quality monitor for home use with TN panel (BenQ XL2420T) up to the professional graphic monitor with IPS panel (NEC SpectraView 241). Main parameters of these monitors are presented in Table 1. Data in the table are provided by the manufacturers of monitors or were made available by specialized websites like Prad (www.prad.de) and TFT Central (www.tftcentral.co.uk).

Tested monitors have different gamuts (i.e. range of colours which can be reproduced by a monitor) therefore the evaluation of colour reproduction quality was limited to the colours which are located in one common colour space, namely in the sRGB colour space (IEC 61966-2-1) [6]. This colour space is in practice the most basic colour space used in commercial applications. Moreover the sRGB space, in accordance with the decision of the World Wide Web Consortium (W3C) [12], is the standard colour space used to describe colours on web pages. The sRGB space is also the default colour space used in the Microsoft Windows operating system [11]. All these evidences allow suppose that correct reproduction of colours from the sRGB space is the minimum requirement which have to be met by a monitor used in commercial applications.

Table 2 Main parameters and settings of the tested calibration systems

Measurement device			
Manufacturer	Datacolor	X-Rite	
Model	Spyder4ELITE	i1 Display Pro	i1Photo Pro 2
Device type	Colorimeter, 7-channel	Colorimeter	Spectroradiometer
Software			
Manufacturer	Datacolor	X-Rite	
Name	Spyder4ELITE	i1 Profiler	
Version	4.5.0	1.5.6	
Calibration settings			
Black luminance	0 cd/m^2	—	
White luminance	120 cd/m^2	120 cd/m^2	
Contrast ratio	—	Native	
White point	6500 K	CIE illuminant D65	
γ coefficient	2.2	2.2	
Number of colour patterns	Unknown no possibility of choice	462	
Other program specific settings	Ambient Light Compensation—off	Flare Correct—off	
	Gray Balance Calibration—on	Ambient Light Smart Control—off	
Profile type	Matrix can not be changed	Table	
ICC profile version	4	4	
Chromatic adaptation mode	Bradford	Bradford	
Automatic Display Control	—	Yes, with compatible monitors	
Calibration settings related to the type of monitor			
BenQ XL2420T Dell	Normal gamut White LED	White LED	Lack of choice due to the type of measurement device
Ultrasharp U2412M Dell			
Ultrasharp U2410 NEC	Wide Gamut Fluorescent (CCFL)	Wide Gamut CCFL	
Spectraview 241			

For both X-Rite systems the data about its software and calibration settings are the same

The white point of the sRGB space is defined by $x = 0.3127$ and $y = 0.3290$ coordinates in the CIE xyY space what corresponds to the light with *Correlated Colour Temperature* (CCT) equal 6500 K (the CIE D65 standard illuminant). Therefore this white point coordinates were used as required settings during monitor

Fig. 1 Measurement devices of the calibration systems used in the research, **a** Datacolor Spyder4ELITE, **b** X-Rite i1 Display Pro, **c** X-Rite i1Photo Pro 2

calibration. According to the sRGB standard the luminance of a monitor white point should be equal to $L_{white} = 80\,\mathrm{cd/m^2}$, but this luminance value is too low for a typical work place in which influence of the day light is not eliminated. For this reason, taking into account the earlier assumptions, it was assumed that required luminance level of the monitor white point is $L_{white} = 120\,\mathrm{cd/m^2}$.

Because only one monitor (NEC) has the ability of hardware calibration the research has been limited to the software calibration, which can be carried for each monitor. During monitor calibration three systems for software calibration were used. The basic information about usd systems is presented in Table 2, and in Fig. 1 the measurement devices (i.e. calibrators) of these systems are shown.

The monitor profile and therefore the quality of reproduced colours depend strongly on settings of calibration software. These settings may vary depending on e.g. the type of calibrated monitor, the aim of a calibration and individual user preferences. Therefore, during the research, for each tested pair of the calibration system and the monitor those settings were individually selected in order to achieve the best results in sense of reproduced colour quality (Sect. 4).

3 The Measurement Conditions and Procedures

All measurements were performed in the darkroom belonging to the Laboratory of Imaging and Radiometric Measurements, in the Institute of Automatic Control of the Silesian University of Technology (Fig. 2). The work in the darkroom eliminates the influence of ambient light on the measurements. During measurements the average level of illuminance was $\bar{E} < 1\,\mathrm{lx}$. In the darkroom air temperature was kept constant at approx. 22 °C. At the same temperature all calibrators and monitors used for tests were stored. Before calibration and subsequent measurements each monitor was always warmed up for at least 40 min.

During the research a personal computer with MS Windows 7 Professional 64-bit and NVIDIA Quadro NVS 450 graphics card was used. Monitors were connected to the computer via a DVI-D interface. Before monitor calibration both the computer's operating system and a monitor were reset, respectively, to default values and to factory settings.

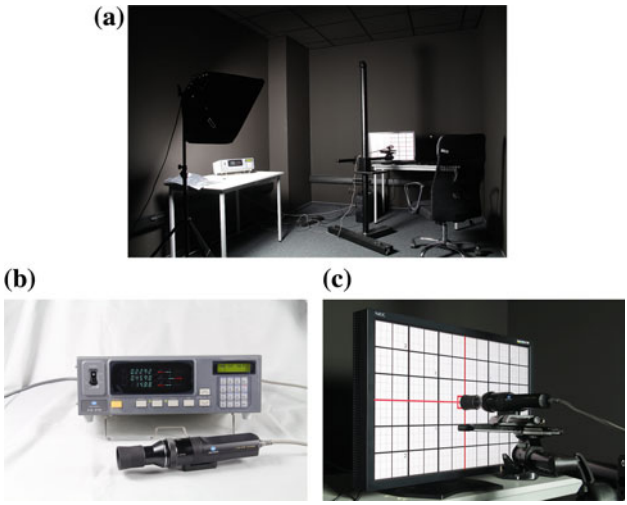


Fig. 2 Equipment used for measurements: **a** the darkroom, **b** Konica Minolta CA-310 colorimeter, **c** CA-310 measuring head during its positioning on a screen surface

The measurements of colours displayed by monitors were performed with use of a professional colorimeter Konica Minolta CA-310 with an universal head CA-PU32/35 (Fig. 2b). This device gives high accuracy and precision measurement results. According to the manufacturer's instruction before each measurement the colorimeter was turn-on for not less than 30 min.

Because all the tested calibrators calculate a monitor profile based on measurements of a central part of a monitor, the same part was measured in the case of evaluation of colour reproduction. In order to set the measuring head in the right position a specially prepared image was used (Fig. 2c).

The evaluation of reproduction quality was based on measurements over 100 colour samples. In order to automate these measurements the CalMAN ColorChecker v. 5.4.0.1833 software from SpectraCal was used.

For each tested monitor such measurement process was repeated four times—one time for each calibrator (i.e. Spyder, i1 Display and i1Pro) and one time for the monitor's sRGB preset mode. In case of Dell U2412M there is no sRGB preset mode, then a custom colour mode for 6500 K was used. Finally this measurement schedule gives a collection of 16 sets of measured data.

Because of big differences in monitor default luminance values when the sRGB modes of the monitors are used, the luminance of all monitors was set to 120 cd/m^2, according to the assumptions presented in Sect. 2. For luminance setting the Konica Minolta CA-310 was used. Such luminance correction was not conducted for measurements made after monitor calibration because the luminance setting is executed during calibration by each tested calibration system.

The calibration process requires user action, which usually is reduced to installation of a calibrator on a monitor screen and setting appropriate display luminance and white point (in most cases this was achieved by changing the gain in channels R, G and B) values based on measurements given by the used calibrator. In the case of the X-Rite calibrators for some of the tested monitors the Automatic Display Calibration (ADC) system was used—it controls monitor settings using VESA DDC standard. When the ADC system is used the calibration procedure, except calibrator installation, is executed without user intervention. Settings used for the monitor calibration are presented in Table 2.

4 Measurement Results and Their Analysis

For the evaluation of colour reproduction quality the ΔE^*_{00} metric was used [3]. It describes the difference between two colours in the CIE Lab space in a manner similar to the way in which a human with normal colour vision perceives it.

Table 3 presents the aggregated results of this evaluation. In each cell, from top to bottom, the following estimators of colour reproduction error are shown:

$$\Delta \check{E}^*_{00} = \min_{i=1,\ldots,i_{max}} \left(\Delta E^*_{00}(i)\right), \tag{1}$$

$$\Delta \tilde{E}^*_{00} = \text{median}\left(\{\Delta E^*_{00}\}\right), \tag{2}$$

$$\Delta \bar{E}^*_{00} = \frac{1}{i_{max}} \sum_{i=1}^{i_{max}} \Delta E^*_{00}(i), \tag{3}$$

$$\Delta \hat{E}^*_{00} = \max_{i=1,\ldots,i_{max}} \left(\Delta E^*_{00}(i)\right), \tag{4}$$

where $\{\Delta E^*_{00}\} = \{\Delta E^*_{00}(1), \ldots, \Delta E^*_{00}(i), \ldots, \Delta E^*_{00}(i_{max})\}$, and $\Delta E^*_{00}(i)$ is the difference for *i-th* colour of the tested set of all i_{max} colours, and shows difference between *i-th* colour defined in the PCS (a target colour) and the colour reproduced on the monitor (a measured i.e. real colour).

In Table 3, independently for each tested monitor, the numbers in parentheses show a relative quality (from best to worst) of the tested calibration method in the sense of different error estimator values (1)–(4). Hence this ranking helps to determine the best calibration method for each of the tested monitor. Additionally boldfaced values represent best calibration results, in the sense of (1)–(4), obtained for each calibration method. Moreover cells with gray background represent monitor-calibration set-ups which do not fulfill the following conditions:

$$\Delta \bar{E}^*_{00} \leq 3, \text{ and } \Delta \hat{E}^*_{00} \leq 5, \tag{5}$$

Table 3 Comparison of calibration errors for tested monitor-calibration set-ups

	BenQ XL2420T	Dell Ultrasharp U2412M	Dell Ultrasharp U2410	NEC SpectraView 241
Monitor default sRGB set-up	1.72 (4)	0.63 (4)	0.65 (2)	**0.33** (2)
	5.28 (4)	3.48 (3)	2.15 (3)	**1.15** (2)
	6.19 (4)	3.38 (3)	2.23 (3)	**1.18** (1)
	15.06 (4)	5.08 (3)	3.64 (1)	**2.52** (1)
Datacolor Spyder4ELITE	1.08 (3)	0.63 (3)	0.56 (2)	**0.53** (4)
	6.38 (4)	4.14 (4)	2.77 (4)	**1.90** (4)
	6.11 (3)	3.92 (4)	2.80 (4)	**2.01** (4)
	11.85 (3)	8.60 (4)	5.00 (4)	**4.77** (4)
X-Rite i1 Display Pro	0.45 (1)	0.26 (2)	0.40 (1)	**0.16** (1)
	2.12 (1)	1.36 (1)	1.43 (1)	**1.10** (1)
	2.15 (1)	1.40 (1)	1.60 (1)	**1.26** (2)
	4.27 (1)	3.47 (2)	4.24 (3)	**2.83** (2)
X-Rite i1Photo Pro 2	0.83 (2)	**0.25** (1)	0.69 (3)	0.51 (3)
	2.40 (2)	1.48 (2)	**1.44** (2)	1.49 (3)
	2.46 (2)	1.50 (2)	1.60 (1)	**1.48** (3)
	4.60 (2)	3.46 (1)	4.13 (2)	**2.96** (3)

In each cell are presented, from top to bottom: $\Delta\check{E}^*_{00}$, $\Delta\tilde{E}^*_{00}$, $\Delta\bar{E}^*_{00}$, $\Delta\hat{E}^*_{00}$ values; information about data highlighting is contained in Sect. 4

which in some of the used programs are accepted as conditions for a correct calibration. Therefore in this paper these conditions have been adopted as the necessary criteria of a good calibration.

Table 4 summarizes, for each tested monitor-calibration set-up, its measured values of the black point L_{black} and the white point L_{white} luminance, and the static contrast ($C = \frac{L_{white}}{L_{black}}$). The best values of L_{black}, L_{white} and C for each monitor are boldfaced, and these values fulfill the following conditions:

$$L^{best}_{black} = \min_{\forall L_{black}} (L_{black}) , \tag{6}$$

$$L^{best}_{white} = \min_{\forall L_{white}} \left(\left| L_{white} - 120\,\mathrm{cd/m^2} \right| \right) , \tag{7}$$

$$C^{best} = \max_{\forall C} (C) . \tag{8}$$

Figure 3 shows the graph of the measured monitor static contrast C, and Fig. 4 presents the graph of correlated colour temperature of monitors (CCT). The criterion for best white point CCT value can be defined as:

Table 4 Measured values of black luminance L_{black}

	BenQ XL2420T	Dell Ultrasharp U2412M	Dell Ultrasharp U2410	NEC SpectraView 241
Monitor default sRGB set-up	**0.12**	**0.09**	**0.13**	**0.16**
	124.04	119.41	115.16	115.86
	1032.8	**1385.7**	**868.9**	711.4
Datacolor Spyder4ELITE	**0.12**	**0.09**	0.19	**0.16**
	118.00	115.00	115.26	113.01
	1004.9	1283.0	616.8	706.7
X-Rite i1 Display Pro	**0.12**	0.10	0.17	0.17
	118.32	118.93	107.67	**119.54**
	999.3	1159.3	640.6	692.3
X-Rite i1Photo Pro 2	0.14	0.12	0.22	0.18
	126.85	**127.03**	**128.07**	130.54
	910.8	1027.9	577.7	**740.9**

White luminance L_{white} (both values in cd/m^2) and static contrast C for tested monitor-calibration set-ups

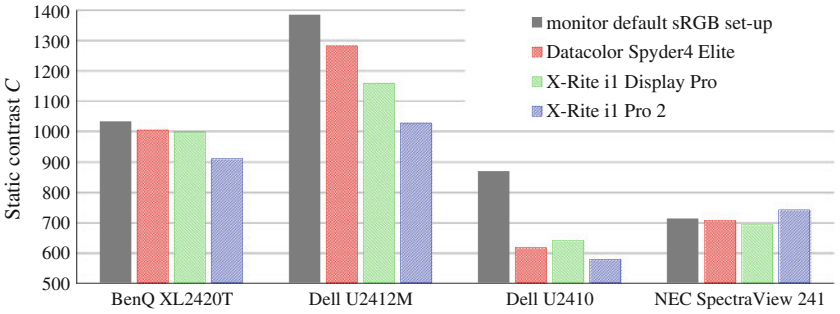


Fig. 3 Measured static contrast C of the monitors

$$CCT^{best} = \min_{\forall CCT} \left(|CCT - 6500\,\mathrm{K}| \right). \tag{9}$$

5 Final Conclusions

An analysis of the results, especially the ΔE^*_{00} values (Table 3), shows that the worst results have been obtained using the Spyder4ELITE system. For three monitors this system did not provide good calibration in the sense of (5). Furthermore this system

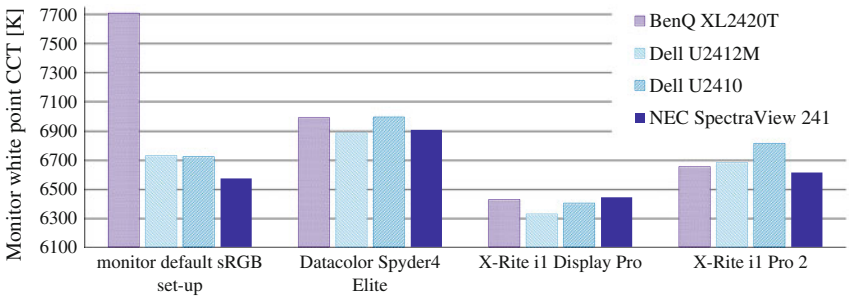


Fig. 4 Measured Correlated Colour Temperature (CCT) of monitors

gives the largest errors of white point CCT setting between all tested systems (Fig. 4). Much better results were obtained using the X-Rite systems, and in particular using the i1 Display Pro. This system for almost every monitor and almost every error estimator gave the best or the second result (Table 3, the values in parentheses). While taking into account only $\Delta\bar{E}^*_{00}$ values the i1 Display Pro system for all monitors gave the best results. For each monitor this system also gave best CCT values in the sense of formula (9) (Fig. 4).

Noteworthy is the performance of the NEC monitor. Comparing to other tested monitors this monitor gave best results in the sense of $\Delta\bar{E}^*_{00}$ and $\Delta\hat{E}^*_{00}$, and its factory sRGB profile is the best profile in the same sense. The NEC monitor also had the best consistency of static contrast (Fig. 3) and for each calibrator it gave the best or near the best CCT value in the sense of (9) (Fig. 4).

This study shows that the quality of monitor colour reproduction strongly depends on a calibration system which has been used for monitor calibration. The usage of a good calibration system (e.g. the X-Rite i1 Diplay Pro) allows to obtain good calibration results (in the sense of the used criteria) even for a monitor which has not been designed for graphics works (e.g. the Benq XL2420T). But for colour critical application the use of a monitor specially designed for graphic works (in the case of this paper the NEC SpectraView 241) is necessary.

Acknowledgments This research was financed from founds for statutory activities of the Institute of Automatic Control of the Silesian University of Technology. The presented experiments were performed in the Laboratory of Imaging and Radiometric Measurements in the Institute of Automatic Control of the Silesian University of Technology.

References

1. Dams, F.E.M., Leung, K.Y.E., van der Valk, P.H.M., Kock, M.C.J.M., Bosman, J., Niehof, S.P.: Technical and radiological image quality comparison of different liquid crystal displays for radiology. Med. Devices: Evid. Res. **7**, 371–377 (2014)

2. Fetterly, K.A., Blume, H.R., Flynn, M.J., Samei, E.: Introduction to grayscale calibration and related aspects of medical imaging grade liquid crystal displays. J. Digit. Imaging **21**(2), 193–207 (2008)
3. Hunt, R.W.G.: The reproduction of colour, 6th edn. Wiley, Chichester (2004)
4. ISO: Image technology color management–Architecture, profile format and data structure–Part 1: Based on ICC. 1, 2004–10, ISO 15076–1:2005
5. ISO: Image technology colour management–Architecture, profile format and data structure–Part 1: Based on ICC.1:2010, ISO 15076–1:2010
6. ISO: Multimedia systems and equipment–Colour measurement and management–Part 2–1: Colour management–Default RGB colour space–sRGB, IEC 61966-2-1
7. Krupinski, E.A., Silverstein, L.D., Hashmi, S.F., Graham, A.R., Weinstein, R.S., Roehrig, H.: Observer performance using virtual pathology slides: impact of LCD color reproduction accuracy. J. Digit. Imaging **25**(6), 738–743 (2012)
8. Liu, S., Ruan, Q., Li, X.: The color calibration across multi-projector display. J. Signal Inf. Process. **2**(2), 53–58 (2011)
9. Lowe, J.M., Brennan, P.C., Evanoff, M.G., McEntee, F.: Variations in performance of LCDs are still evident after DICOM gray-scale standard display calibration. Am. J. Roentgenol. **195**(1), 181–187 (2010)
10. MacDonald, L. (ed.): Digital heritage: applying digital imaging to cultural heritage. Elsevier, Amsterdam (2006)
11. Stokes, M.: Windows platform design notes, designing hardware forthe Microsoft®Windows®family of operating systems, sRGB color management case studies. Technical Report, Microsoft (2001)
12. Stokes, M., Anderson, M., Chandrasekar, S., Motta, R.: A standard default color space for the Internet–sRGB. Technical Report, W3C (1996)

Optical Flow Methods Comparison for Video FPS Increase

Jan Garus and Tomasz Gąciarz

Abstract This paper contains a comparison of two different optical flow methods, which were employed in a simple procedure for increasing the FPS of video sequences. This routine produces additional frames of the video using a computed dense flow's vector field between each pair of consecutive images of the original sequence. The proposed algorithm has a variety of applications, especially for improving a poor quality videos. Described research consists of a definition of general procedure and an attempt to select the best optical flow algorithm (and its parameters). Selected techniques were evaluated using both synthetic and real-world videos. The main intention of the research was to investigate how accurate (and time-consuming) an optical flow engine has to be in order to produce output videos of satisfying quality level.

Keywords FPS increase · Optical flow · Video processing · Image processing

1 Introduction

Quality of motion picture is determined by two main factors: image resolution and FPS (frame per second) number. While increasing the frame dimensions seems to be well described in a number of publications, the second aspect is quite a novel idea.

Our intention is to *generate additional frames* of video sequence using information gathered from existing ones. Such a procedure would double the FPS, producing a noticeably smoother motion picture. There are already some such systems, both commercial and opensource. One of the most known is *slowmoVideo* [3]. It uses information obtained from Optical Flow algorithm to find out where pixels move in the video, and then uses this information to calculate the additional frames.

J. Garus (✉) · T. Gąciarz
Faculty of Physics, Mathematics and Computer Science, Institute of Teleinformatics, Cracow University of Technology, Cracow, Poland
e-mail: jangarus@gmail.com

T. Gąciarz
e-mail: tga@pk.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_29

It seems to be a very attractive idea for enhancing the quality of any video, especially poor quality records taken from CCTV, which FPS number is usually much below the acceptable level of 25. Of course, there are plenty of requirements, which are essential for real-world applications. In particular, we have to obtain sharp object segmentation and precise movement (velocity and direction) estimation, for unrestricted kinds of motion, which are great challenges themselves.

2 Materials and Methods

2.1 Definition of an Optical Flow

According to [9], an *optical (optic) flow* is the most general version of motion estimation, which consists of the computation of an independent estimate of motion at *each* pixel.

Let us assume that we have two images I_{t_1} and I_{t_1+1} belonging to some sequence $I := \{I_1, I_2, \ldots, I_t\}$. An optical flow's goal is to find such vector field

$$w : I \to \mathbb{R}^3 \tag{1}$$

$$w_{(x,y,t)} := (u_{(x,y,t)}, v_{(x,y,t)})^T \quad \forall_{(x,y,t)} \tag{2}$$

that pixel at coordinates (x, y) on the first image I_{t_1} corresponds with the pixel at the coordinates $(x + u_{(x,y,t_1)}, y + v_{(x,y,t_1)})$ on the second image I_{t_1+1}.

In this paper we use the following abbreviations:

$$\boldsymbol{x} := (x, y)^T$$
$$\boldsymbol{w}_{\boldsymbol{x},t} := (u_{(\boldsymbol{x},t)}, v_{(\boldsymbol{x},t)})^T$$

Sand and Teller [8] have emphasized that this problem formulation is very similar to *feature tracking*, except its of a much higher density and short-term scope (usually only two consecutive frames). It is also important to notice, that optical flow does not estimate an object motion. For this reason it suffers similar visual illusions like people, which in fact can be an advantage in some applications.

2.2 Main Algorithms

There are three main families of optical flow methods: local frame-searching *intuitive* (aka '*correlative*') algorithms, *frequency-domain* techniques and *gradient-based* approaches [9].

Correlative methods—due to their simplicity—produce rather poor quality results and require high computational complexity. They are designed mostly for the translational motions. Also the frequency-domain approach has a limitation in the type of supported motion. Even more important the disadvantages of this technique are an inability to deal with large displacements and a model, which concerns only global (meaning whole-frame) changes. These characteristics are unacceptable for enhancing videos in general.

Far better results can be achieved with gradient-based algorithms. In fact this is a state-of-the-art approach. Those methods are based on the *grey value constancy assumption* (pixel's color won't change due to its displacement) and involve other method-specific constraints, like *flow field smoothness* or *gradient constancy assumption*. Most widely known techniques where invented by: [4, 6] (this one is used in [3]), [2, 10]. Another interesting approach, which combines the best parts of the above methods, was presented by [5].

2.3 Proposed Procedure Schema

Our FPS increase algorithm consists of a few key steps:

1: $n \leftarrow 1$
2: read frame I_n
3: **repeat**
4: $n \leftarrow n + 1$
5: read frame I_n
6: $D \leftarrow$ find differences between frames I_{n-1} and I_n
7: $I_{(n-1,n)} \leftarrow$ generate new middle frame using D and I_n
8: output frame I_{n-1}
9: output frame $I_{(n-1,n)}$
10: **until** exist more frames
11: output frame I_n

2.4 Optical Flow Approach

Choice Justification. Our approach involves the use of *optical flow* algorithms in order to find differences between each pair of consecutive frames. This idea came directly from the optical flow's formulation, as it concerns determining a displacement vector which starts at the pixel coordinates in the first image and ends at the location of this pixel in the second one. This is exactly what we need to generate in an additional video's frame.

Algorithm Selection. There are a few key requirements for an optical flow method to be employed as our engine. First if all it has to be as accurate as possible, in

terms of direction and velocity exactness. Subpixel precision is certainly needed. We also require that any kind of motion must be properly modeled, including non-whole-frame translations, rotations, shearing, natural rigid motion etc. Moreover, sharp flow segmentation and high consistency within the moving area are crucial. We put effort rather on quality than on high efficiency, so the algorithm's speed is not an important factor.

Putting it all together we decided to examine two gradient-based algorithms: hierarchical [2] (with the most modifications proposed by [8]) and the fastest [5] method.

2.5 Warping Subprocedure

By *warping* we mean the process of generating an additional frame $I_{(n-1,n)}$ using only the second of the source images I_n and the displacement information expressed as the flow field, denoted by D. (We could use I_{n-1} instead of I_n—it is only our convention).

In the ideal world we would simply set the color of the pixel which lies in the middle of the segment designated by the flow vector $\boldsymbol{w}_{\boldsymbol{x},n-1}$ with the value which we find in the original image I_n at the end of that segment:

$$I_{(n-1,n)}(\boldsymbol{x} + \boldsymbol{w}_{\boldsymbol{x},n-1}\xi_{motion}) = I_n(\boldsymbol{x} + \boldsymbol{w}_{\boldsymbol{x},n-1}) \quad \forall \boldsymbol{x} \tag{3}$$

$$\xi_{motion} = 0.5 \tag{4}$$

Unfortunately, the above formula cannot be used directly due to different domains: $\boldsymbol{x} \in \mathbb{ZZ}^2$ and $\boldsymbol{w}_{\boldsymbol{x},n-1} \in \mathbb{R}^2$. To solve this problem, a bilinear interpolation has been involved on the right side of Eq. 3 in order to determine the pixel color.

Another difficulty is modifying the target frame at the non-integer coordinates (see the left side of Eq. 3). Some interpolation technique (based on the bilinear method) has been implemented as well, but we decide to rather modify Eq. 3 than to complicate our routine. Instead of setting the pixel color in the middle of the segment designated by the flow vector, we modify pixel color at the beginning of this segment with the value at the location of it's middle point:

$$\begin{aligned} I_{(n-1,n)}(\boldsymbol{x}) &= I_n(\boldsymbol{x} + \boldsymbol{w}_{\boldsymbol{x},n-1}\xi_{motion}) \\ &= B(I_n(\boldsymbol{x} + \boldsymbol{w}_{\boldsymbol{x},n-1}\xi_{motion})) \quad \forall \boldsymbol{x} \end{aligned} \tag{5}$$

where $B(\ldots)$ denotes bilinear interpolation. $I_{(n-1,n)}$ is initially filled with I_n. Quite surprisingly, the proposed simplification doesn't have a significant impact on achieved results, but it speeds up warping execution time.

Table 1 Characteristics of the test sequences

Name	Dimensions	Colors	Origin	Motion type	Shift
Yos Clouds	316 × 252 px	GS	Synthetic	Translational	Small
Ettlinger Tor	512 × 512 px	GS	Real	Translational + curved	Small
Text	49 × 57 px	GS	Real	Globally translational	Tiny
Geometry	800 × 600 px	RGB	Synthetic	Translational, rotation	Large
Windmill1	640 × 480 px	RGB	Real	Rotation	Large
Windmill2	640 × 480 px	RGB	Real	Rotation + globally translational	Large
Window	800 × 480 px	RGB	Real	Human (natural rigid)	Large

2.6 Test Sequences

The proposed procedure was evaluated on both synthetic and real-world videos (see Table 1). Two of them are commonly used to test and compare motion analysis algorithms: *Yosemite Clouds* [1], and *Ettlinger Tor* [7]. The other sequences have been created by authors of this paper.

3 Results

3.1 Algorithms' Comparison

This test's goal was to find out, which optical flow algorithm produces more accurate results. For this purpose we set $\xi_{motion} := 1$, so the theoretically resulting image $I_{(1,2)}$ will be identical to I_1. The procedure was quite simple:

1. Compute flow D between I_1 and I_2
2. Generate I_1's estimate $\hat{I}_1 := I_{(1,2)}$ using I_2 and D
3. Compare $\hat{I}_1$ and I_1 using *SSIM* (*Structural SIMilarity index* [11]).

Both optical flow techniques were executed with different settings:

Norazlin A Number of iterations $k_{max} = 2$.
Norazlin B Number of iterations dependent on image dimensions.
Brox A Image dimensions reduction factor $\eta = 0.8$ (resolution hierarchy), 1 outer and 100 inner iterations.
Brox B Factor $\eta = 0.8$, 3 outer and 100 inner iterations.

Brox C Factor $\eta = 0.8$, 3 outer and 150 inner iterations.
Brox D Factor $\eta = 0.9$, 3 outer and 150 inner iterations.

The results are listed in Table 2. Please note, that our similarity metric is sensitive to such factors like an area of frame, that has changed in time, so the results shouldn't be compared between test sequences. Execution times are presented in Table 3. A visual comparison is shown in Figs. 2, 3 and 4. Optical flow visualization legend is explained in Fig. 1.

Table 2 Structural SIMilarity index $SSIM(I_1, \hat{I}_1)$ in comparison test of Norazlin et al. (No.) and Brox et al. (Br.) methods

	Yos Clouds	Ettlinger Tor	Text	Geom.	Windm.1	Windm.2	Window
No. A	0.5828	0.8966	0.7437	0.8571	0.9163	0.8067	0.9752
No. B	0.4438	0.8547	0.6945	0.8559	0.8819	0.6865	0.9522
Br. A	0.8986	0.9699	0.9314	0.9010	0.9504	0.9032	0.9918
Br. B	0.9334	0.9757	0.9196	0.9168	0.9620	0.9323	0.9935
Br. C	0.9334	0.9757	0.9195	0.9168	0.9620	0.9323	0.9935
Br. D	0.9507	0.9783	0.9139	0.9608	0.9734	0.9468	0.9947

Table 3 Execution times [s] in the comparison test of Norazlin et al. (No.) and Brox et al. (Br.) methods. C++ implementation executed on 1.83 GHz machine

	Yos Clouds	Ettlinger Tor	Text	Geom.	Windm.1	Windm.2	Window
No. A	0.4092	1.1095	0.0645	1.7990	1.1469	1.2648	1.4654
No. B	0.4264	1.2952	0.0661	2.0083	1.3531	1.5342	1.7446
Br. A	5.0545	25.3900	0.1402	47.5238	25.1880	25.2643	33.9617
Br. B	10.3943	43.1543	0.2791	81.1497	46.2393	46.3706	60.4924
Br. C	11.4496	48.9324	0.2539	102.8170	56.3145	60.2515	66.1096
Br. D	26.2761	110.4640	0.6527	236.2300	130.7090	140.0080	157.252

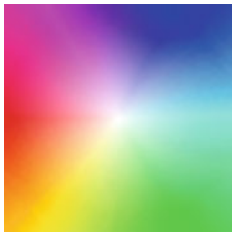


Fig. 1 The optical flow visualization legend. Saturation indicates displacement magnitude, the hue shows the direction

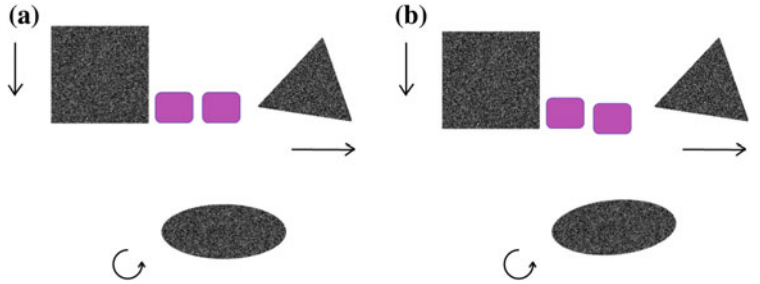


Fig. 2 Original images I_1 (**a**) and I_2 (**b**) from sequence *geometry*

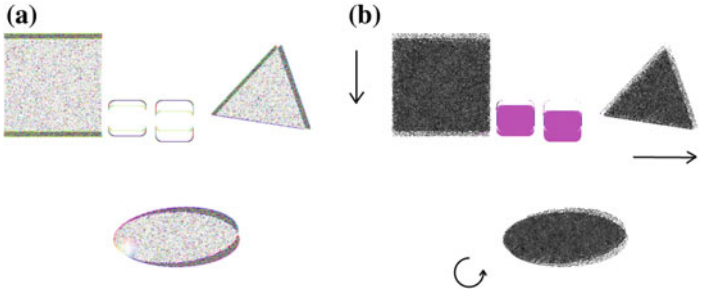


Fig. 3 Ibrahim et al. [5] algorithm (variant A), sequence *geometry*: **a** optical flow field, **b** output $\hat{I}_1$

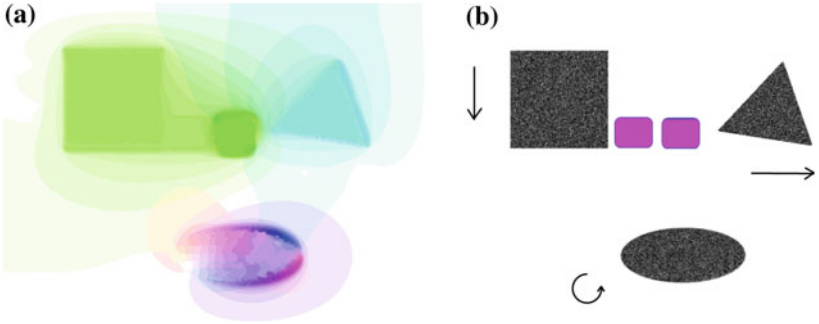


Fig. 4 Brox et al. [2] algorithm (variant B), sequence *geometry*: **a** optical flow field, **b** output $\hat{I}_1$

3.2 FPS Increase

In order to verify the accuracy of our solution, we ran another test in which we tried to restore the removed frame of the video and then compare it with the original one. Although this kind of test has some limitations (results are most reliable for uniform linear motion), it gives us a general idea about estimation precision.

Test procedure for each image sequence $\{I_1, I_2, I_3\}$ consists of the following steps:

1. Cut out middle frame I_2
2. Compute flow D between I_1 and I_3
3. Generate I_2's estimate $\hat{I}_2 := I_{(1,3)}$ using I_3 and D
4. Compare $\hat{I}_2$ and I_2 using *SSIM* (*Structural SIMilarity index* [11]).

The above steps where repeated for a few different ξ_{motion} values. Only the Brox et al. method was examined, with $\eta = 0.8$, 3 outer and 100 inner iterations. The results are listed in Table 4, selected generated frames can be found in Figs. 5, 6, 7 and 8.

Table 4 Structural SIMilarity index $SSIM(I_2, \hat{I}_2)$ in FPS increase test—[2] optical flow method

ξ_{motion}	Yos Clouds	Ettlinger Tor	Text	Geom.	Windm.1	Windm.2	Window
0.4	0.8860	0.9655	0.7924	0.8380	0.8853	0.8946	0.9846
0.5	0.9266	0.9694	0.7976	0.8521	0.8919	0.9201	0.9885
0.6	0.9433	0.9713	0.8011	0.9294	0.9004	0.9263	0.9907
0.65	0.9404	0.9715	0.8025	0.9275	0.9048	0.9217	0.9909
0.7	0.9300	0.9712	0.8033	0.8974	0.9082	0.9126	0.9904

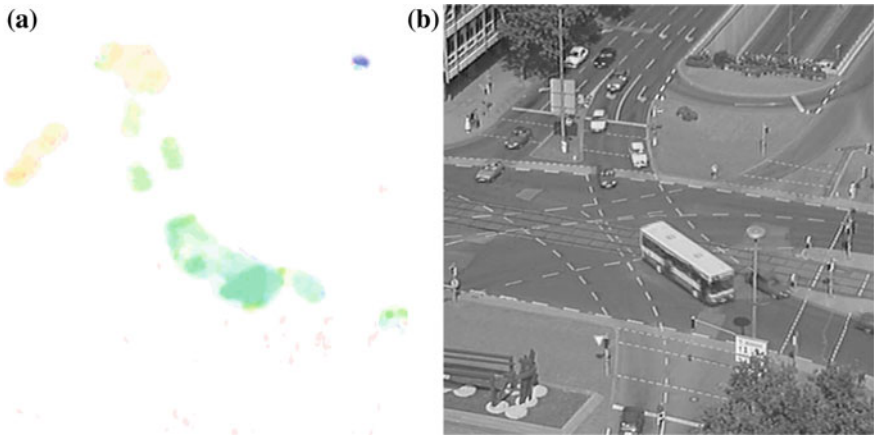


Fig. 5 FPS increase test—sequence *ettlinger tor*, method [2] (variant B), $\xi_{motion} = 0.6$: **a** optical flow field, **b** output frame $\hat{I}_2$

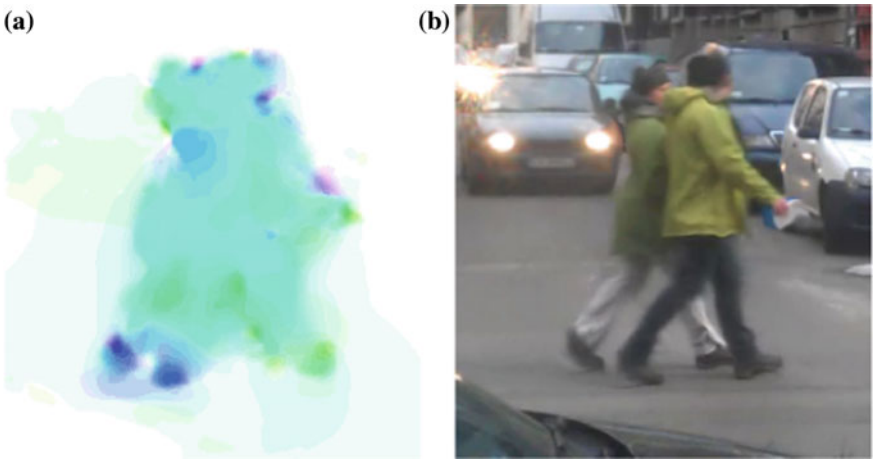


Fig. 6 FPS increase test—sequence *window* (fragment), method [2] (variant B), $\xi_{motion} = 0.6$: **a** optical flow field, **b** output frame $\hat{I}_2$

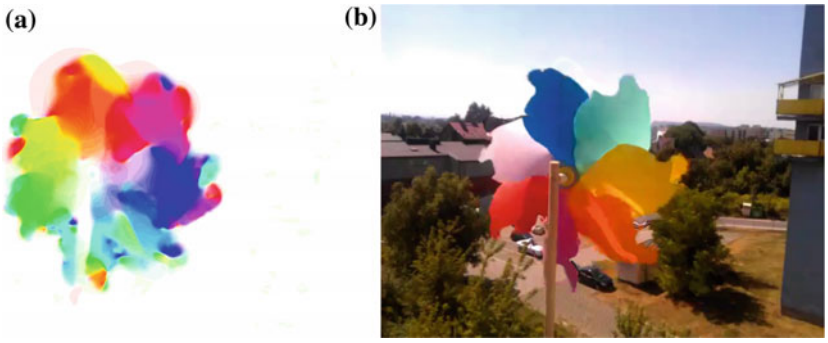


Fig. 7 FPS increase test—sequence *windmill1*, method [2] (variant B), $\xi_{motion} = 0.6$: **a** optical flow field, **b** output frame $\hat{I}_2$

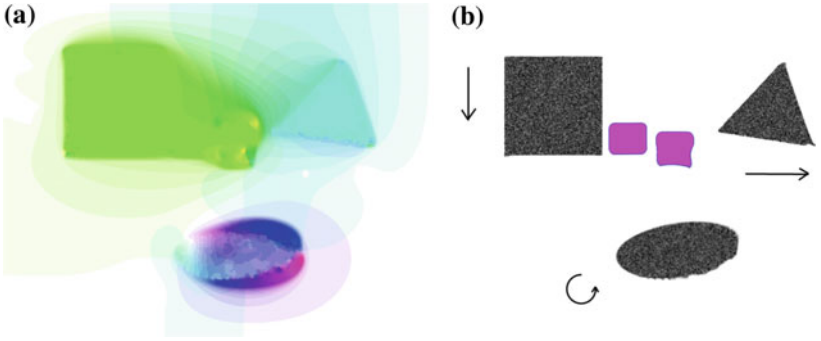


Fig. 8 FPS increase test—sequence *geometry*, method [2] (variant B), $\xi_{motion} = 0.6$: **a** optical flow field, **b** output frame $\hat{I}_2$

4 Discussion

According to both the visual and numerical results, the Brox et al. method is far better than Norazlin et al., which fails wherever color gradient vanishes. It produces an optical flow field with no smoothness or consistency, what completely disqualifies the usage of this algorithm in the considered application. On the other hand, the Norazlin et al. technique may be very useful in other problems, where precision and smoothness aren't so important, but almost real time efficiency is required.

Our FPS increase procedure produces quite satisfying results in the case of translational motion, but rotations are supported much worse. Another problem is a lack of sharp edges and some visible interfering between those frame areas, which move independently. To extent this can be fixed with proper global and local smoothness parameters, but it probably cannot be totally eliminated in the Brox et al. method.

Further Research and Applications. Apart from the issues listed in the previous subsection, some other improvements may be considered. One of the most important is the warping subprocedure, which may be responsible for the significant worsening of the final results. Especially, the value of the optimal ξ_{motion} which doesn't equal 0.5 is quite suspicious. Some attempts were already made as described earlier, but no relevant improvement was achieved so far.

Increasing FPS has a number of applications, like improving quality of low frame rate records, generating smooth slow motion or interpolation between multiple cameras shooting the same scene from different angles. An investigation of the optical flow algorithms has an important practical aspect in the context with some applications where better image quality is required and the features set coming from optical flow serves for features vector construction used for objects classification. As an example of such application we deal with, is our novel, biometric technique of hand movement analyze. Increasing FPS in this case results in cheaper hardware, lower communication bound, and higher detection rate.

Another attractive idea is to combine our procedure with *super resolution techniques* in order to increase both FPS and image dimensions. We will certainly investigate this idea in the future. Authors plan to use the optical flow and video quality enhancement methods to build fast document scanner, based on high resolution video (for books scanning). Each page scan is built up from several, consecutive video frames recorded during browsing a book. Enhanced resolution significantly improves image binarization and OCR results.

References

1. Bayerl, P.: Yosemite Clouds. http://www.informatik.uni-ulm.de/ni/staff/PBayerl/homepage/animations
2. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: Computer Vision—ECCV 2004, LNCS, vol. 3024, pp. 25–36. Springer, Heidelberg (2004)

3. Eugster, S.: slomoVideo: slow-motion for video with optical flow. Bachelor thesis (2011). http://slowmovideo.granjow.net/
4. Horn, B., Schuck, B.: Determining optical flow. Artif. Intell. **9**, 185–203 (1981)
5. Ibrahim, N., Wan Zaki, W.M.D., Hassan, A., Mustafa, M.M.: Optical flow improvement towards real time and natural rigid motion estimation. In: IEEE ICSIPA 2009, pp. 322–325. Kuala Lumpur, Malaysia (2009)
6. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: IJCAI 1981, pp. 121–130. Vancouver, Canada (1981)
7. Nagel, H.: Ettlinger Tor. http://i21www.ira.uka.de/image_sequences
8. Sand, P., Teller, S.: Particle video: long-range estimation using point trajectories. Int. J. Comput. Vis. **80**(1), 72–91 (2008)
9. Szeliski, R.: Computer Vision: Algorithms and Applications. Springer, London (2010)
10. Tomasi, C., Kanade, T.: Detection and Tracking of Point Features. In: Carnegie Mellon University Technical Report CMU-CS-91-132 (1991)
11. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)

Towards the Automatic Definition of the Objective Function for Model-Based 3D Hand Tracking

Konstantinos Paliouras and Antonis A. Argyros

Abstract Recently, model-based approaches have produced very promising results to the problems of 3D hand tracking. The current state of the art method recovers the 3D position, orientation and 20 DOF articulation of a human hand from markerless visual observations obtained by an RGB-D sensor. Hand pose estimation is formulated as an optimization problem, seeking for the hand model parameters that minimize an objective function that quantifies the discrepancy between the appearance of hand hypotheses and the actual hand observation. The design of such a function is a complicated process that requires a lot of prior experience with the problem. In this paper we automate the definition of the objective function in such optimization problems. First, a set of relevant, candidate image features is computed. Then, given synthetic data sets with ground truth information, regression analysis is used to combine these features in an objective function that seeks to maximize optimization performance. Extensive experiments study the performance of the proposed approach based on various dataset generation strategies and feature selection techniques.

Keywords 3D hand tracking · human motion capture · optimization · regression analysis · PSO

K. Paliouras · A.A. Argyros (✉)
Institute of Computer Science, Foundation for Research and Technology, Heraklion, Greece
e-mail: argyros@ics.forth.gr

K. Paliouras
e-mail: kpal@ics.forth.gr

K. Paliouras · A.A. Argyros
Computer Science Department, University of Crete, Heraklion, Crete, Greece

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_30

1 Introduction

The automatic capture and analysis of human motion has several applications and high scientific interest. Long standing unresolved human-computer interaction problems could be solved by direc tly using the human body and, in particular the human hands for interacting with computers.

Several solutions have been proposed for the problem of 3D hand tracking [6, 11]. These solutions can be divided into two main categories, the *appearance-based* and the *model-based* approaches. The appearance-based approaches try to solve the problem by defining a map between the feature space and the solution space. This map is usually constructed with offline training of a prediction model, which can be either a regression or classifier model. The regression models are used to predict the exact configuration of the hand in the solution space, while the classifier models usually try to predict the posture of the observed hand. In contrast, model-based approaches operate directly on the solution space. Usually this involves making multiple hypotheses in the solution space that are evaluated in feature space by comparing the appearance of the observed hand and the estimated appearance of the hypotheses. This is formulated as an optimization problem whose objective function evaluates hypotheses (candidate hand configurations that are rendered through graphics techniques by taking into account its kinematic model) against the actual hand observations. The objective function has a determining role in the quality of the obtained solutions as well as to the convergence properties of the optimization process. Its formulation requires prior-experience on the problem and a lot of fine tuning that is performed on a trial-and-error basis.

In this paper, our aim is to automate the process of objective function definition with a methodology that does not demand deep prior-knowledge of the problem. More specifically, a set of features that are relevant to the problem are supposed to be given. Then, regression analysis is used to define an objective function that given the available features, approximates as better as possible the true distance between different hand poses. To do so, synthetic datasets are used which are built by sampling the multi-dimensional solution space with two different strategies.

2 Related Work

A lot of published work exists for recovering the full 3D configuration of articulated objects, especially the human body or parts of it like hands, head etc. Moeslund et al. [11] have made an extensive survey on vision-based human body capture and analysis. Hand tracking and body tracking problems share many similarities, like hierarchical tree structure, problem dimensionality and complexity, occlusions and anatomic constraints. Erol et al. [6] present a variety of methods for hand pose estimation or tracking. Depending on the output of these methods they are divided in partial and full pose estimation methods. Further categorization is between

appearance-based and model-based methods. Typically appearance-based methods [14, 15, 17] solve the problem by modelling the mapping between the feature space and the pose space either analytically, or through machine learning techniques that are trained on specific datasets to perform either classification or regression. Appearance-based methods perform fast on prediction and are suitable for gesture recognition. A common problem though, for these methods, is that they usually demand large training dataset which, thus, is typically relevant to a specific application and/or scenario. To compensate for potential bias of the training dataset, Shotton et al. [15] generated a large synthetic dataset, permitting good generalization.

On the other hand, model-based methods [5, 8, 10, 12] search directly for the solution in the configuration space. Each hypothesis is rendered in feature space. An error/objective function evaluates visual discrepancies, which usually demands high computational resources. On the positive side, model-based methods do not need offline training, making them easier to employ as they are not biased to specific training datasets.

In all optimization problems, there are two major components, the error/objective function and the optimization algorithm. Different options exist for both components but the performance is dependent on the correct combination of the two. As an example, de La Gorce et al. [5] used a quasi-newton optimization algorithm and a hand made error function that considers textures and shading of the model which proved to perform well on the problem. Oikonomidis et al. [12] used the *Particle Swarm Optimization* (PSO) algorithm and they also crafted a special objective function, taking into account the skin color and depth information provided by a RGB-D sensor or from a multi-camera setup [13]. Recent works have tried to combine the advantages of both methods basically using machine learning. Xiong et al. [18] mentioned the effectiveness of 2nd order descendent methods, as well as the difficulty in using them in computer vision, mainly because it is hard to analytically differentiate the objective function. They proposed instead, a supervised descent method that models the behaviour of an objective function in offline training. At prediction time, this method consults the learned model to estimate the descent direction, allowing the usage of any non linear objective function. Cao et al. [4] have presented an appearance-based method for tracking facial expression. In their work, they have used regression analysis for modelling the correlation between the feature space and the 3D positions of a face shape model.

3 The Baseline Method

For the purposes of studying and evaluating an alternative method for constructing the objective function of model-based hand-trackers, we used the work of Oikonomidis et al. [13]. In that work, the authors present a model-based approach for recovering and tracking the full configuration of a human hand with 27 DOFs. The problem is formulated as an optimization problem seeking the best solution in the configuration space of 27 dimensions. The methodology can be divided into 3 main components.

- *Observation*: Responsible for acquiring input from the sensor and pre-processing data. At the pre-process stage, it detects and isolates all the areas with skin color [2].
- *Hypothesis evaluation*: For each hypothesis made in configuration space its discrepancy to the observed hand pose must be evaluated. This component is responsible for quantifying this discrepancy by considering the visual discrepancies in feature space. To estimate the appearance of each hypothesis, rendering techniques are used to project the hypothesis in the feature space. To achieve this, the hand shape and its kinematic model are supplied to the rendering pipeline.
- *Optimization*: Performs optimization on solution space in order to find the hypothesis with the minimum distance to the observed hand. The PSO [9] optimization algorithm was chosen for this task and the hypothesis evaluation component is used to score all candidates during the optimization process.

Figure 1 (left) illustrates a high-level flow diagram of the methodology followed in [13]. The objective function defined in [13] receives the configuration of the hypothesized hand h and the visual observation model of the tracked hand O and quantifies their discrepancy as:

$$D(O,h,C) = \frac{\sum min(|o_d - r_d|, d_M)}{\sum(o_s \vee r_m) + \varepsilon} + \\ + \lambda \left(1 - \frac{2\sum(o_s \wedge r_m)}{\sum(o_s \wedge r_m) + \sum(o_s \vee r_m)}\right) + \lambda_k \cdot kc(h). \quad (1)$$

The first term of Eq. (1) measures the difference between depth maps of an observed and a hypothesized hand pose. The second term of Eq. (1) performs a comparison of the segmented region of the hand with the corresponding map of the hypothesis. To perform the comparison, a mapping of the hypothesis h to feature space is applied by means of rendering. Finally, the third term adds a penalty to kinematically implausible hand configurations by penalizing adjacent finger interpenetration. For more details in that approach the reader is referred to [13].

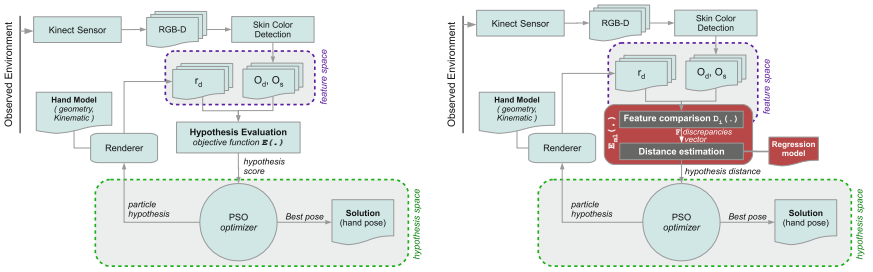


Fig. 1 Overview of baseline method pipeline (*left*) and the proposed modification (*right*) In the proposed work, the objective function is not handcrafter but rather estimated through an offline trained regression model

4 Methodology

In this work, various simple features are computed to create a vector of scalar feature discrepancies. Then, by using *regression methods*, the correlation between the vector of feature discrepancies and the *true distance* is modeled and a learned function $E_{ml}(.)$ is defined to replace *baseline objective function* $E(h, O)$ (Eq. 1). Figure 1 (right) shows how the new $E_{ml}(.)$ function is integrated in the *baseline method*. The proposed method consists of three-steps, (a) create a set of algorithms to calculate per feature discrepancy, quantified in a scalar variable (Sect. 4.1), (b) construct a dataset that will be used to train and evaluate the performance of various objective functions (Sect. 4.2) and, (c) train a machine-learned function using the dataset from the previous step. This function, will form the new objective function E_{ml} (Sect. 4.3).

4.1 Features and their Comparison

We consider a number of features as well as functions $D_i(O, h)$ that measure the discrepancy of each of the feature between the observation O and a hypothesis h. In the following, o_d is the depth map of the observation, o_s is the skin map segmented using skin color detection, and r_d is the rendered depth map of hypothesis h. In all cases N is the total number of pixels of the feature maps.

Sum of Depth Differences $\mathbf{D}_1(.)$ Depth discrepancy is very informative of the correlation between two poses. Unlike the baseline method, we define it without any data type of post-processing (e.g., clamping etc.):

$$D_1(O, h) = \sum |o_d - r_d|. \tag{2}$$

Variance of Depth Distances $\mathbf{D}_2(.)$ This provides another statistical perspective of the depth discrepancy and is defined as:

$$D_2(O, h) = \frac{1}{N}\sqrt{\sum (o_d - r_d)^2}. \tag{3}$$

Occupied Area $\mathbf{D}_3(.)$ This is defined as the difference of the areas (in pixels) covered by the segmented hand observation and hypothesis The area is calculated based on the corresponding depth maps as the number of non zero-valued pixels:

$$D_3(O, h) = \left| \sum_{pixel(o_d)\neq 0} 1 - \sum_{pixel(r_d)\neq 0} 1 \right|. \tag{4}$$

Accuracy of the Skin Map $\mathbf{D}_4(.)$ Measures the compatibility between the observed skin color map and that of the hand hypothesis. This is quantified as the $F1$ measure between the two maps as:

$$D_4(O, h) = 2TP/(2TP + FP + FN), \tag{5}$$

where $TP = \sum o_d \wedge r_d$, $TN = \sum \neg o_d \wedge \neg r_d$ and $FN = \sum o_d \wedge \neg r_d$.

Depth Map Edges $\mathbf{D}_5(.)$ Edges are computed with the Canny edge detector [3] resulting in the edge maps o_e and r_e for observation and hypothesis, respectively. Then the Euclidean distance map (ℓ_2 distance transformation) o_{ed} is generated over the edge map o_e, using [7]. A scalar value is the computed as:

$$D_5(O, h) = \sum o_{ed} \wedge r_e. \tag{6}$$

Hand Contour $\mathbf{D}_6(.)$ The method used in feature discrepancy $D_5(.)$ is also applied to compare contours rather than skin colored regions. Specifically, the o_s map and the binary mask r_m that corresponds to the occupied pixels of r_d are used to generate the edge maps of the observation and the hypothesis. Thus,

$$D_6(O, h) = \sum o_{sd} \wedge r_d, \tag{7}$$

where o_{sd} is the distance transform of o_s.

4.2 Dataset

Given two hand poses h_α and h_β we define their true distance $\Delta(h_\alpha, h_\beta)$ as

$$\Delta(h_\alpha, h_\beta) = \frac{1}{37} \sum_{i=1}^{37} ||p_i(h_\alpha) - p_i(h_\beta)||. \tag{8}$$

$\Delta(h_\alpha, h_\beta)$ is the averaged Euclidean distance between the centers of the 37 primitive shapes p_i comprising the hand model [13].

A dataset with examples of compared hand poses is needed for modeling the correlation between feature discrepancies and *true distance*. Every example in this dataset consists of the feature discrepancy vector F_i and the *true distance* Δ_i between an observed hand model and a hypothesis. To create a dataset, the search space of the optimization procedure must be sampled appropriately. The size of this space is huge to permit dense sampling. Therefore, two different sampling strategies are introduced. The first one uses a low-discrepancy sequence to select hand poses quasi-randomly. The second samples densely around the area that is mostly used during optimization.

Sampling with Low-Discrepancy Sequence: The advantage of using low-discrepancy sequences is that they offer more uniform sampling of a multidimensional space, even for a few samples. Several such algorithms have been developed. In this paper we use the Sobol sequence [16]. The sampling procedure is performed in the following steps: (a) Create a set P of n quasi-random hand pose configurations (b) Generate the feature maps p_d and p_s (depth, skin map) for every hand pose in P and (c) for all possible combinations of P set, generate an example in the dataset that consists of the the true distance Δ_k and the F_k feature discrepancies vector calculated by the $D_i(.)$ functions. The bounds of each dimension are selected based on the usable frustum of the Kinect sensor and the possible movements of hand joints based on anatomical studies [1]. In particular, for the 27-DOF parameters of the hand pose the boundaries of the global position are selected so that the hand is always inside the perspective frustum. The boundaries of each finger are the same as the boundaries of the PSO optimization module in baseline method. In the special case of global orientation, another quasi-random algorithm is used to create random quaternions.

Sampling Biased to Optimization: The previous method provides a good strategy for uniformly sampling the search space. However, in practice, the optimization module of the baseline method considers solutions close to the actual one. This happens because the particles of PSO are distributed around the previous solution of the previous frame. Therefore, a second sampling strategy is proposed that samples the search space where the *baseline method* usually searches. To do so, we use the logs of previous hand tracked poses and the particles that PSO considered.

4.3 Regression Model

$E_{ml}(.)$ (Eq. 9) is constructed by modelling the *training dataset* that was created using either of the methods described in Sect. 4.2. That is, given the outcomes of $D_i(.)$ functions (Sect. 4.1) we need to come up with a function $E_{ml}(.)$ that approximates as closely as possible the true distance $\Delta(.)$

$$E_{ml} = f(D_1(.), D_2(.), ..., D_n(.)) \tag{9}$$

Different regression analysis algorithms have been developed to find the correlation between parameters on a dataset, depending on the nature of their relation. In our problem formulation, we have employed and experimented with four different models, (a) linear model using mean squared error, (b) polynomial model of 2nd degree, (c) polynomial model of $3rd$ degree and (d) random forests with 6 sub-trees.

5 Experimental Evaluation

The experimental evaluation of the proposed approach was performed based on both synthetic and real data. All experiments ran on a computer equipped with a quad-core intel i7 930 CPU, 16 GBs RAM and the Nvidia GTX 580 GPU with 1581 GFlops processing power and 1.5 GBs memory.

5.1 Evaluation on Synthetic Data

We evaluated the performance of tracking on a synthetic data set that was created using recorded real-life hand motion. Having such a dataset, we evaluate the performance of a tracker by measuring the error between the ground truth and the solution proposed by the tracker using the $\Delta(.)$ function in Eq. (8).

Regression Models Performance: We first evaluate the performance of each feature $D_i(.)$ depending on the employed regression model. For this test we trained the four proposed models (Sect. 4.3) using only one feature discrepancy function $D_i(.)$ and using simultaneously all six $D_i(.)$ functions. In all cases the models were trained on the same training dataset and were evaluated on the same ground truth dataset. The PSO was configured to 64 particles and 25 generations.

Figure 2 illustrates the results of this test. Note that the combination of all features produced better results than using any single feature. Using only one feature resulted

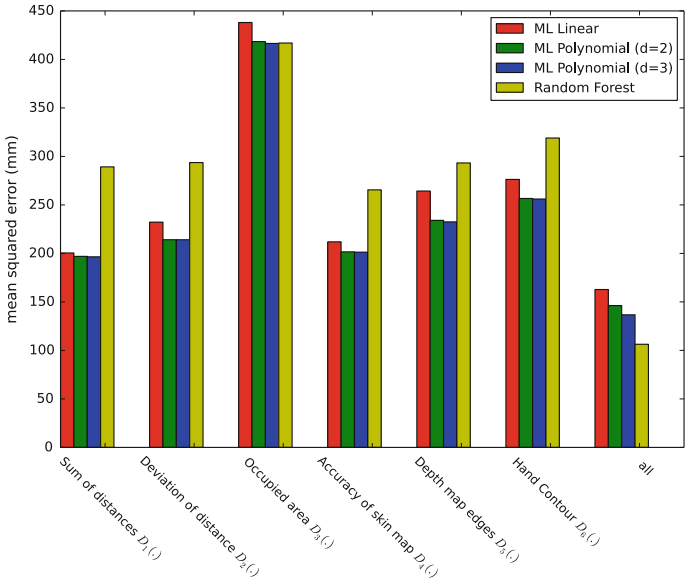


Fig. 2 Performance of models when trained on dataset that contained only one feature, compared to performance when trained on all features simultaneously

in almost the same accuracy for all $D_i(.)$ functions except for $D_3(.)$ which showed reduced performance. It is interesting that the *random forest* algorithm had different performance compared to other models when using one $D_i(.)$ function or all six combined. Specifically, it was significantly outperformed by all other closed form algorithms when using only one feature. Still, it had better performance than all other algorithms when using all $D_i(.)$ functions together.

Tracking Performance: We evaluated different configurations for the regression model, the training dataset and the PSO algorithm. More specifically with respect to the regression model we considered (a) a linear model (b) 2nd degree polynomial and (c) random forests. For the training dataset we employed one with 4096 poses using quasi-random sampling and a second one generated on previous tracking logs. PSO ran with 5, 10, 15, 20, 25 generations and 8, 32, 64 particles. Due to the stochastic nature of PSO, for each configuration, the test was run 20 times and the mean error was considered. The error was measured in *mm* using the $\Delta(.)$ function Eq. 8.

Figure 3 (left) shows tracking accuracy for the training dataset generated by *quasi-random* sampling. The baseline method has 10.7 mm minimum error which is the best for all cases. However, a simple linear method with no prior knowledge of the problem complexity is only 8 mm worse than the *baseline method*. The polynomial method has an error of 2*S*8.0 mm. The explanation for this is that the polynomial function was over-fitted on the dataset which had more samples at long distances as explained in Sect. 4.2. This led to poor generalization at small distances which are extensively searched by the PSO. Finally, the *Random Forests* algorithm had the worst performance with 100.4 mm minimum error and 1500 mm maximum error. *Random Forests* make no assumption of the modelled space which makes them a bad predictor for areas where no training samples exist.

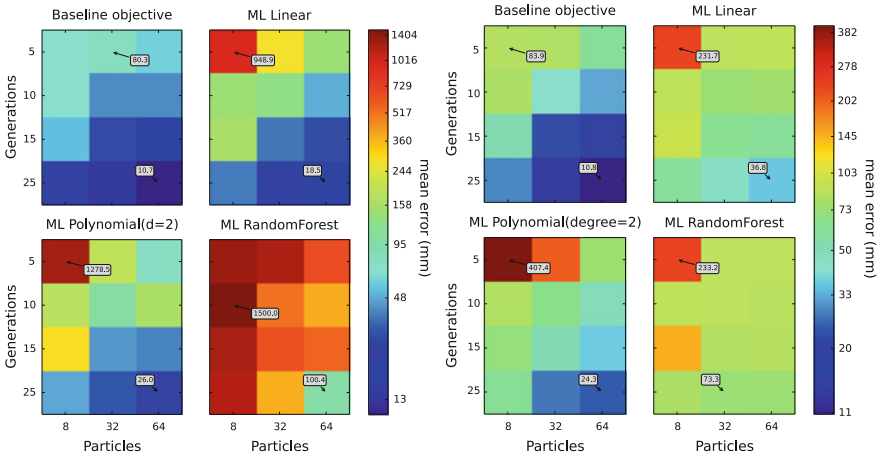


Fig. 3 Tracking performance tested on the ground truth "cooljason" dataset. Models were trained on dataset generated by 4096 quasi-random poses (*left*) and on the tracking logs using the objective function (*right*)

Figure 3 (right) shows the same test but using the training dataset generated from a previous tracking log. The major difference with Fig. 3 is that all regression models have improved performance on low PSO budget. For higher PSO budget configurations, optimization performance varies in relation to the regression model used. More specifically, both the linear and the polynomial regression models perform worse than th previous datasets but now polynomial performs better that the liner model. In contrast, *Random Forests* have improved performance. This is expected as this dataset includes examples of poses that are not that far apart, so *Random Forests* managed to learn the behaviour of the objective function in that area of the pose space.

The results of this test show that the proposed method is sustainable and can, to some degree, replace the manual procedure of designing an objective function. At the same time, it has also been shown that the generation of a training dataset plays a key role on performance and should not be overlooked.

6 Discussion

In this work, we acknowledge the importance of the structure of the objective function on the optimization problem and the difficulties one has to face in order to construct a function that performs well within the problem. The goal of this work was to find a systematic, automated method to construct such an objective function without a deep knowledge of the problem at hand. To apply and evaluate our method, different regression algorithms were tested on two different training dataset generation techniques. We evaluated the influence of several feature comparison functions $D_i(.)$ isolated against each tested regression model. The experiments showed that all $D_i(.)$ functions could approach the solution but the performance of their combination was by far the best. Another interesting result is that the Random Forests algorithm used for regression analysis was more effective in multi-dimensional space than any of the other methods. In the last step of evaluation, we measured the performance of hand-tracking by replacing the baseline objective function with the automatically estimated objective function $E_ml(.)$. For this experiment we tested multiple configurations of the PSO algorithm, regression models and training dataset generation approaches. It has been verified that none of the configuration profiles managed to outperform the baseline objective function but many profiles approached competitive results. This is quite important, especially considering the reduced demands of expertise on the problem of hand tracking. Finally, we should mention that this method is not constrained to the problem of hand-tracking but can be used in any optimization problem that depends on complex objective functions.

Acknowledgments This work was partially supported by the EU project FP7-IP- 288533 Robohow. The contributions of Iason Oikonomidis and Nikolaos Kyriazis, members of FORTH/CVRL, are gratefully acknowledged.

References

1. Albrecht, I., Haber, J., Seidel, H.P.: Construction and animation of anatomically based human hand models. In: SIGGRAPH, pp. 98–109. San Diego, USA (2003)
2. Argyros, A.A., Lourakis, M.I.: Real-time tracking of multiple skin-colored objects with a possibly moving camera. In: Pajdla, T., Matas, J. (eds.) Computer Vision–ECCV 2004. LNCS, vol. 3023, pp. 368–379. Springer, Berlin (2004)
3. Canny, J.: A computational approach to edge detection. IEEE Trans. PAMI-Pattern Anal. Mach. Intell. **8**(6), 679–698 (1986)
4. Cao, C., Weng, Y., Lin, S., Zhou, K.: 3D shape regression for real-time facial animation. ACM Trans. Graphics **32**(4), 41:1–41:10 (2013)
5. de La Gorce, M., Paragios, N., Fleet, D.J.: Model-based hand tracking with texture, shading and self-occlusions. In: CVPR, pp. 1–8. Anchorage, USA (2008)
6. Erol, A., Bebis, G., Nicolescu, M., Boyle, R.D., Twombly, X.: Vision-based hand pose estimation: a review. Comput. Vis. Image Underst. **108**(1), 52–73 (2007)
7. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient belief propagation for early vision. Int. J. Comput. Vision **70**(1), 41–54 (2006)
8. Hamer, H., Schindler, K., Koller-Meier, E., Van Gool, L.: Tracking a hand manipulating an object. In: ICCV, pp. 1475–1482. Kyoto, Japan (2009)
9. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: ICNN, vol. 4, pp. 1942–1948. Perth, Australia (1995)
10. Kyriazis, N., Oikonomidis, I., Argyros, A.: A GPU-powered computational framework for efficient 3D model-based vision. Techincal report, ICS-FORTH (2011)
11. Moeslund, T.B., Hilton, A., Krüger, V.: A survey of advances in vision-based human motion capture and analysis. Comput. Vis. Image Underst. **104**(2), 90–126 (2006)
12. Oikonomidis, I., Kyriazis, N., Argyros, A.: Efficient model-based 3D tracking of hand articulations using Kinect. In: BMVC, pp. 101.1-101.11. Dundee, UK (2011)
13. Oikonomidis, I., Kyriazis, N., Argyros, A.A.: Markerless and efficient 26-DOF hand pose recovery. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) Computer Vision–ACCV 2010. LNCS, vol. 6494, pp. 744–757. Springer, Berlin (2011)
14. Romero, J., Kjellstrom, H., Kragic, D.: Monocular real-time 3d articulated hand pose estimation. In: Humanoids, pp. 87–92. Paris, France (2009)
15. Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., Cook, M., Moore, R.: Real-time human pose recognition in parts from single depth images. Commun. ACM **56**(1), 116–124 (2013)
16. Sobol, I.M.: On the distribution of points in a cube and the approximate evaluation of integrals. USSR Comput. Math. Mathematical Phys. **7**(4), 86–112 (1967)
17. Wu, Y., Huang, T.S.: View-independent recognition of hand postures. In: CVPR, vol. 2, pp. 88–94. Hilton Head Island, USA (2000)
18. Xiong, X., De la Torre, F.: Supervised descent method and its applications to face alignment. In: CVPR, pp. 532–539. Portland, USA (2013)

Optimizing Orthonormal Basis Bilinear Spatiotemporal Representation for Motion Data

Przemysław Skurowski, Jolanta Socała and Konrad Wojciechowski

Abstract The paper describes an attempt to estimate the optimal division of a number of base vectors between space (shape) and time (trajectory) for bilinear spatiotemporal representation of motion capture data. The spatiotemporal model is a matrix consisting of $K_s \cdot K_t$ amount of coefficients. In the paper we discuss using of orthonormal spatial and temporal basis: PCA-PCA, DCT-DCT, PCA-DCT and DCT-PCA to represent real MoCap data.

Keywords Spatiotemporal representation · Motion capture · Motion data · Bilinear model

1 Introduction

Optical motion capture (Mocap) data [7, 8] comprises stored 3D trajectories of certain points over a time. The bilinear spatiotemporal representation [3] for motion data is a novel concept which offered an opportunity to store the motion information as spatiotemporal matrix being very efficient and compact form. The bilinear representation method can be considered as a combination of both spatial (shape) and temporal (trajectory) bases. The preferred bases are orthonormal ones, of which these obtained with principal component analysis (PCA) and from the discrete cosine transform (DCT) are most appreciated for a shape and trajectory respectively.

The model is very general as there are no limitations, so both rigid and soft objects can be represented. As it was demonstrated by the authors of the method,

P. Skurowski (✉)
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: przemyslaw.skurowski@polsl.pl

J. Socała
Department of Mathematics, University of Bielsko-Biała, Bielsko-Biała, Poland
e-mail: socala.jolanta@gmail.com

K. Wojciechowski
Polish-Japanese Academy of Information Technology, Bytom, Poland
e-mail: konrad.wojciechowski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_31

the representation appeared to be effective for denoising and labeling of non-rigid (facial) motion capture data. The model was also used to identify roles of football players [6, 12] by analysis of the team shape and motion of players.

In the paper, we address the problem of the optimal number of spatial and temporal dimensions which still remain open. We study all the four combinations of PCA and DCT basis for the reconstruction root mean square error (RMSE) for the bases with significantly reduced number of dimensions. We are looking for the optimal sharing of a number of basis vectors between shape and trajectory for approximately constant (fixed) overall number of coefficients. The experiments were conducted for a number of MoCap sequences acquired mostly in the Human Motion Lab of Polish Japanese Academy of Information Technology.

2 The Method

2.1 Bilinear Spatiotemporal Basis

Assume we have the time-varying structure of a set of P points sampled at F time instances. It can be represented as a sequence of 3D points:

$$\mathbf{S}_{F\times 3P} = \begin{bmatrix} \mathbf{X}_1^1 & \dots & \mathbf{X}_p^1 \\ \vdots & & \vdots \\ \mathbf{X}_1^F & \dots & \mathbf{X}_p^F \end{bmatrix}, \tag{1}$$

where: $\mathbf{X}_j^i = [X_j^i, Y_j^i, Z_j^i]$ denotes the $3D$ coordinates of the j-th point at the i-th time instance. Obviously, the time-varying structure matrix $\mathbf{S}$ contains $3FP$ parameters. We indicate row-index as superscript and column-index as subscript.

We can represent the $3D$ shape at each time instance as a linear combination of a small number K_s ($K_s \ll 3P$) of shape basis vectors $\mathbf{b}_j$ weighted by coefficients ω_j^i [4, 5],

$$\mathbf{s}^i = \sum_j \omega_j^i \mathbf{b}_j^T. \tag{2}$$

Every shape basis vector represents a $3D$ structure of length $3P$. The structure matrix $\mathbf{S}$ can be represented as:

$$\mathbf{S} = \Omega \mathbf{B}^T, \tag{3}$$

where: $\mathbf{B}$ is a $3P \times K_s$ matrix containing K_s shape basis vectors as its rows and Ω is a $F \times K_s$ matrix containing the corresponding shape coefficients ω_j^i.

We have also another representation. We can represent every trajectory as a linear combination of a small number K_t ($K_t \ll F$) of trajectory basis vectors $\boldsymbol{\theta}_i$ weighted by coefficients a_i^j [2, 11],

$$\mathbf{s}_j = \sum_i a_i^j \boldsymbol{\theta}_i. \tag{4}$$

Every trajectory basis vector represents a structure of length F. The structure matrix $\mathbf{S}$ can be represented as:

$$\mathbf{S} = \boldsymbol{\Theta}\mathbf{A}^T, \tag{5}$$

where: $\boldsymbol{\Theta}$ is a $F \times K_t$ matrix containing K_t trajectory basis vectors and $\mathbf{A}$ is a $3P \times K_t$ matrix containing the corresponding trajectory coefficients a_i^j.

We will use the third method—the Bilinear Spatiotemporal Basis. We assume: the vectors $\mathbf{b_j}$ are orthonormal and the vectors $\boldsymbol{\theta_i}$ are orthonormal too. The following theorem [3] give us a bilinear representation of $\mathbf{S}$.

Theorem 1 *Let us assume that we have* $\mathbf{S} = \Omega\mathbf{B}^T$ *and* $\mathbf{S} = \boldsymbol{\Theta}\mathbf{A}^T$. *Then it holds:*

$$\mathbf{S} = \boldsymbol{\Theta}\mathbf{C}\mathbf{B}^T, \tag{6}$$

where $\mathbf{C} = \boldsymbol{\Theta}^T\Omega = \mathbf{A}^T\mathbf{B}$ *is a* $K_t \times K_s$ *matrix of spatiotemporal coefficients.*

Above theorem is a case of a perfect reconstruction. It is also possible to consider a reduced base model, where the number of basis vectors is significantly smaller than the original number. Furthermore, it is worth to note that if $K_s \ll 3P$ and $K_t \ll F$ than the $K_t \times K_s$ coefficients in $\mathbf{C}$ can be orders of magnitude fewer than the $F \times K_s$ coefficients in Ω or the $K_t \times 3P$ coefficients in $\mathbf{A}$. The following theorem [3] gives us an estimation of a reconstruction error of the bilinear spatiotemporal model in such a case. The $||\cdot||_F$ is the Frobenius norm.

Theorem 2 *Let* $\epsilon_t = ||\mathbf{S} - \boldsymbol{\Theta}\mathbf{A}^T||_F$ *is the reconstruction error of the trajectory model and* $\epsilon_s = ||\mathbf{S} - \Omega\mathbf{B}^T||_F$ *is the reconstruction error of the shape model. Then for the reconstruction error of the bilinear spatiotemporal model* $\epsilon = ||\mathbf{S} - \boldsymbol{\Theta}\mathbf{C}\mathbf{B}^T||_F$ *we have* $\epsilon \leq \epsilon_t + \epsilon_s$.

For a shape basis $\mathbf{B}$ and a trajectory basis $\boldsymbol{\Theta}$, we compute the bilinear model coefficients $\mathbf{C}$, minimizing the reconstruction error for a given $\mathbf{S}$. We will use formula appropriate for the orthonormal bases [3]:

$$\mathbf{C} = \boldsymbol{\Theta}^T\mathbf{S}\mathbf{B}. \tag{7}$$

2.2 Considered Orthonormal Bases

In the original paper [3] proposing the bilinear spatiotemporal model, there were suggested two orthonormal bases—PCA and DCT based. The choice is very reasonable. The PCA is capable to adopt to virtually any set of body pose configurations or trajectories. The PCA can be obtained [9] through singular value decomposition:

$$\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T, \tag{8}$$

hence the transform is given as:

$$\mathbf{T} = \mathbf{X}\,\mathbf{V}, \tag{9}$$

where: $\mathbf{V}$ contains the eigenvectors as columns, Σ nonzero singular values.

Alas, for long sequences, obtaining full trajectory PCA can be computationally intensive and may require a relatively large amount of memory (gigabytes). The natural alternative is the commonly used base of Discrete Cosine Transform (DCT). The DCT is good approximation of PCA for 'natural' signals—also for motion—so trajectories should be represented well. On the other hand, it is demonstrated in the experimental part of the paper, one should not expect good performance in the shape representation. The DCT base [1] is given as:

$$D_{u,f} = \begin{cases} \dfrac{1}{\sqrt{F}} & , \quad \text{for } u = 0,\ 0 \le f \le F-1 \\ \sqrt{\dfrac{2}{F}} \cos \dfrac{\pi(2f+1)u}{2F} & , \quad \text{for } 0 \le u \le F-1,\ 0 \le f \le F-1 \end{cases} \tag{10}$$

3 Experiments

In order to reveal efficiency of the bilinear model we performed two experiments. First, to obtain overview of the efficiency in a function of K_s and K_t we performed exhaustive evaluation of model accuracy with RMSE. Each of the basis combination was tested. These results allowed us to neglect two of the combinations and further experiments on the selection appropriate K_s and K_t for the fixed number of coefficients were performed using selected approaches only.

3.1 The Data

For the testing purposes, we selected small, yet comprehensive set of MoCap recordings. It contains sequences recorded using different parameters: speed (60–200 Hz), lengths and number of markers (25–53) of various subjects and actions. Two subjects—male HJ and female IM—*Range of movement* (ROM) sequences (exercising all limbs and all rotation extremes for every joint) which caused the large variance in poses. Ordinary motion of human and non-human (dog) subjects that have a smaller variance in poses. Two non-rigid facial animations demonstrating simple spelling of the alphabet and a kind of 'ROM' for facial expressions (with head movements)—presenting smaller and larger pose variance respectively. Finally a hands typing the keyboard sequence which has limited variance in poses.

3.2 Overview of the Efficiency of Bases

In the experiment we examined all four combinations of PCA and DCT as spatiotemporal basis. The evaluations with the RMSE were performed exhaustively for a number of base vectors varying between 1 and 20. It provided a general overview of the performance of each type of basis and a brief view into the performance gain with the growing number of coefficients. The interpretation of these can be performed on observation how fast an error reduces with the growing number of base vectors. If it reduces slowly for the PCA basis, it implies more varied movement (larger number of poses). When we observe relatively large error (slow reduction) up to the large number of DCT base vectors, it suggests the presence of high frequencies and therefore fast motions in the sequence.

In the Fig. 1 we see the results for a ROM sequence for HJ subject, the results for IM were almost the same they are not illustrated. Wide set and range of possible body poses cause slower error decreasing with increasing number of base functions than it is in case of more ordinary motion sequences such as walk.

Ordinary motions for sit, walk and dog sequences also share a similar characteristics in bilinear model. The Fig. 2 illustrates a representative example (walk sequence). Such sequences demonstrate limited variability in the poses—error reduces very fast with the growing number of PCA base vectors. Also for the DCT in temporal domain error decays fast as the number of base vectors increases.

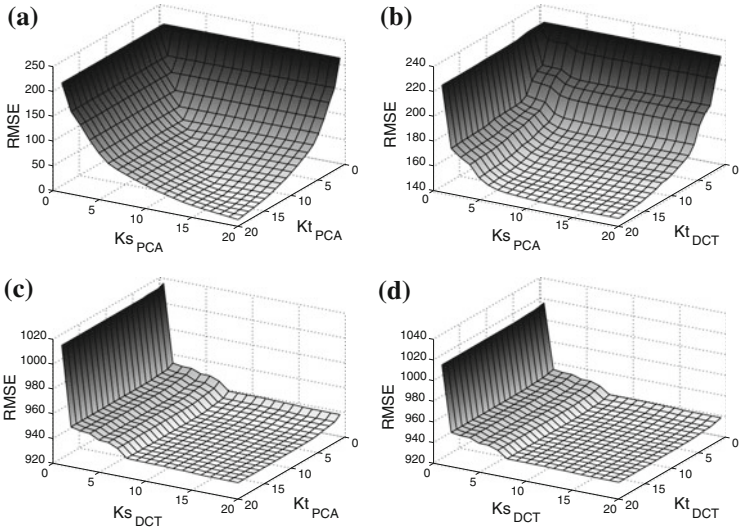


Fig. 1 Comparison of all combinations of spatio-temporal bases for ROM of HJ subject: **a** PCA-PCA, **b** PCA-DCT **c** DCT-PCA **d** DCT-DCT. (Please mind the scales)

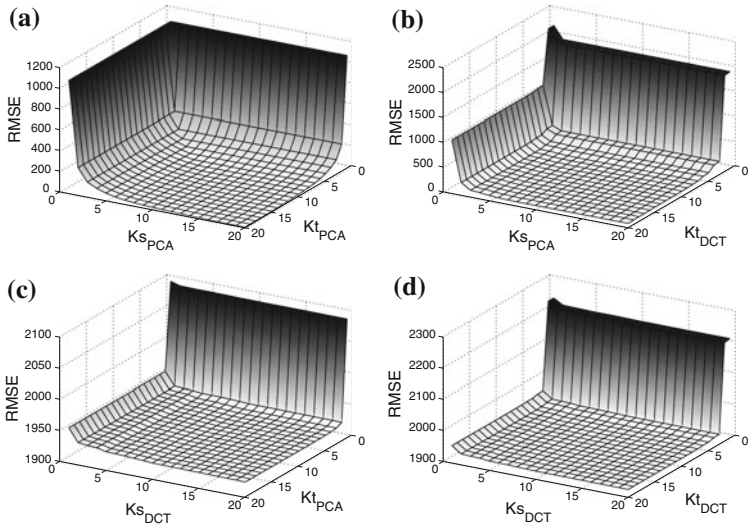


Fig. 2 Comparison of all combinations of spatio-temporal bases for HJ walk: **a** PCA-PCA, **b** PCA-DCT **c** DCT-PCA **d** DCT-DCT. (Please mind the scales)

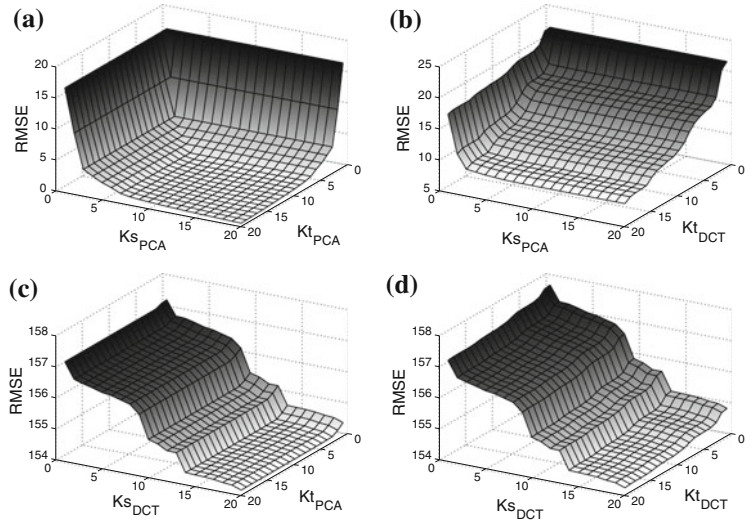


Fig. 3 Comparison of all combinations of spatio-temporal bases for facial expressions: **a** PCA-PCA, **b** PCA-DCT **c** DCT-PCA **d** DCT-DCT. (Please mind the scales)

Facial and hand sequences shared another, common characteristics—please see the Fig. 3 as representative one. They demonstrate quite a limited set of poses (especially spelling face and hands) so to represent their shape one needs relatively small amount of base vectors. Although, in the temporal domain of these cases, we observe that error reduces relatively slowly with the growing amount of DCT base vectors, so one can suspect that there are fast motions (frequencies) present in these recordings.

The test revealed the best performance of PCA-PCA basis with diagonal-symmetric RMSE reduction with the growing number of coefficients. The DCT appeared to perform poorly as a base for the representation of complex shape structure, whereas it is a good representation for the temporal data. So in further tests we rejected both model combinations based on the DCT as a shape basis.

3.3 Optimal Division of Base Vectors Number

In the case when we have a limited number of spatiotemporal coefficients—fixed N—there is a problem how to divide N between K_s and K_t. To address this question we conducted some dedicated tests, which were a series of RMSE evaluations of bilinear model for a number of feasible divisions. Since the coefficients array is rectangular where $N = K_s \cdot K_t$ it is not possible to get integer sizes in every case. Therefore, we decided to check the RMSE for N which was constant only approximately and so the characteristics in Fig. 4 might look a bit 'jaggy'. The iterations were for $K_s^i = K_s^{\min}, \ldots, K_s^{\max}$ where:

$$K_s^{\min} = \left\{ \begin{array}{l} \text{if } N > F : \lceil N/F \rceil \\ \text{else} \quad : \quad 1 \end{array} \right\}, \qquad K_s^{\max} = \left\{ \begin{array}{l} \text{if } N > 3P : 3P \\ \text{else} \quad : N \end{array} \right\},$$

and K_t was evaluated as $K_t^i = \text{round}\left(\frac{N}{K_s^i}\right)$.

Let's define $\alpha = K_s/(K_s + K_t)$ a relative share of spatial bases in the overall number of base vectors. We are looking for the α^{opt} resulting in minimal error. Having chosen α it is easy to show the numbers of base vector numbers are:

$$K_s = \text{round}(\sqrt{\frac{N\alpha}{1-\alpha}}), \qquad K_t = \text{round}(\frac{N}{K_s}). \tag{11}$$

Due to neglecting (in Sect. 3.2) of the usability of DCT as a base for the structured information (shape), further tests were performed using PCA-PCA and PCA-DCT base combinations only. The results are presented in Fig. 4 for all the test datasets. The reconstruction error for each of the datasets was tested, against various overall numbers of coefficients: $N = 2^2,\ 2^5,\ 2^{10},\ 2^{15},\ 2^{20},\ 2^{30},\ 2^{50}$.

The PCA-PCA base characteristics, with the optimal equal division of base vectors($\alpha^{opt} = 0.5$), is obvious because of the symmetry of importance of shape/time bases observed in Sect. 3.2. Although, the PCA-DCT result is not so trivial. We found out that $\alpha^{opt} \in (0.1,\ 0.3)$ so the shape base vectors should be approximately 10–30 % of an overall number (see dashed lines in Fig. 4). As one can note there is a

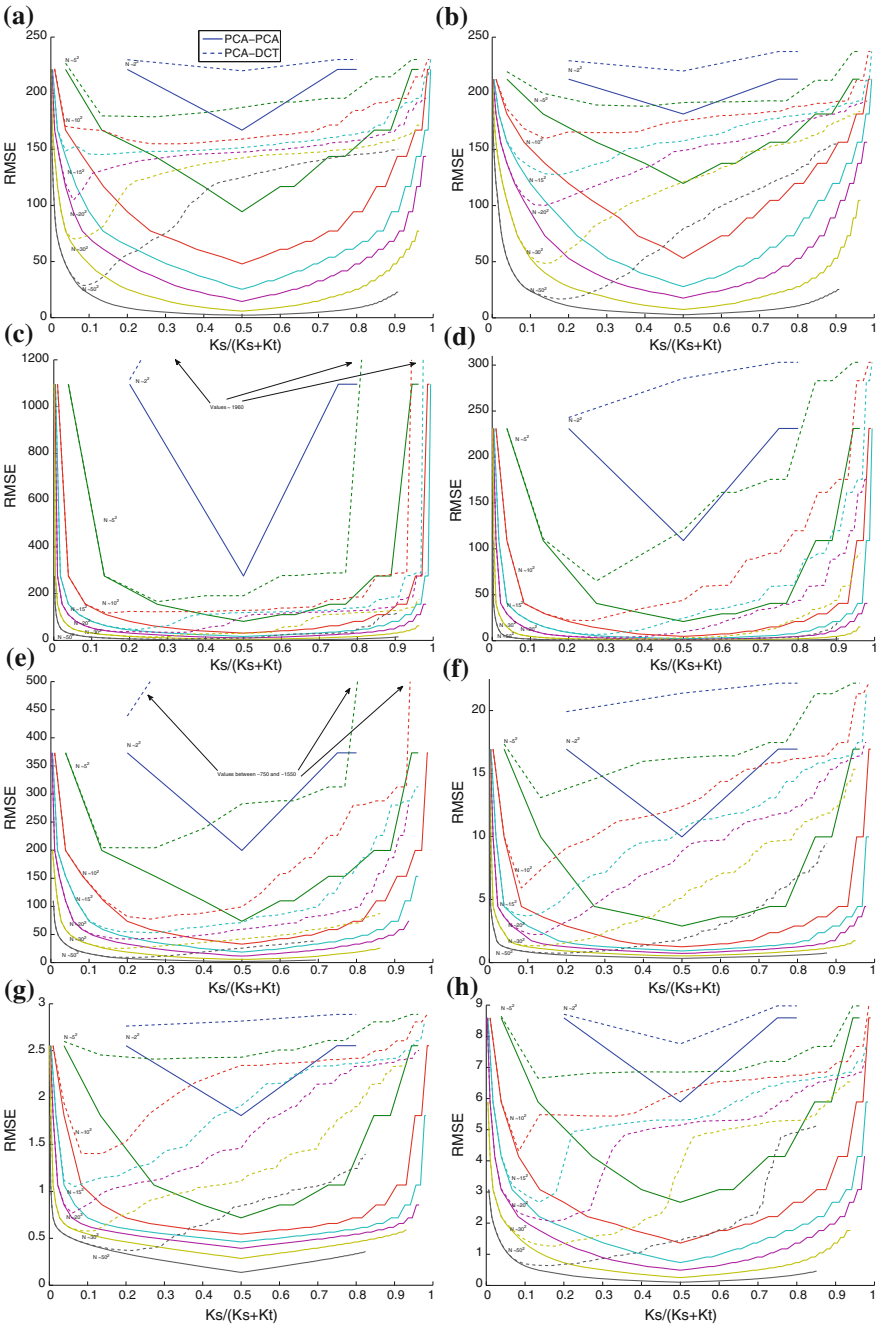


Fig. 4 Evaluation of bilinear model for various $\alpha = K_s/(K_s + K_t)$ coefficients division for the PCA-PCA (—) and PCA-DCT (—) bases—source sequences (**a**–**h**) as in Table 1

Table 1 Experimental MoCap sequences

No	Name	Description of sequence	Frames (F)	Mark. (P)
(a)	HJ-rom	Range of movement a male subject	10486 (52.4 s@200 Hz)	53
(b)	IM-rom	Range of movement a female subject	3675 (36.7 s@100 Hz)	53
(c)	HJwalk	Walk—turn (180°)—walk	1857 (9.3 s@200 Hz)	53
(d)	HJsit	Tpose-sit-standup	1618 (8.1 s@200 Hz)	52 (1 lost)
(e)	Dog[1]	Dog run, jump, turn, walk, step onto and off the table	717 (11.9 s@60 Hz)	25
(f)	Face-exp	Head moves, ROM for expressions emphasis on mouth and eyebrows	3918 (65.3 s@120 Hz)	45
(g)	Face-say	Face spelling alphabet	1780 (17.8 s@100 Hz)	36
(h)	Hands	Both hands typing the keyboard	794 (7.9 s@100 Hz)	40

Sequence from 'Dog package' (http://www.mocapclub.com/Pages/Library.htm)

single and well visible minimum in the characteristics, so other choices of dimensions result in degradation of reconstruction. Such observation is consistent for all the test cases so 20–80 division might be useful suboptimal choice (see Fig. 6) when we cannot search for optimal division.

The visual demonstration of results obtained is presented in the Fig. 5. It includes original body poses and reconstructions from bilinear representation with both considered bases for various and (sub)optimal base sizes. We can observe that PCA-DCT gradually (and slower) converges to the original shape with the growing number of coefficients, whereas PCA-PCA reaches proper shape quite fast and the shape is just a bit refined with the larger number of coefficients.

4 Summary

The bilinear representation for a motion capture sequences is a novel idea. In this study, we analyzed its performance using combinations of two fundamental basis (PCA and DCT) for their reconstruction error. We verified commonly known fact that there is no use to employ the DCT as a shape basis. The most obvious result is for PCA-PCA basis. It is symmetric and results in the best error reduction with the growing number of coefficients, which result in 50-50 optimal sharing between shape and trajectory.

The most notable result, we obtained for the PCA-DCT as a basis. We discovered interesting property for division of a base vectors number between spatial and temporal parts for respectively PCA and DCT bases. The optimal division favors trajectory bases with the relative share of shape bases in the overall number between 0.1 and 0.3, therefore, one could consider 20–80 division.

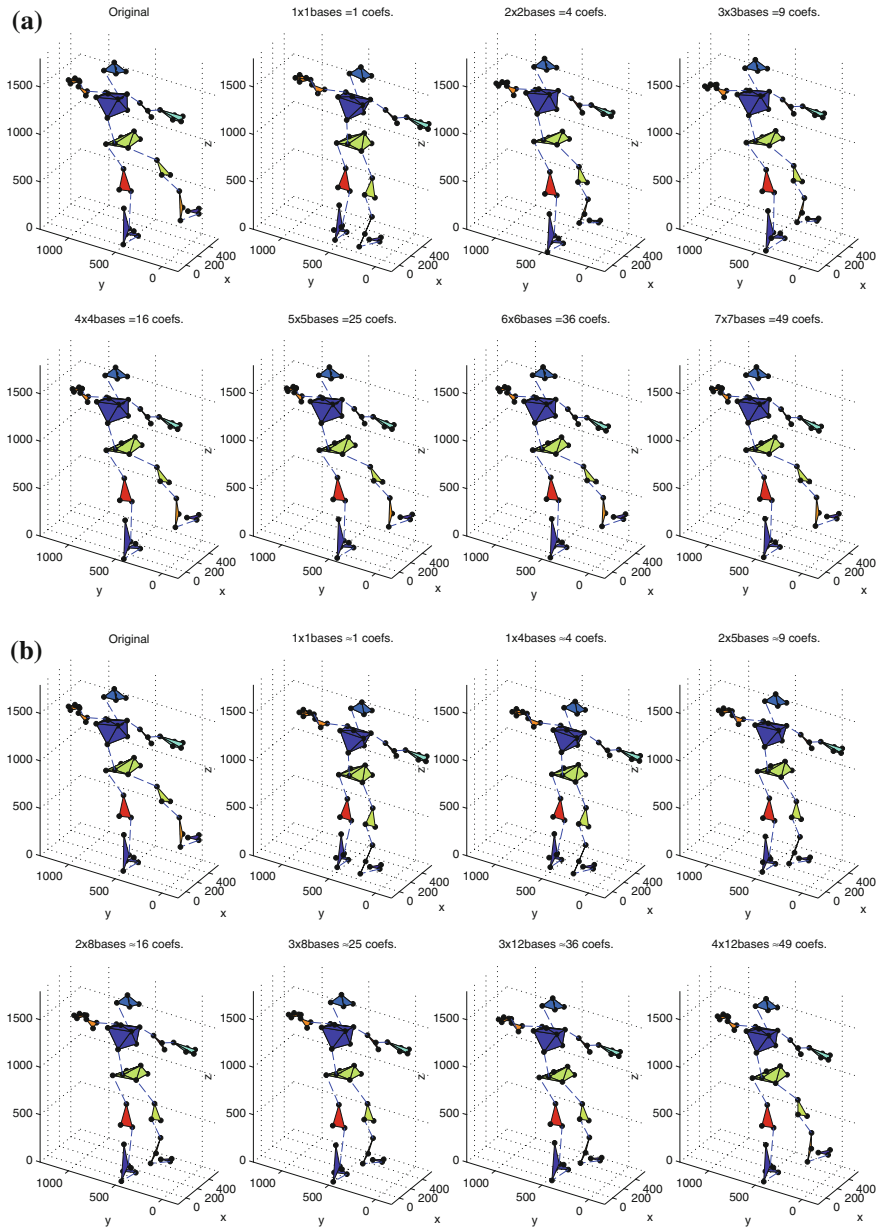


Fig. 5 A single frame from HJrom sequence reconstructed with increasing number of PCA-PCA (**a**) and PCA-DCT (**b**) bases (20–80)—body visualized with FBM [10]

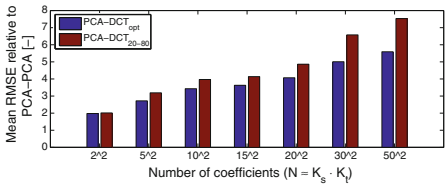


Fig. 6 Mean RMSE of PCA-DCT relative to PCA-PCA for α_{opt} and 20–80 setup

Further research will focus on using the bilinear model for pattern recognition. In such a case it would be necessary to use the ability of class discrimination as performance assessment criteria. Also, other basis should be also taken into consideration instead of PCA-ICA or LDA seem to be appropriate.

Acknowledgments This research has been supported by Demonstrator + Programme of NCRD. Project UOD-DEM-1–183/001 "System inteligentnej analizy wideo do rozpoznawania zachowań i sytuacji w sieciach monitoring".

References

1. Ahmed, N., Natarajan, T., Rao, K.: Discrete cosine transform. IEEE Trans. Comput. C-**23**(1), 90–93 (1974)
2. Akhter, I., Sheikh, Y., Khan, S., Kanade, T.: Nonrigid structure from motion in trajectory space. In: NIPS, pp. 41–48. Vancouver, Canada (2009)
3. Akhter, I., Simon, T., Khan, S., Matthews, I., Sheikh, Y.: Bilinear spatiotemporal basis models. ACM Trans. Graphics **31**(2), 1–12 (2012)
4. Bregler, C., Hertzmann, A., Biermann, H.: Recovering non-rigid 3d shape from image streams. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 690–696. Hilton Head Island, SC, USA (2000)
5. Cootes, T.F., Taylor, C.J., Cooper, D.H., Graham, J.: Active shape models;their training and application. Comput. Vis. Image Underst. **61**(1), 38–59 (1995)
6. Lucey, P., Bialkowski, A., Carr, P., Morgan, S., Matthews, I., Sheikh, Y.: Representing and discovering adversarial team behaviors using player roles. In: CVPR, pp. 2706–2713. Portland, USA (2013)
7. Moeslund, T.B., Granum, E.: A survey of computer vision-based human motion capture. Comput. Vis. Image Underst. **81**(3), 231–268 (2001)
8. Moeslund, T.B., Hilton, A., Krueger, V.: A survey of advances in vision-based human motion capture and analysis. Comput. Vis. Image Underst. **104**(2), 90–126 (2006)
9. Shlens, J.: A tutorial on principal component analysis. arXiv:1404.1100. (2014)
10. Skurowski, P., Pawlyta, M.: Functional body mesh representation—a simplified kinematic model. In: ICNAAM 2014, vol. 1648, pp. 660008–1–4. Rhodes, Greece (2015)
11. Torresani, L., Bregler, C.: Space-Time Tracking. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) Computer Vision—ECCV 2002. LNCS, vol. 2351, pp. 801–812. Springer, Berlin (2002)
12. Wei, X., Sha, L., Lucey, P., Morgan, S., Sridharan, S.: Large-scale analysis of formations in soccer. In: DICTA, pp. 1–8. Hobart, Australia (2013)

Evaluation of Improvement in Orientation Estimation Through the Use of the Linear Acceleration Estimation in the Body Model

Agnieszka Szczęsna, Przemysław Pruszowski, Janusz Słupik, Damian Pęszor and Andrzej Polański

Abstract The need for broadly defined measures of human motion and estimation of motion parameters occurs in many research disciplines. The article concerns the evaluation of improvement in basic orientation estimation methods for IMU sensors, throughout the use of rigid body (segment in skeleton model) constraints. The verified method utilizes the correlation between rational motions and linear accelerations in the body model and reduces the impact of the external acceleration on the estimation of orientation. The experimental study concerns comparison of this method to other methods of leveling the influence of linear external acceleration.

Keywords Inertial motion capture · IMU sensor · Orientation filters · Linear acceleration estimation

1 Introduction

Motion capture (MOCAP) technology involves the saving of complex, multi-segmental or multi-body dynamical scenes on the basis of specialized instrumentation and measurement systems.

A. Szczęsna (✉) · P. Pruszowski · D. Pęszor
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: agnieszka.szczesna@polsl.pl

P. Pruszowski
e-mail: przemyslaw.pruszowski@polsl.pl

D. Pęszor
e-mail: damian.peszor@polsl.pl

J. Słupik
Institute of Mathematics, Silesian University of Technology, Gliwice, Poland
e-mail: janusz.slupik@polsl.pl

A. Polański
Research Center of Polish-Japanese Academy of Information Technology, Bytom, Poland
e-mail: apolanski@pjwstk.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_32

The research described in this article is necessary for the construction of the *Inertial Motion Capture System* for out-door acquisition and motion analysis. The system operates primarily by determining the orientation rather than position of each sensor. A person wears a costume with multiple IMU sensors (like presented in [4]), each attached to the tracked body segment. Segments are defined by bones in skeleton. The mapping of the estimated orientations of inertial sensors (IMU) to specific segments on body model which is composed of rigid bodies allows for the motion capture of subject. The orientations of the sensors are typically estimated by fusing a rate gyroscope (ω), an accelerometer (a), a magnetometer (m) and reference values such as Earth's gravity vector (g) and magnetic field vector (mg) by a complementary filter [7] or different Kalman filters [8, 12, 16, 17]. With knowledge of the orientations of each body segment over time with the structure of skeleton, the overall pose can be tracked.

In this paper we are not proposing a new algorithm for the estimation of the orientation of the IMU sensor. Instead, we propose an evaluation, in a simple experiment, of different methods to overcome errors in orientation estimation arising out of the linear acceleration, and especially the method based on the body model.

Algorithms for estimation of the orientation of sensors are a basis for all IMU motion tracking systems and were already developed and studied in numerous papers in the literature (example review in [9, 10]). Presented are also descriptions of the entire kinematic chains based on detailed assumptions about the skeletal and the dynamics of movement [1, 5]. In such configurations a great influence on the calculated orientation have also the other system elements, such as static (initialization calibration poses) and dynamic (due to the possibility of changing the orientation of the sensor relative to the motion segment and soft tissues) calibration. The main purpose was to check in a simple experimental configuration how the use of the linear acceleration estimation in the model-based skeleton improves the determined orientation. Therefore, a basic filter has been selected as simple complementary filter. The following techniques were compared: simple gated acceleration, body model linear acceleration estimation and adaptation mechanism.

2 Orientation Estimation Filters

In this paper we are focusing only on filters based on a quaternion representation of body orientation. Quaternions are used to represent orientation to improve computational efficiency and avoid singularities. The filters are based on the well-known correlation between the angular velocity ω and the quaternion derivative:

$$\dot{\boldsymbol{q}} = \frac{1}{2}\boldsymbol{q} \otimes \boldsymbol{\omega} \tag{1}$$

where q is the orientation quaternion and $\otimes$ denotes quaternion multiplication.

Based on this equation the orientation is determined by integrating the output signal from the gyroscope (ω), so the accuracy of determining the angle to the greatest extent depends on the sensor stability of zero. A zero drift (for example due to changes in temperature) results, in a short time, in the large error values in the determined angle. The integration accumulates the noise over time and turns noise into the drift, which yields unacceptable results.

For a stationary sensor in an environment free of magnetic anomalies, it is simple to determine the orientation by measuring the Earth's gravitational and magnetic fields along all three axes of the orthogonally mounted sensors. The combination of the two resulting vectors can provide complete orientation information. Such solutions are based on a well defined Wahba problem [11, 13]. In more dynamic applications, high frequency angular rate information can be combined in a complementary manner with accelerometer and magnetometer data through the use of a sensor fusion algorithms like that of complementary [7] or Kalman filters [8, 12, 16, 17].

2.1 Nonlinear Complementary Filter (NCF)

The primary nonlinear complementary filter (NCF) was used as a basis to make modifications to overcome the problem of influence of external linear acceleration on the orientation estimation (Fig. 1). In this filter it is assumed that there is no external acceleration of the sensor or that the magnitude of the external acceleration is negligible compared to the gravity acceleration [7]. Measurements provided by the magnetic and acceleration sensors are sufficient for computation of the orientation of the IMU and gyroscopic measurements can be additionally used for increasing accuracy and robustness.

The idea of the construction of an orientation estimator is in the modification of the equation of kinetics of motion by using the nonlinear feedback based on the instantaneous estimate of relative rotation: $q_{inst} = q_{inst}(a_k, m_k)$, where a_k and m_k are output k sample from the accelerometer and magnetometer sensor.

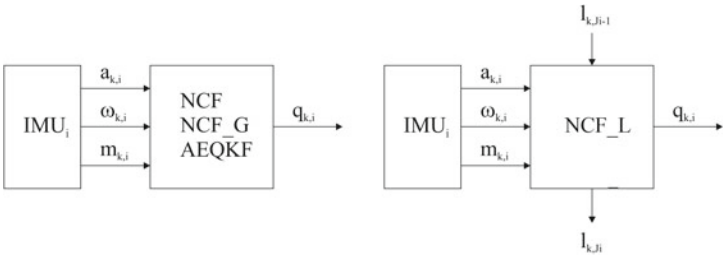


Fig. 1 The implemented filters block diagram: NCF, NCF_G, AQKF, NCF_L

The equation for the dynamics of orientation estimate, with nonlinear feedback, using the quaternion representation of orientation, has the following form:

$$\frac{d}{dt}\hat{q}(t) = \frac{1}{2}\hat{q}(t) \otimes (q_\omega + k_p\tilde{\omega}) \tag{2}$$

where q_ω denotes the pure quaternion from gyroscope measurement ω_k, k_p is the feedback gain and $\tilde{\omega}$ is the nonlinear term corresponding to the error between filter estimate $\hat{q}$ and the instantaneous estimate of the orientation quaternion $q_{inst}(a_k, m_k)$. The $\otimes$ denotes quaternion multiplication.

The computational implementation of the nonlinear, complementary filter involves a discretized version of (2). The initial condition is $\hat{q}_0 = q_{inst}(a_0, m_0)$.

2.2 Gated Nonlinear Complementary Filter (NCF_G)

The measurement equation for accelerometers, is following: $a = -g + l + w$. The g is constant acceleration due to gravity, l is the linear acceleration of the sensor due to movement and w is the noise. Any component of l not parallel to g will cause an error in orientation estimation. The effects of linear acceleration l can be reduced if we can detect this situation and do the estimation without a corrupted vector. A most simple approach to detecting the linear acceleration is based on the magnitude of the measured acceleration vector. In the gated filter the correction based on measurements from acceleration is only performed, when: $\|g\| - a_T < \|a\| < \|g\| + a_T$, where a_T is acceleration threshold. The gated acceleration is used in implemented nonlinear complementary filter. When the magnitude of acceleration is out of bounds, the higher weights k_p (in (2)) for the gyroscopic measurements and lower weights for the accelerometer measurements are used (Fig. 1).

2.3 Adaptive Extended Quaternion Kalman Filter (AEQKF)

Another method is based on the adaptation mechanism implemented in the extended Kalman filter. In the original design [8] the state vector (here denoted by x^{orig}) includes three components (blocks) $x^{orig} = \begin{bmatrix} q \\ b^a \\ b^m \end{bmatrix}$. The quaternion q represents the body orientation and two vectors b^a and b^m represents biases of the sensors, accelerometers and magnetometers. This is a direct-state quaternion-based formulation of the EKF, where angular velocity is considered a control input and active compensation (gyro bias and magnetic effects) is achieved by using state-augmentation techniques. Since we ignore biases b^a and b^m, here we simplify the definition of the state vector by retaining only the quaternion representing the body orientation, as $x = q$ (Fig. 1).

The discretized state equation of the orientation kinematics process corresponding to (1), is the following:

$$x_{k+1} = \Phi_k x_k + w_k = \exp[\frac{1}{2} M_R(\omega_k)\Delta t] x_k + w_k. \tag{3}$$

In this equation x_k is the discrete—time state vector, $x_k = q_k$, and $M_R(\omega_k)$ denotes a matrix representation of the quaternion right multiplication corresponding to the pure quaternion ω_k and Φ, which is a state transition matrix. The process noise covariance matrix Q_k is following:

$$Q_k = (\Delta t/2)^2 \Xi_k (\sigma_g^2 I_{4x4}) \Xi_k^T. \tag{4}$$

where for $q_k = (a, [b, c, d])$ we define as follows

$$\Xi_k = \begin{bmatrix} a & -d & c \\ d & a & -b \\ -c & b & a \\ -b & -c & -d \end{bmatrix}$$

The measurement model is of the form:

$$z_{k+1} = \begin{bmatrix} a_{k+1} \\ m_{k+1} \end{bmatrix} = f(x_{k+1}) + \begin{bmatrix} w_k^a \\ w_k^m \end{bmatrix} = \begin{bmatrix} R(q_{k+1}) & 0 \\ 0 & R(q_{k+1}) \end{bmatrix} \begin{bmatrix} g \\ mg \end{bmatrix} + \begin{bmatrix} w_k^a \\ w_k^m \end{bmatrix} \tag{5}$$

where $R(q)$ is a rotation matrix defined by quaternion q.

There are two adaptation mechanisms assumed in [8], one for accelerometers and one for magnetometers. Here we implement only the adaptation regarding the accelerometer measurement, where the covariance matrix of the measurement V_{k+1} in Kalman gain equation (6)

$$K_{k+1} = P_{k+1}^- H_{k+1}^T (H_{k+1} P_{k+1}^- H_{k+1}^T + V_{k+1})^{-1} \tag{6}$$

depends on the deviation of the value of the gravitational acceleration $\|g\|$ and the measured acceleration magnitude $\|a_{k+1}\|$, i.e.

$$V_{k+1} = \begin{bmatrix} \sigma_a^2 \cdot I_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & \sigma_m^2 \cdot I_{3\times 3} \end{bmatrix} \tag{7}$$

if $|\|a_{k+1}\| - \|g\|| < \epsilon$ or

$$V_{k+1} = \begin{bmatrix} \infty \cdot I_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & \sigma_m^2 \cdot I_{3\times 3} \end{bmatrix} \tag{8}$$

otherwise. Thus, the influence of accelerometers on the orientation estimation is reduced in the presence of external acceleration.

3 Linear Acceleration Estimation

Any orientation estimation based purely on local sensor knowledge is limited in its accuracy by the effects of gyroscope drift and linear external acceleration measured with a gravity vector. Method, based on the distributed estimation of linear accelerations in a segment, can have the orientation error reduced by the subtraction of these accelerations from measured accelerations [6, 14, 15]. The previously described methods treat body segments individually without the use of important skeleton connectivity constraints.

The skeleton model consists of joints which connect the segments. Every segment has a sensors attached to it with a constant vector offset from the center of rotation. Each segment (with IMU) and joint has a local coordinate frame, related with a coordinate frame of sensor. The joints form a hierarchy structure with the position of a child joint given by an offset from the parent joint center. Results orientations are calculated in a world coordinate frame based on two reference vectors (gravity g and magnetic field mg).

The estimated linear acceleration of the sensor is considered in the case of rigid body rotating about a point, fixed at the origin with angular velocity ω. Every point on this body will have a radial linear acceleration:

$l_r = (\omega \bullet o)\omega - o\,\|\omega\|^2$, where o is the offset of the point from the center of rotation. In the presented model it is offset of i sensor o_{S_i} or joint o_{J_i} and ω is the gyroscope output signal.

Also, every point on the rigid body has a tangential acceleration:
$l_t = \alpha \times o$, where α is an angular acceleration calculated from angular velocity as:
$\alpha = \frac{\omega_{k+1} - \omega_{k-1}}{2\Delta t}$.

The whole segment is in a rotating frame with a linear acceleration l_f and this is a linear acceleration of parent segment in skeleton model. The result linear acceleration of a point under these assumptions is therefore: $l = l_f + l_r + l_t$

All linear accelerations that are passed between segments i.e. from parent to child in the skeleton (Fig. 1), are in the known the world coordinate frame. In calculations two equations are used. The equation for i sensor:

$$l_{S_i} = l_{J_{i-1}} + \alpha_{S_i} \times o_{S_i} + (\omega_{S_i} \bullet o_{S_i})\omega_{S_i} - o_{S_i} \left\|\omega_{S_i}\right\|^2 \tag{9}$$

Equation for all child joints of sensor i:

$$l_{J_i} = l_{J_{i-1}} + \alpha_{S_i} \times o_{J_i} + (\omega_{S_i} \bullet o_{J_i})\omega_{S_i} - o_{J_i} \left\|\omega_{S_i}\right\|^2 \tag{10}$$

4 Experiments

For the sake of comparison of the three methods that are outlined in order to determine the leveling effect of acceleration on orientation estimation, a 3-segment pendulum has been build. As reference, data is used from the optical system of motion capture

(Vicon system). Experiments demonstrate that using body model constraints can slightly improve the accuracy of the inertial motion capture system by removing the effects of external linear acceleration. Through simple manual calibration and mounting sensors permanently we remove the effect of bad calibration factors on the estimation of orientation.

In papers in the literature described such methods, experiments were based on synthetically generated input filter signals [14] for human skeleton or real data from human arm model [15]. Experiments covered a short period of time (max 13.5 s), when the drift of integration is still small and doesn't have an influence on angular acceleration calculations α. The filter with adaptation mechanism which was used in the comparison were not considered in other papers.

The pendulum was built with three segments connected by movable joints. An IMU sensor built at the *Silesian University of Technology, Department of Automatic Control and Robotics* [3] was fixed to each segment. These IMU sensors have been marked as IMU1, IMU2 and IMU3 (Fig. 2). On the pendulum markers were also attached, marked as R1, R2, W1, W2, W3, W4, W5, and W6.

Twenty three capture recordings with eight different scenarios (each scenario repeated 3 times) were carried out using the Vicon system with a frequency of 100 Hz. The IMU sensors also worked with such a frequency. The recordings had a length from 9600 to 19,840 samples. The recorded movement is characterized by high values of acceleration amplitudes of about 20 $\mathrm{m/s^2}$. The optical system also enabled calibration of sensors and calculation of the necessary distances.

Filter parameters are following:

- reference vectors: $g^N = [0, 0, -9.81]^T$ and $mg^N = [\cos(\varphi^L) - \sin(\varphi^L)]^T$, where φ^L is the geographical latitude angle. For the geographical position of the laboratory, wherein measurements were done, we have $\varphi^L = 66° = 1.1519$ rad;
- gating threshold in NCF_G filter: $a_T = 0.1$;

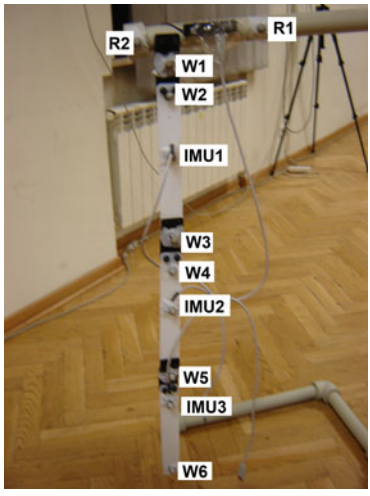


Fig. 2 The pendulum

- parameter k_p in NCF and NCF_L filters: $k_p = 2$;
- parameter k_p in NCF_G filter: if acceleration is in the bound $k_p = 2$, and if elsewhere $k_p = 0.2$;
- parameters of AEQKF filter: $\epsilon = 0.4$, $\sigma_a^2 = 0.001$ and $\sigma_m^2 = 0.00001$.

4.1 Data Synchronization

Each experiment was recorded using Vicon Nexsus system with a sampling frequency of 100 Hz. In order to provide an informative comparison of orientation data streams with different reference frames and measured according to separate timers with the same frequency, the data must be normalized. Such a procedure can be divided into two steps: normalization in the time domain (time synchronization) and transforming orientations to the same reference frame.

Transforming one orientation data stream from one reference frame to the other one is a simple geometric operation—rotation. Only knowledge about the relationship between two world reference frames—navigation and body—is required. As a reference body frame, the first body frame from time domain is chosen.

Signals from the Vicon system and IMU sensors are captured at the same frequency. Knowing that, in order to synchronize the time domain we need to find the time offset (Δt) between the two signals. A time window is chosen $< -\Delta t^{Max}, \Delta t^{Max} >$ where Δt^{Max} is the maximal offset we expected ($-\Delta t^{Max} < \Delta t < \Delta t^{Max}$). The distance between the two signals for each time offset in the window is calculated. Synchronization is performed on the ω_{IMU}^{B} signal. The Vicon system does not calculate angular velocity of the body directly, so it must be calculated by the equation:$\omega_{Vicon} = 2 * q_{Vicon}^{-1} \otimes \frac{dq_{Vicon}}{dt}$, where q^{-1} is the inverse of q.

4.2 Error Calculation

The evaluation of performances of the presented algorithms was done on the basis of the average deviations between true and estimated orientations of the body [2]. Here we use the deviation index DI corresponding to the geodesic distance between two quaternions—filter estimate $\hat{q}$ and the true rotation q from the Vicon system, on the hypersphere S^3: $DI = 2 * arccos(|\hat{q} * q|)$. All evaluations and comparisons of the performances of algorithms for orientation estimation are based on deviation index averaged over the experiment time horizon.

5 Result Discussion

In Fig. 3 the results of NCF_L estimation of orientation are presented, the results are in the form of three Euler angles for the segment 2. As a reference, the angles computed using Vicon system are also shown. The data has been synchronized in time and converted to the same frame using algorithms described in Sect. 4.1. All algorithms correctly estimate the yaw angle. The signal from the accelerometer is involved in the estimation of the roll and pitch angle, thus the biggest errors occur here. In the filter NCF_G when large acceleration is detected, the dominant operation in the orientation calculation is the integration of the gyroscope. Therefore, it can be seen in the results of periodic influence of drift. The similar adaptation mechanism is in filter AEQKF. In filter NCF_L the measured acceleration is corrected.

In Fig. 4 are shown the average errors (in radians) for the three segments (S1, S2, and S3) for all experiments. The biggest errors are for segment 3 because the greatest acceleration occurs there. In segment 3 the greatest improvement is by using

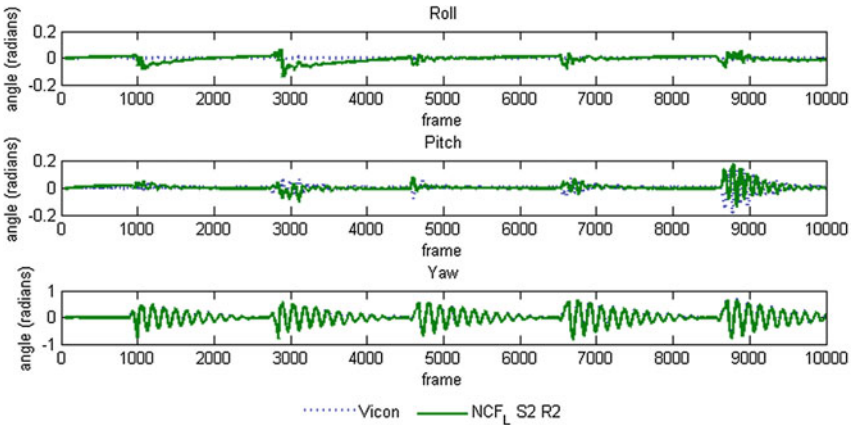


Fig. 3 Result Euler angles estimated by filter NCF_L and reference data from Vicon system, for segment 2 (S2) in capture 2 (R2) (first 10,000 frames)

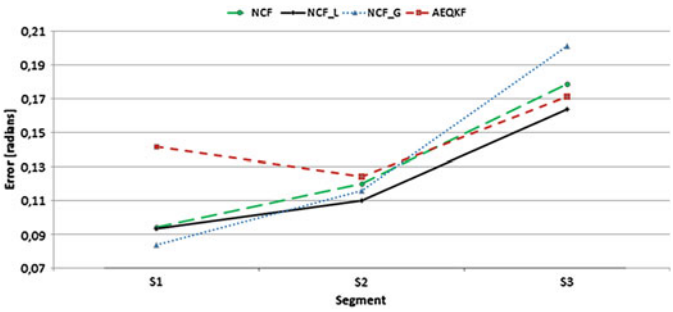


Fig. 4 Error angles between IMU based orientation estimation and orientation from optical motion capture system

the linear acceleration estimation by NCF_L filter. The AEQKF filter performed with worse results for segment 1 and 2. But for segment 3 filter AEQKF obtains good results, better than NCF and NCF_G. The AEQKF filter is more complex and needs better parameter tuning to adapt the filter to the specific motion dynamics, in order to achieve better results. In reality this is very difficult because we do not know the future body motion. In experiments we can see that a simple gating or adaptation mechanism leads to slightly larger errors than by using an acceleration estimation based on model constraints. This shows the strong influence of other factors on orientation estimation, i.e. gyroscope drift, signal noise and magnetic interference.

6 Summary

The article presents an evaluation of opportunities to improve the orientation estimation by using skeleton constraints. The results are shown for the three filters: nonlinear complementary NCF, nonlinear complementary with the mechanism of gating NCF_G, and the nonlinear complementary with skeleton acceleration estimation method NCF_L and the extended Kalman filter with the adaptation mechanism AEQKF. The results show a superiority of the solution based on the estimation of acceleration in the body model (skeleton), especially in child segments. The next step in the research will be to use this method in costume for Inertial Motion Capture with a human skeleton model.

Acknowledgments This work was supported by statute project of Silesian University of Technology, Institute of Informatics (BK-266/RAU-2/2014). This work was partly performed using the infrastructure supported by POIG.02.03.01-24-099/13 grant: "GCONiI—Upper-Silesian Center for Scientific Computation". Data was captured in Human Motion Laboratory of Polish-Japanese Academy of Information Technology (http://hm.pjwstk.edu.pl/en/).

References

1. El-Gohary, M., McNames, J.: Shoulder and elbow joint angle tracking with inertial sensors. IEEE Trans. Biomed. Eng. **59**(9), 2635–2641 (2012)
2. Gramkow, C.: On averaging rotations. J. Math. Imaging Vis. **15**(1–2), 7–16 (2001)
3. Jedrasiak, K., Daniec, K., Nawrat, A.: The low cost micro inertial measurement unit. In: ICIEA 2013, pp. 403–408. Melbourne, Australia (2013)
4. Kulbacki, M., Koteras, R., Szczęsna, A., Daniec, K., Bieda, R., Słupik, J., Segen, J., Nawrat, A., Polański, A., Wojciechowski, K.: Scalable, wearable, unobtrusive sensor network for multimodal human monitoring with distributed control. In: MBEC 2014, pp. 914–917. Dubrovnik, Croatia (2015)
5. Lin, J.F., Kulić, D.: Human pose recovery using wireless inertial measurement units. Physiol. Meas. **33**(12), 2099 (2012)
6. Luinge, H.J., Veltink, P.H.: Measuring orientation of human body segments using miniature gyroscopes and accelerometers. Med. Biol. Eng. Comput. **43**(2), 273–282 (2005)

7. Mahony, R., Hamel, T., Pflimlin, J.M.: Nonlinear complementary filters on the special orthogonal group. IEEE Trans. Autom. Control **53**(5), 1203–1218 (2008)
8. Sabatini, A.M.: Quaternion-based extended kalman filter for determining orientation by inertial and magnetic sensing. IEEE Trans. Biomed. Eng. **53**(7), 1346–1356 (2006)
9. Sabatini, A.M.: Estimating three-dimensional orientation of human body parts by inertial/magnetic sensing. Sensors **11**(2), 1489–1525 (2011)
10. Sabatini, A.M.: Kalman-filter-based orientation determination using inertial/magnetic sensors: observability analysis and performance evaluation. Sensors **11**(10), 9182–9206 (2011)
11. Shuster, M.D.: The generalized Wahba problem. J. Astronaut. Sci. **54**(2), 245–259 (2006)
12. Słupik, J., Szczęsna, A., Polański, A.: Novel lightweight quaternion filter for determining orientation based on indications of gyroscope, magnetometer and accelerometer. Comput. Vis. Graph. LNCS **8671**, 586–593 (2014)
13. Wahba, G.: A least squares estimate of satellite attitude. SIAM Rev. **7**(3), 409–409 (1965)
14. Young, A., Ling, M.J., Arvind, D.: Distributed estimation of linear acceleration for improved accuracy in wireless inertial motion capture. In: IPSN 2010, pp. 256–267. Stockholm, Sweden (2010)
15. Young, A.D.: Use of body model constraints to improve accuracy of inertial motion capture. In: BSN 2010, pp. 180–186. Singapore (2010)
16. Yun, X., Aparicio, C., Bachmann, E.R., McGhee, R.B.: Implementation and experimental results of a quaternion-based kalman filter for human body motion tracking. In: ICRA 2005, pp. 317–322. Barcelona, Spain (2005)
17. Yun, X., Bachmann, E.R.: Design, implementation, and experimental results of a quaternion-based Kalman filter for human body motion tracking. IEEE Trans. Robot. **22**(6), 1216–1227 (2006)

Part VII
Decision Support and Expert Systems

Data Cleansing Using Clustering

Petr Berka

Abstract One of the data quality issues, that heavily influences the performance of the classifiers learned from data is the amount of examples, that are indistinguishable but belong to different classes. Such situation occurs not only if the data (either original or preprocessed) contain contradictions, i.e. examples that have same values of input attributes but different class labels but also if different (but very similar) examples of different classes remain indistinguishable during the subsequent machine learning step. We propose a clustering based approach that tries to resolve such situation by identifying and removing these "weak" contradictions. The effect of data cleansing is then evaluated using decision tree learning algorithm on the reduced data set. To experimentally evaluate our method, we used some benchmark data from the UCI Machine Learning repository.

Keywords Data quality · Data cleansing · Clustering · Decision trees

1 Introduction

One of the data quality issues, that heavily influences the performance of the classifiers learned from data is the amount of contradictory examples (contradictions). By contradictory examples we understand examples, that have same values of input attributes but differ in the class label. Such examples are indistinguishable by any of the machine learning algorithms and thus degrade the classification accuracy. The sources of contradictions in data can be manifold. The original data can contain attributes that violate the similarity-based learning assumption. According to this assumption, examples that belong to the same class are similar and thus form clusters in the attribute (feature) space. E.g. in loan application domain, where the classes correspond to the loan approval decision, attributes related to the financial status of an applicant (e.g. `monthly_income` or `account_balance`) seem to fulfill the similarity-based assumption while attributes describing her/his visage

P. Berka (✉)
University of Economics, Prague, Czech Republic
e-mail: berka@vse.cz

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_33

(e.g. `hair_color` or `wearing_glasses`) apparently do not. Another source of contradictions in original data are, of course, typing errors or wrong values. Contradictions can also be added during data pre-processing. Examples of different classes that differ in the original data (original set of attributes) can be turned after discretization or attribute selection into examples that are represented by same values of the modified set of input attributes.

We already discussed the question of contradictions caused by discretization in our previous work [1]. In this paper, we extend the concept of contradictions to groups (clusters) of similar examples that belong to different classes. Thus we deal here not with "strong" contradictions but with examples that even if different became indistinguishable by the classifier created in the subsequent learning step. We propose and experimentally evaluate a clustering based method to identify such "weak" contradictions. The found contradictions are then resolved by removing selected examples from each cluster containing contradictory examples with the aim to improve the accuracy of classifiers created from the reduced data set. To check the effect of removing contradictions we use a decision tree learning algorithm that partitions data into groups of examples (regions in the attribute space) so that each group is assigned to a single class.

The rest of the paper is organized as follows. Section 2 reviews some work on using clustering for data cleansing, Sect. 3 describes the proposed approach, Sect. 4 reports results of experimental evaluation of the method and Sect. 5 concludes the paper.

2 Related Work

Data cleansing is the process of detecting and correcting (or removing) corrupt or inaccurate records from a database. The term refers to identifying incomplete, incorrect, inaccurate, inconsistent or irrelevant parts of the data and then replacing, modifying, or deleting them. Some work has been already carried out on using clustering for this purpose.

An early approach to detecting duplicities in databases was presented by Hernandes and Stolfo [6]. Here data were clustered first and then, for each cluster, sorted-neighborhood method was applied. This method sorts data (here for each cluster separately and in parallel) and then applies pairwise matching only to records in a small neighborhood within the sorted list. Loureiro, Torgo and Soares propose a methodology for the application of hierarchical clustering methods to the task of outlier detection in a transactional data. The aim of their approach is to find clusters that contain rarely occurring transactions that significantly differ from the majority of transactions. They assume, that outliers will be distant from normal observations and thus will be placed in smaller clusters [8]. Khan et al. [7] use k-means clustering for detecting fully or partially duplicated records in a data warehouse. They propose a three step algorithm that first converts all field values into numeric form. Then, K-means clustering is performed to store matching records in the same cluster.

Finally pairs of records within same cluster are compared to find a match. Guo et al. [3] describe a method that uses fuzzy c-means clustering algorithm and the Levenshtein distance (edit distance) to detect similar duplicate data. Like in the approach by Khan, cluster analysis is used first to group similar records and then, within each cluster records are compared (using the Levenstein distance) to find records, that can be deleted. Ciszak proposes to use data mining methods in data cleansing on the level of attribute values. He distinguishes context independent attribute correction, where each attribute is treated separately (i.e. regardless to values of other attributes), and content dependent attribute correction, where the values of an attribute are corrected with respect to values of other attributes. Clustering is used for the first type of correction while association rule mining (to find validation rules) is used for the second type of correction [2].

In all the papers mentioned above, no target (class) attribute is assumed to exist in the data and to guide the cleansing process. So all the methods correspond to an unsupervised learning scheme. Our approach can also be related to the condensed nearest neighbor rule, a method proposed in 1970th to reduce the number of examples used for nearest neighbor classifier (see e.g. [5, 10]).

3 Proposed Methodology

Unlike the work reported in Sect. 2 where the aim of data cleansing was to improve the data quality itself, we are interested in improving the performance of a classifier build from the cleansed data. To identify and remove contradictory examples from data, we propose a methodology consisting of following steps:

1. cluster examples using input attributes only,
2. check the class distribution within each cluster,
3. resolve the contradictions and remove selected examples from data,
4. check the effect of removing contradictions using machine learning algorithm.

Let us give details for each of the steps.

At first we are looking for examples, that are similar in the values of input attributes. To find such examples, we use k-means clustering algorithm with mixed Euclidean distance. Mixed Euclidean distance (MED) is a heterogeneous distance measure capable to handle both numeric and categorial attributes:

$$MED(x, y) = \sqrt{\sum_{A=1}^{m} d_A(x_A, y_A)^2}. \tag{1}$$

The distance between two values x_A and y_A of an attribute A is defined as

$$d_A(x_A, y_A) = \begin{cases} \text{overlap}(x_A, y_A), & \text{if } A \text{ is categorial} \\ \text{normdiff}(x_A, y_A), & \text{if } A \text{ is numeric} \end{cases} \tag{2}$$

Here

$$\text{overlap}(x_A, y_A) = \begin{cases} 0, & \text{if } x_A = y_A \\ 1, & \text{otherwise} \end{cases} \tag{3}$$

and

$$\text{normdiff}(x_A, y_A) = \frac{x_A - y_A}{range_A}, \qquad range_A = max_A - min_A \tag{4}$$

The key parameter of k-means clustering is the required number of clusters k. In our method k becomes crucial because it actually controls the reduction rate (i.e. the ratio of removed examples to all original examples). The value of k can in principle vary between 1 and the number of examples. Obviously, both these bounds need not to be considered. If we set $k = 1$ all examples will be placed into a single cluster and when removing all "weak" contradictions we will end up only with the examples of the majority class. Having as many clusters as is the number of examples we will detect "strong" contradictions, but these duplicities can be also identified just by checking the data.

It is difficult to say what is the correct value for parameter k. Actually more experiments should be carried out for varying k. As a rule of thumb, for larger data we start with $k = 100$ (to cover on average 1 % of examples in each cluster) and then we increase k, for smaller data we start with k set to such a value that a cluster covers on average 1 example and then we decrease k. Our assumption is, that when increasing k, less examples will be identified as weak contradictions and thus less examples will be removed.

To resolve the found contradictions, we check the class distribution within each cluster. If the examples are uniformly distributed, we remove all of them, if the examples are not uniformly distributed, we keep examples of the majority class and remove examples of all other classes.

To check the data quality on the reduced set, we use the C4.5 tree learning algorithm [9]. This algorithm recursively partitions the attribute space by growing a decision tree. Information gain is used to choose the splitting attribute:

$$\text{InfoGain}(A) = -\sum_{j=1}^{S} \frac{s_j}{n} log_2 \frac{s_j}{n} - H(A), \tag{5}$$

where

$$H(A) = \sum_{i=1}^{R} \frac{r_i}{n} \left(-\sum_{j=1}^{S} \frac{a_{ij}}{r_i} \log_2 \frac{a_{ij}}{r_1} \right). \tag{6}$$

In the formulas a_{ij} is the number of examples that have the ith value of the attribute A and the jth value of the target attribute, r_i is the number of examples that have the ith value of the attribute A, s_j is the number of examples that have the jth value of the target attribute and n is the number of all examples.

The algorithm performs binary splits on numeric attributes but creates as many splits as is the number of distinct values for categorial attributes. Post-pruning is used to reduce the created tree (by substituting a subtree by a leaf) and other stopping criteria (e.g. minimal number of examples in a leaf) can also be used to prevent the tree to grow. The created decision tree divides the attribute space into regions (clusters) of examples that are assigned to a same class.

4 Experiments

To evaluate our methodology, semi-automated setting based on the `Weka` data mining system was used. `Weka` is a widely used freely available data mining tool developed at the Waikato University, New Zealand [4].

To cluster data, `SimpleKMeans` procedure from Weka was used with (mixed) Euclidean distance and normalization of numeric attributes. We run `SimpleKMeans` in the "Classes to clusters evaluation" cluster mode. The found clustering was stored and analyzed in MS Excel. `SimpleKMeans` procedure assigns cluster label to every example, so it was possible to create a contingency table that shows the frequencies of different classes within each cluster. Based on the removal strategy, corresponding examples have been marked for deletion; this has been done by adding a binary attribute "remove" to the data. The modified data has been imported back to Weka, marked examples have been filtered out and `J48` procedure (the Weka implementation of C4.5 algorithm) has been run in two modes (with pruning and without pruning). We always used default values of parameters for `J48` (to make the data modifications only to account for changes in classification accuracy) and test the created tree both on the whole (reduced) training set and using 10-fold cross-validation.

We evaluated our approach on five data sets. `Australi`, `German`, `Iris` and `JapCred` are benchmark data from the UCI Machine Learning repository [11], `Re85` is a data set from geology. Table 1 summarizes the basic characteristics of the data sets: number of examples (column "examp.") number of input attributes (column "inp.att.") and number of classes (column "classes"). It also shows the

Table 1 Data description

Data set	Data characteristics				Pruned C4.5 (%)		Unpruned C4.5 (%)	
	examp.	inp.att.	Classes	max.acc (%)	Training	CV	Training	CV
Australi	690	15	2	100.0	90.7	86.1	94.9	81.9
German	1000	20	2	100.0	85.5	70.5	94.0	68.3
Iris	150	4	3	100.0	98.0	96.0	98.0	96.0
JCred	125	10	2	100.0	89.6	80.0	90.4	78.4
Re85	891	6	2	86.3	75.5*	75.5*	82.7	73.0

maximal possible accuracy (column "max.acc") and the accuracy reached (using C4.5 algorithm in both pruning and unpruning mode) on training set and using 10-fold cross-validation (columns "training" and "CV"). The maximal possible accuracy refers to a classifier that correctly classifies all non-contradictory examples and in the case of contradictions it classifies a group of same examples to the majority class of this group, this value is classifier independent and can be computed directly from the data. The asterix (*) denotes a situation, where all examples are classified to the majority class—similar notation applies also to Table 6.

As can be seen from the Table 1, the used data belong to three groups. The `Iris` data represent situations where maximal, training and cross-validation accuracies all are high; there is a low amount of both strong and weak contradictions in this set. The `Australi`, `German` and `JCred` sets represent situations where maximal accuracy is high but both training and cross-validation accuracies are lower; there is a low amount of strong contradictions but higher number of weak contradictions. The `Re85` set represents situations where all accuracies are low; there is a high amount of both strong and weak contradictions in the data. Simultaneously, `Iris` and `JCred` represent small data sets, while the other three files represent larger data sets.

We run C4.5 on the respective data sets in two modes: with and without pruning. We test the created decision tree on training data (to check the quality of the reduced data itself) and using 10-fold cross-validation (to evaluate the generalization ability of the tree learned on the reduced data). The Tables 2, 3, 4, 5 and 6 show for a particular data set the number of clusters (column "k"), the minimal, maximal and average number of examples in the created clusters (columns "min", "max" and "avg"), the percentage of clusters containing contradictions (column "cclust"), the percentage of examples that remain for training after the reduction (column "examples") and the accuracy reached (using C4.5 algorithm in both pruning and unpruning mode) on training set and using 10-fold cross-validation (columns "training" and "CV").

Table 2 Australi data results

k	Clustering			Reduction		Pruned C4.5 (%)		Unpruned C4.5 (%)	
	min	max	avg	cclust (%)	examp (%)	Training	CV	Training	CV
100	1	45	6.9	17.0	97.0	91.9	86.4	95.8	85.5
150	1	23	4.6	10.7	97.5	90.7	86.4	94.9	83.3
200	1	13	3.5	10.0	96.1	92.9	88.1	95.9	86.0

Table 3 German data results

k	Clustering			Reduction		Pruned C4.5 (%)		Unpruned C4.5 (%)	
	min	max	avg	cclust (%)	examp (%)	Training	CV	Training	CV
100	3	29	10	82.0	71.4	95.8	90.6	97.8	90.2
150	1	20	7	72.0	74.5	94.0	85.2	96.8	82.3
200	1	12	5	61.0	75.1	93.3	82.7	96.1	81.4

Table 4 Iris data results

k	Clustering			Reduction		Pruned C4.5 (%)		Unpruned C4.5 (%)	
	min	max	avg	cclust (%)	examp (%)	Training	CV	Training	CV
100	1	4	1.5	2.0	98.0	98.6	95.2	98.6	95.2
75	1	5	2	6.7	95.3	99.3	96.5	99.3	96.5
50	1	9	3	12.0	94.0	100.0	97.1	100.0	97.1
30	1	15	5	13.3	96.0	99.3	96.5	99.3	96.5

Table 5 JCred data results

k	Clustering			Reduction		Pruned C4.5 (%)		Unpruned C4.5 (%)	
	min	max	avg	cclust (%)	examp (%)	Training	CV	Training	CV
100	1	3	1.25	10.0	88.8	91.0	75.7	91.9	73.9
60	1	5	2.1	25.0	80.0	96.0	86.0	97.0	88.0
40	1	9	3.1	50.0	69.6	96.6	87.4	97.7	90.8
30	1	9	4.2	56.7	73.6	96.7	85.9	97.8	90.2

Table 6 Re85 data results

k	Clustering			Reduction		Pruned C4.5 (%)		Unpruned C4.5 (%)	
	min	max	avg	cclust (%)	examp (%)	Training	CV	Training	CV
100	1	44	8.9	66.0	78.1	92.2*	92.2*	98.3	95.8
150	1	28	5.9	53.3	76.9	90.9*	90.9*	97.7	93.3
200	1	25	4.5	42.5	77.8	89.2*	89.2*	97.1	92.9

To summarize the results of our experiments, we can see that:

- the accuracy reached by C4.5 on the reduced data was always higher than the accuracy reached on the original data (the cleansing process really improves the data quality in this sense),
- when the number of clusters is close to the number of examples, the accuracies achieved on the reduced data set are close to the accuracies on the original data (this was observed for the two small sets `Iris` and `JCred` but we believe that this is a general rule),
- when decreasing the number of clusters, the percentage of clusters containing contradictions increases (this was observed for all five sets, we believe this is a general rule),
- our expectation that when the percentage of clusters containing contradictions increases, the percentage of examples that remain for creating a classifier will decrease was not confirmed by the experiments (the expected behavior was observed only for the data `JCred`),

- the smaller the number of clusters, the higher the classification accuracy (this seems to be a side effect of the fact, that when decreasing the number of clusters, more contradictions are removed),
- higher number of weak contradictions, i.e. greater difference between maximal accuracy and accuracy reached by C4.5 for the original data (see Table 1) results in higher reduction rate (smaller percentage of examples that remain for creating a classifier).

Anyway, more experiments are necessary to make a general claims from these findings.

5 Conclusions and Further Work

Our work deals with the question how to improve the quality of data by removing contradictory examples from the training data. We propose a method based on clustering that allows us to identify close examples that belong to different classes. Our first experimental results allow us to formulate some hypotheses about the mutual relationships between number of clusters, amount of contradictions, reduction rate and accuracy; nevertheless more experiments are necessary to make some general conclusions. More sophisticated reduction strategies can be used as well. So far we removed all examples from a cluster only if they are uniformly distributed among the classes (and keep examples of the majority class only otherwise). But we can consider to remove all examples from a cluster if they are "almost" uniformly distributed among the classes.

Acknowledgments This paper was prepared with the support of Institutional funds for a long-term development of science and research at the Faculty of Informatics and Statistics of the University of Economics, Prague.

References

1. Berka, P.: Learning compositional decision rules using the KEX algorithm. Intell. Data Anal. **16**(4), 665–681 (2012)
2. Ciszak, L.: Application of clustering and association methods in data cleaning. In: IEEE ICCSIT 2008, pp. 97–103. Singapore (2008)
3. Guo, L., Wang, W., Chen, F., Tang, X., Wang, W.: A similar duplicate data detection method based on fuzzy clustering for topology formation. Electr. Rev. **88**(01b), 26–31 (2012)
4. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. ACM SIGKDD Explor. Newsl. **11**(1), 10–18 (2009)
5. Hart, P.E.: The condensed nearest-neighbor rule. IEEE Trans. Inf. Theory **14**(3), 515–516 (1968)
6. Hernández, M.A., Stolfo, S.J.: The merge/purge problem for large databases. ACM SIGMOD Rec. **24**(2), 127–138 (1995)

7. Khan, B., Rauf, A., Javed, H., Khusro, S.: Removing fully and partially duplicated records through k-means clustering. Int. J. Eng. Technol. **4**(6), 750–754 (2012)
8. Loureiro, A., Torgo, L., Soares, C.: Outlier detection using clustering methods: a data cleaning application. In: Proceedings of Data Mining for Business Workshop (2004)
9. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Francisco (1993)
10. Tomek, I.: Two modifications of CNN. IEEE Trans. Syst. Man Cybern. **6**(11), 769–772 (1976)
11. UCI: Machine Learning Repository (2013). http://archive.ics.uci.edu/ml/

Estimation of the Joint Spectral Radius

Adam Czornik, Piotr Jurgaś and Michał Niezabitowski

Abstract The joint spectral radius of a set of matrices is a generalization of the concept of spectral radius of a matrix. Such notation has many applications in the computer science, and more generally in applied mathematics. It has been used, for example in graph theory, control theory, capacity of codes, continuity of wavelets, overlap-free words, trackable graphs. It is impossible to provide analytic formulae for this quantity and therefore any estimation are highly desired. The main result of this paper is to provide an estimation of the joint spectral radius in the terms of matrices norms and spectral radii.

Keywords Joint spectral radius · Graph theory · Capacity of codes · Continuity of wavelets · Estimation · Overlap-free words · Trackable graphs

1 Introduction

Let $\Sigma = \{A_i : i \in I\}$ be a bounded set or real s-by-$s$matrices. Denote by $\|\cdot\|$ the Euclidean norm and the generated operator norm. For a square matrix A by $\rho(A)$ we will denote the spectral radius of A, i.e. the greatest value of modulus of eigenvalues of A. If $\rho(A) < 1$ then the matrix A will be called asymptotically stable. Moreover, D is the set of all sequences of elements of I, i.e.

$$D = \{d = (d(0), d(1), \ldots) : d(i) \in I\}. \tag{1}$$

By a discrete linear inclusion $DLI(\Sigma)$, which will by denoted by

$$x(j+1) \in A_{d(j)}x(j), \quad j = 0, 1, 2 \ldots, \tag{2}$$

A. Czornik (✉) · P. Jurgaś · M. Niezabitowski
Institute of Automatic Control, Silesian University of Technology, Gliwice, Poland
e-mail: adam.czornik@polsl.pl

M. Niezabitowski
e-mail: michal.niezabitowski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_34

we will understand the set of all sequences $(x(j))_{j\in\mathbb{N}}$, $x(j) \in \mathbb{R}^s$ such that

$$x(j+1) = A_{d(j)}(x(j)), \quad j = 0, 1, 2, \ldots \tag{3}$$

for certain $d \in D$. Each such a sequence $(x(j))_{j\in\mathbb{N}}$ will be called trajectory of $DLI\,(\Sigma)$ and $x(0)$ will be called initial value of this trajectory. For $m \geq 1$, Σ^m is the set of all products of matrices in Σ of length m, i.e.

$$\Sigma^m = \{A_1 A_2 \ldots A_m : A_i \in \Sigma, i = 1, \ldots, m\}. \tag{4}$$

Set $\overline{\alpha}_m = \sup_{A\in\Sigma^m} \|A\|$; $\underline{\alpha}_m = \inf_{A\in\Sigma^m} \|A\|$, and define:

- the joint spectral subradius

$$\widehat{\rho}_*(\Sigma) = \inf_{m\geq 1} \left(\underline{\alpha}_m\right)^{\frac{1}{m}}; \tag{5}$$

- the joint spectral radius

$$\widehat{\rho}(\Sigma) = \inf_{m\geq 1} \left(\overline{\alpha}_m\right)^{\frac{1}{m}}. \tag{6}$$

The concept of the joint spectral radius was first defined in 1960 in [52]. The concept of joint spectral subradius was introduced in [20] in finite-dimensional case. In recent years notions of joint spectral radius and subradius have found applications in a large number of engineering fields and are still a topic of active research: control theory (hybrid systems) [2, 32]; curve design (subdivision schemes) [25]; autonomous agents (consensus rate) [8]; wavelets (continuity of wavelets) and refinement equations [9, 22, 24]; number theory (asymptotics of the partition function) [47, 51]; coding theory (constrained codes) [5, 40]; sensor networks (trackability) [10]; combinatorics of words (overlap-free words) [3, 33]; probabilistic automata [4]; probability theory [48].

In control theory very important role is played by the concept of stability. The formal definition of stability of $DLI\,(\Sigma)$ is as follows.

Definition 1 $DLI\,(\Sigma)$ is called absolute stable if and only if for all trajectory $(x(j))_{j\in N}$ we have

$$\lim_{j\to\infty} x\,(j) = 0. \tag{7}$$

The stability of finite-dimensional version of (3) has been widely discussed in the literature [2, 31, 42] and the references therein. This concept may be defined in several ways and even in the finite-dimensional case these different concepts are not equivalent. The joint spectral radius and subradius are closely connected to Lyapunov exponents of (3), see [13, 15–19], and Bohl exponents, see [1, 11, 14, 21, 43, 44]. The following theorem contains the relation between the joint spectral radius and absolute stability of $DLI\,(\Sigma)$.

Theorem 1 ([7]) *DLI* (Σ) *is absolute stable if and only if* $\widehat{\rho}(\Sigma) < 1$.

Now we briefly survey how these quantities can be computed or approximated. The inequalities:

$$\sup\left\{\rho\left(\prod_{i=1}^{k} A_i\right) : A_i \in \Sigma \quad \text{for } 1 \le i \le k\right\} \le \rho(\Sigma) \le$$

$$\sup\left\{\left\|\prod_{i=1}^{k} A_i\right\| : A_i \in \Sigma \quad \text{for } 1 \le i \le k\right\}. \tag{8}$$

proved in Lemma 3.1 of [23] can be used to derive algorithms which compute arbitrarily precise approximations for generalized spectral radius (see, for example, [27] for one of such algorithms).

To obtain such algorithms two approaches are applied. The first one is to try to construct the so-called extremal norm if it exists. The sufficient and necessary condition for existence is known and it is the non-defectiveness property [28]. Different methods of constructing the extremal norm are presented in [29, 30, 36, 41, 45, 54]. The second approach uses the invariant cone of the considered set of matrices, when such a cone exists. In [46, 49] iterative algorithms, build on this idea, are presented. Also in [50] this idea is explored to obtain a new conic programming method. In general, the existence of an invariant cone is very restrictive condition and excludes many interesting cases in real applications.

From the stability of discrete linear inclusion point of view the following question is very important: is the spectral radius greater or smaller than 1? To this day it is so far unknown whether this problem is decidable. In [6] the problem if spectral radius is less or equal to 1 was considered. Even today it is unknown whether this problem is algorithmically solvable (see [37] for a discussion of this issue and for a description of its connection to the finiteness conjecture, see also the discussion in [31]). A negative result of this discussion is given by Kozyakin, who shows in [35] that the set of pairs of 2-by-2 matrices that have a joint spectral radius less than one, is not semialgebraic.

The first result (Theorem 1) presented in [53] shows that, unless $P = NP$, approximating algorithms for generalized spectral radius can not possibly run in polynomial time. More precisely, it shows that, unless $P = NP$, there is no algorithm that can compute generalized spectral radius with a relative error bounded by $\varepsilon > 0$, in polynomial time, that will be a function of numbers of the elements in the Σ, size of matrices Σ and ε. As a corollary, the authors of the above-mentioned publication show that it is *NP*-hard to decide if all possible products of two given matrices tend to zero. The situation for the lower spectral radius is somewhat different from that of the joint spectral radius. Possibility of calculation of the upper bounds for generalized spectral subradius for the case, where Σ consists of nonnegative matrices is given in [53]. In publication [39] we can find an analytic solution to the case when Σ consists of 2-by-2 matrices, one of which is singular.

2 Main Results

It is well-known that the condition $\rho(A_i) < 1$ for $i \in I$ is only the necessary but not sufficient condition for absolute stability of $DLI(\Sigma)$ [12]. In our further considerations we will show that this condition together with some bounds on $\|A_i - A_j\|$ for $i, j \in I$ implies the absolute stability of $DLI(\Sigma)$.

Let us fix $d \in D$ and consider the time-varying linear system

$$x(n+1) = A_{d(n)} x(n). \tag{9}$$

This system will be called asymptotically stable if all solutions tend to zero.

To prove the main result of our paper we will need the following facts from the literature.

Lemma 1 ([34]) *If for system (9) there exists a function $V : \mathbb{N} \times \mathbb{R}^s \to [0, \infty)$ such that*

1.

$$\|x\|^2 \le V(n, x) \le C_1 \|x\|^2 \tag{10}$$

2.

$$V(n+1, x(n+1)) - V(n, x(n)) \le -C_2 \|x(n)\|^2 \tag{11}$$

for all $x \in \mathbb{R}^s$, $n \in \mathbb{N}$ and certain positive constants C_1, C_2, then (9) is asymptotically stable.

The function V from the above lemma is called the Lyapunov function.

Lemma 2 ([38]) *For a matrix $A \in \mathbb{R}^{s\times s}$ the following conditions are equivalent:*

1. *matrix $A \in \mathbb{R}^{s\times s}$ has all eigenvalues in the open unit circle;*
2. *for each positive definite matrix $Q \in \mathbb{R}^{s\times s}$ there exists a positive definite matrix $P \in \mathbb{R}^{s\times s}$ such that the following Lyapunov equation is satisfied*

$$A^T P A - P = -Q; \tag{12}$$

3. *there are positive definite matrices P, $Q \in \mathbb{R}^{s\times s}$ such that (12) is satisfied.*

Moreover if $A \in \mathbb{R}^{s\times s}$ has all eigenvalues in the unit circle then the solution of (12) is given by

$$P = \sum_{k=0}^{\infty} \left(A^T\right)^k Q A^k. \tag{13}$$

Suppose that for certain $\varepsilon > 0$ we have

$$\sup_{i \in I} \rho(A_i) < 1 - \varepsilon. \tag{14}$$

Moreover denote $a = \sup_{i \in I} \|A_i\|$.

Lemma 3 ([26]) *If the condition (14) is satisfied then*

$$\left\| (A_i)^N \right\| \leq m\delta^N \tag{15}$$

for all $i \in I$ and $N \in \mathbb{N}$, where $\delta = 1 - \varepsilon$ and

$$m = (1 - \varepsilon) \frac{(1 - \varepsilon + a)^{s-1}}{\varepsilon^s}. \tag{16}$$

The next lemma provides bounds for the solutions of the Lyapunov equations corresponding to all matrices in Σ.

Lemma 4 *Suppose that (14) is satisfied, denote $d = \sup_{i,j \in I} \|A_i - A_j\|$ and let P_i be the unique positive definite solution of $A_i^T P A_i - P = -I$, then*

$$\sup_{i,j \in I} \|P_i - P_j\| \leq 2da \left(1 + \frac{m^2\delta^2}{1 - \delta^2}\right)^2. \tag{17}$$

Proof By the definition of P_i and P_j we have

$$A_i^T \left(P_j - P_i\right) A_i - \left(P_j - P_i\right) = \left(A_i - A_j\right)^T P_j A_i + A_j^T P_j \left(A_i - A_j\right). \tag{18}$$

Denoting the right hand side of the last equation by $M_{i,j}$ we have

$$\|M_{i,j}\| \leq 2 \|A_j\| \|P_j\| \|A_i - A_j\|. \tag{19}$$

Using (13) and (15) we may estimate P_j as follows

$$\|P_j\| \leq \sum_{k=0}^{\infty} \left\| \left(A_j^T\right)^k A_j^k \right\| \leq 1 + m^2 \sum_{k=1}^{\infty} \left(\delta^{2k}\right) = 1 + \frac{m^2\delta^2}{1 - \delta^2}. \tag{20}$$

Taking into account (20) we obtain from (19) that

$$\|M_{i,j}\| \leq 2ad \left(1 + \frac{m^2\delta^2}{1 - \delta^2}\right). \tag{21}$$

Applying formula (13) to (18) we get

$$P_j - P_i = M_{i,j} + \sum_{k=1}^{\infty} \left(A_i^T\right)^k M_{i,j} A_i \tag{22}$$

and therefore

$$\left\| P_j - P_i \right\| \leq 2ad \left(1 + \frac{m^2\delta^2}{1-\delta^2}\right) \left[1 + \sum_{k=1}^{\infty} m^2\delta^{2k}\right] =$$

$$2ad \left(1 + \frac{m^2\delta^2}{1-\delta^2}\right)^2. \tag{23}$$

The last inequality ends the proof of the lemma.

The next theorem contains the main result of our work.

Theorem 2 *Suppose that the condition (14) is satisfied and the numbers* $a,\ d,\ m,\ \delta$ *are such that*

$$2ad \left(1 + \frac{m^2\delta^2}{1-\delta^2}\right)^2 < 1, \tag{24}$$

then $\widehat{\rho}(\Sigma) < 1$.

Proof Denote the left hand side of inequality (24) by $1-\eta$. According to Theorem 1 is enough to show that $DLI(\Sigma)$ is absolutely stable. Let us fix a sequence $d \in D$ and consider the system (3). We will show, that the function

$$V(n, x) = x^T P_{d(n)} x \tag{25}$$

satisfies the assumptions of Lemma 1, where $P_{d(n)}$ is the unique solution of the equation

$$A_{d(n-1)}^T P_{d(n)} A_{d(n-1)} - P_{d(n)} = -I. \tag{26}$$

From the definition of P_i and the inequality (20) it is clear that

$$\|x\|^2 \leq V(n, x) \leq \left(1 + \frac{m^2\delta^2}{1-\delta^2}\right) \|x\|^2. \tag{27}$$

Moreover

$$V(n+1, x(n+1)) - V(n, x(n)) =$$

$$x^T(n) \left(P_{d(n+1)} - P_{d(n)} - I\right) x(n) \leq -\eta \|x(n)\|^2. \tag{28}$$

So in fact the function $V(n, x)$ satisfies the assumptions of Lemma 1 and the system (3) is absolutely stable what ends the proof.

Example 1 Consider the following set

$$\Sigma = \left\{ A_i = \begin{bmatrix} -q(i) & q(i) \\ 1 & 0 \end{bmatrix} : i = 1, 2, \ldots \right\}, \tag{29}$$

where

$$q(i) = \frac{\sin(\ln(\ln(i+12)))}{r}, \tag{30}$$

where $r > 0$. Using Theorem 2 we will find the values of r such that $\widehat{\rho}(\Sigma) < 1$. We have

$$\|A_i\| \leq 2 \max |q(i)| \leq \frac{2}{r}, \tag{31}$$

therefore

$$a \leq \frac{2}{r}. \tag{32}$$

Moreover it is easy to estimate that

$$\rho(A_i) \leq \frac{1}{2r} + \frac{1}{2}\sqrt{\frac{1}{r^2} + \frac{4}{r}}. \tag{33}$$

Therefore

$$\delta \geq \frac{1}{2r} + \frac{1}{2}\sqrt{\frac{1}{r^2} + \frac{4}{r}}. \tag{34}$$

Finally, let us estimate d. Suppose that $i > j$. Then we have

$$\left\| A_i - A_j \right\| \leq |q(i) - q(j)|. \tag{35}$$

According to Lagrange theorem

$$|q(i) - q(j)| = \left| f'(c) \right|, \tag{36}$$

where $c \in (\min\{i, j\}, \max\{i, j\})$ and

$$f(x) = \frac{\sin(\ln(\ln(x+12)))}{r}. \tag{37}$$

Since

$$\frac{d}{dx} \frac{\sin(\ln(\ln(x+12)))}{r} =$$

$$\frac{1}{\ln(x+12)}\frac{\cos(\ln(\ln(x+12)))}{12r+rx}, \tag{38}$$

then

$$\left|f'(c)\right| \le \frac{1}{(12r+rc)\ln(c+12)} \le \frac{1}{12r\ln 12}. \tag{39}$$

Combining (35) with (39) we conclude

$$d \le \frac{1}{12r\ln 12}. \tag{40}$$

Using (32), (34) and (40) we may estimate the right hand side of (24) as follows

$$2ad\left(1+\frac{m^2\delta^2}{1-\delta^2}\right)^2 \le$$

$$\frac{4}{r} - \frac{16r+56rt+\frac{72}{r}t+192t+\frac{336}{r}+\frac{72}{r^2}+296}{12r^3t-8r^2t-12r-10r^4t+8r^5t+2t+\frac{2}{r}+4r^2+10r^3-12r^4-4r^5+4},$$

where $t=\sqrt{\frac{4}{r}+\frac{1}{r^2}}$. Finally, by numerical analysis we conclude that $\widehat{\rho}(\Sigma)<1$ for $r>6.4058$.

3 Conclusion

In the paper we have consider two generalizations of spectral radius of a matrix: the joint spectral radius and the joint spectral subradius. These notions deal with the set of matrices and when the set reduces to the one element set they coincide with the definition of the spectral radius. These quantities have found many applications, among others in stability of switched systems. For the above-mentioned notions we present upper bounds, which are expressed in the terms of spectral radii of the matrices and the diameter of the considered set. Our results give a new sufficient condition for stability of discrete linear inclusions. Finally, we have presented a numerical example.

Acknowledgments The research presented here were done by the authors as parts of the projects funded by the National Science Centre granted according to decisions DEC-2012/07/B/ST7/01404, DEC-2012/05/B/ST7/00065 and DEC-2012/07/N/ST7/03236, respectively. The calculations were performed with the use of IT infrastructure of GeCONiI Upper Silesian Centre for Computational Science and Engineering (NCBiR grant no POIG.02.03.01-24-099/13).

References

1. Babiarz, A., Czornik, A., Niezabitowski, M.: On the number of upper Bohl exponents for diagonal discrete time-varying linear system. J. Math. Anal. Appl. **429**(1), 337–353 (2015)
2. Barabanov, N.: Lyapunov indicators of discrete inclusions. Autom. Remote Control **49**, 152–157 (1988)
3. Berstel, J.: Growth of repetition-free words-a review. Theor. Comput. Sci. **340**(2), 280–290 (2005)
4. Blondel, V., Canterini, V.: Undecidable problems for probabilistic automata of fixed dimension. Theory Comput. Syst. **36**(3), 231–245 (2003)
5. Blondel, V., Jungers, R., Protasov, V.: On the complexity of computing the capacity of codes that avoid forbidden difference patterns. IEEE Trans. Inf. Theory **52**(11), 5122–5127 (2006)
6. Blondel, V., Tsitsiklis, J.: Boundedness of all products of a pair of matrices is undecidable. Syst. Control Lett. **41**(2), 135–140 (2000)
7. Brayton, R., Tong, C.: Constructive stability and asymptotic stability of dynamical systems. IEEE Trans. Circuits Syst. **27**(11), 1121–1130 (1980)
8. Brooks, R., Friedlander, D., Koch, J., Phoha, S.: Tracking multiple targets with selforganizing distributed ground sensors. J. Parallel Distrib. Comput. **64**(7), 874–884 (2004)
9. Collela, D., Heil, D.: Characterization of scaling functions: continuous solutions. SIAM J. Matrix Anal. Appl. **15**, 496–518 (1994)
10. Crespi, V., Cybenko, G., Jiang, G.: The theory of trackability with applications to sensor networks. ACM Trans. Sens. Netw. **4**(3), 1–42 (2008)
11. Czornik, A.: The relations between the senior upper general exponent and the upper Bohl exponents. In: MMAR 2014, pp. 897–902. Miedzyzdroje, Poland (2014)
12. Czornik, A., Jurgas, P.: Falseness of the finiteness property of the spectral subradius. Int. J. Appl. Math. Comput. Sci. **17**(2), 173–178 (2007)
13. Czornik, A., Jurgas, P.: Set of possible values of maximal Lyapunov exponents of discrete time-varying linear system. Automatica **44**(2), 580–583 (2008)
14. Czornik, A., Klamka, J., Niezabitowski, M.: About the number of the lower Bohl exponents of diagonal discrete linear time-varying systems. In: ICCA 2014, pp. 461–466. Taichung, China (2014)
15. Czornik, A., Nawrat, A., Niezabitowski, M.: On the Lyapunov exponents of a class of the second order discrete time linear systems with bounded perturbations. Dyn. Syst. Int. J. **28**(4), 473–483 (2013)
16. Czornik, A., Nawrat, A., Niezabitowski, M.: On the stability of lyapunov exponents of discrete linear systems. In: ECC 2013, pp. 2210–2213. Zurich, Switzerland (2013)
17. Czornik, A., Niezabitowski, M.: Lyapunov exponents for systems with unbounded coefficients. Dyn. Syst. Int. J. **28**(2), 140–153 (2013)
18. Czornik, A., Niezabitowski, M.: Lyapunov exponents for systems with unbounded coefficients. Dyn. Syst. Int. J. **28**(2), 299–299 (2013)
19. Czornik, A., Niezabitowski, M.: On the spectrum of discrete time-varying linear systems. Nonlinear Anal. Hybrid Syst. **9**, 27–41 (2013)
20. Czornik, A.: On the generalized spectral subradius. Linear Algebra Appl. **407**, 242–248 (2005)
21. Czornik, A., Niezabitowski, M.: Alternative formulae for lower general exponent of discrete linear time-varying systems. J. Frankl. Inst. **352**(1), 399–419 (2015)
22. Daubechies, I.: Orthonormal bases of compactly supported wavelets. Commun. Pure Appl. Math. **41**(7), 909–996 (1988)
23. Daubechies, I., Lagarias, J.: Sets of matrices all infinite products of which converge. Linear Algebra Appl. **161**, 227–263 (1992)
24. Daubechies, I., Lagarias, J.: Two-scale difference equations ii. local regularity, infinite products of matrices and fractals. SIAM J. Math. Anal. **23**(4), 1031–1079 (1992)
25. Derfel, G., Dyn, N., Levin, D.: Generalized refinement equations and subdivision processes. J. Approx. Theory **80**(2), 272–297 (1995)

26. Desoer, C.: Slowly varying discrete system xi+1 = aixi. Electron. Lett. **6**(11), 339–340 (1970)
27. Gripenberg, G.: Computing the joint spectral radius. Linear Algebra Appl. **234**, 43–60 (1996)
28. Guglielmi, N., Wirth, F., Zennaro, M.: Complex polytope extremality results for families of matrices. SIAM J. Matrix Anal. Appl. **27**(3), 721–743 (2005)
29. Guglielmi, N., Zennaro, M.: Finding extremal complex polytope norms for families of real matrices. SIAM J. Matrix Anal. Appl. **31**(2), 602–620 (2009)
30. Guglielmi, N., Zennaro, M.: An algorithm for finding extremal polytope norms of matrix families. Linear Algebra Appl. **428**(10), 2265–2282 (2008)
31. Gurvits, L.: Stability of discrete linear inclusion. Linear Algebra Appl. **231**, 47–85 (1995)
32. Jungers, R., Protasov, V.: Weak stability of switching dynamical systems and fast computation of the p-radius of matrices. In: CDC 2010, pp. 7328–7333. Atlanta, USA (2010)
33. Jungers, R., Protasov, V., Blondel, V.: Computing the growth of the number of overlap-free words with spectra of matrices. In: Laber, E., Bornstein, C., Nogueira, L., Faria, L. (eds.) LATIN 2008: Theoretical Informatics. LNCS, vol. 4957, pp. 84–93. Springer, Berlin (2008)
34. Khalil, H.: Nonlinear Systems. Prentice Hall, New York (2001)
35. Kozyakin, V.: Algebraic unsolvability of problem of absolute stability of desynchronized systems. Avtomatika i Telemekhanika **51**(6), 754–759 (1990)
36. Kozyakin, V.: Iterative building of barabanov norms and computation of the joint spectral radius for matrix sets. Discrete Contin. Dyn. Syst. Ser. B **14**(1), 143–158 (2010)
37. Lagarias, J., Wang, Y.: The finiteness conjecture for the generalized spectral radius of a set of matrices. Linear Algebra Appl. **214**, 17–42 (1995)
38. Lancaster, P., Rodman, L.: Algebraic Riccati Equations. Clarendon Press, New York (1995)
39. Lima, R., Rahibe, M.: Exact Lyapunov exponent for infinite products of random matrices. J. Phys. Math. Gen. **27**(10), 3427–3437 (1994)
40. Lind, D., Marcus, B.: An Introduction to Symbolic Dynamics and Coding. Cambridge University Press, Cambridge (1995)
41. Maesumi, M.: Optimal norms and the computation of joint spectral radius of matrices. Linear Algebra Appl. **428**(10), 2324–2338 (2008)
42. Molchanov, A., Pyatnitskiy, Y.: Criteria of asymptotic stability of differential and difference inclusions encountered in control theory. Syst. Control Lett. **13**(1), 59–64 (1989)
43. Niezabitowski, M.: About the properties of the upper Bohl exponents of diagonal discrete linear time-varying systems. In: MMAR 2014, pp. 880–884. Miedzyzdroje, Poland (2014)
44. Niezabitowski, M.: On the Bohl and general exponents of the discrete time-varying linear system. In: Sivasundaram, S (ed.) ICNPAA 2014, vol. 1637, pp. 744–749 (2014)
45. Parrilo, P., Jadbabaie, A.: Approximation of the joint spectral radius using sum of squares. Linear Algebra Appl. **428**(10), 2385–2402 (2008)
46. Protasov, V.: The joint spectral radius and invariant sets of linear operators. Fundamental'naya i Prikladnaya Matematika **2**(1), 205–231 (1996)
47. Protasov, V.: Asymptotic behaviour of the partition function. Matematicheskii Sbornik **191**(3), 65–98 (2000)
48. Protasov, V.: Refinement equations with nonnegative coefficients. J. Fourier Anal. Appl. **6**(1), 55–78 (2000)
49. Protasov, V.: The geometric approach for computing the joint spectral radius. In: CDC-ECC 2005, pp. 3001–3006. Seville, Spain (2005)
50. Protasov, V., Jungers, R., Blondel, V.: Joint spectral characteristics of matrices: a conic programming approach. SIAM J. Matrix Anal. Appl. **31**(4), 2146–2162 (2010)
51. Protasov, V.: On the asymptotics of the binary partition function. Matematicheskie Zametki **76**(1), 151–156 (2004)
52. Rota, G., Strang, G.: A note on the joint spectral radius. Proc. Neth. Acad. **22**, 379–381 (1960)
53. Tsitsiklis, J., Blondel, V.: The Lyapunov exponent and joint spectral radius of pairs of matrices are hard—when not impossible—to compute and to approximate. Math. Control Signals Syst. **10**(1), 31–40 (1997)
54. Wirth, F.: The generalized spectral radius and extremal norms. Linear Algebra Appl. **342**, 17–40 (2002)

AspectAnalyzer—Distributed System for Bi-clustering Analysis

Pawel Foszner and Andrzej Polański

Abstract In this study we describe a software package accessible as standalone application for performing various bi-clustering methods. System has already implemented 5 algorithms and there is a possibility to add further. In addition to the algorithms from literature, the software includes an original method developed by the authors. System is fully distributed, and has a module to create a computer farm consisting of the computational equipment on which this software were installed. It also allows for creating highly efficient computational environment that will solve wide range of problems related to bi-clustering. The software has been released to the public on the Internet, along with extensive service organized in the form of a blog. At the address http://aspectanalyzer.foszner.pl a ready to use installer has been posted, along with a complete user manual. In addition, the portal allows reporting bugs, new features, and questions about the software. All software is provided free of charge and will include a complete, ready-to-run package.

Keywords Bi-clustering · Clustering · Data mining · Machine learning

1 Introduction

Bi-clustering is a technique for searching subsets of attributes along one dimension that also reveals similarity for a subset of attributes along the second dimension, in two-dimensional datasets.

There are many motivations for developing bi-clustering software environments. They come from different areas of research, text analysis, pattern recognition, data mining, bioinformatics. In bioinformatics when algorithms for interpreting results of experiments with DNA microarrays are developed, bi-clustering is often used for grouping gene expression data according to two dimensions simultaneously. The first

P. Foszner (✉) · A. Polański
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: pawel.foszner@polsl.pl

A. Polański
e-mail: andrzej.polanski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_35

dimension is given by different genes and the second one is given by different experimental conditions, from analyses of genomic data, protein interactions, analysis and annotations of gene signatures.

One can distinguish multiple classification of bi-clusters regarding to its structure or position in data matrix. Each case may need a different approach. The task of selecting the appropriate method requires a very good understanding of the data to be analysed. A very difficult task is choosing the appropriate number of bi-clusters (which is often the input parameter for many bi-clustering algorithms). The algorithm of processing data in bi-clustering may look as presented on Fig. 1. There is a large number of data structures, and very often algorithms from literature specialized only in specific one.

We are never able to say with absolute certainty that we have data containing bi-clusters of a certain structure. Therefore, the process of obtaining bi-clusters is always an iterative process. Each iteration includes activities related to the selection of parameters, and very often an attempt to determine the number of bi-clusters. Each of these steps is usually performed manually by the scientist responsible on organizing the computational experiment and on interpreting the results of data analysis.

The idea of AspectAnalyzer software is to implement all major literature algorithms for data bi-clustering, features for decision support, parameters tuning and wrap it in a user-friendly interface. The strategy of the performed research was oriented towards simplifying the analysis of bi-clustering to a pipeline as simple as possible: providing data on the input and getting the results on the output. The role of the user in this system is limited to the loading on the input data. However, it may also adjust the parameters used in the analysis.

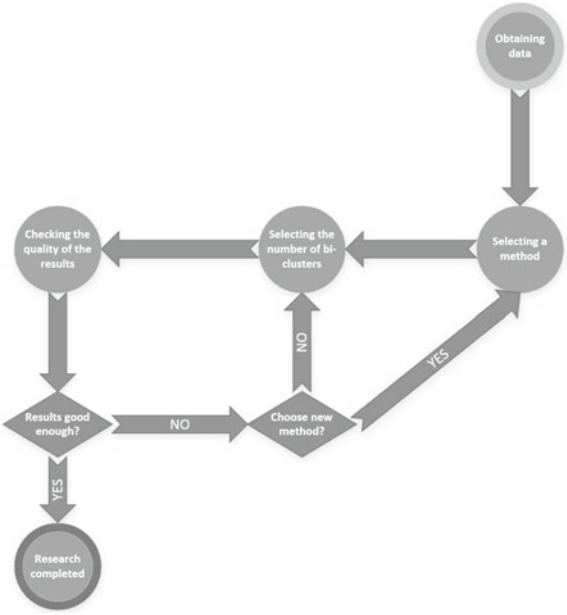


Fig. 1 Bi-clustering analysis sample workflow

2 Related Work

There is a few papers describing software packages with bi-clustering algorithms. Almost all were created on the occasion of a comparative analysis of the implemented algorithms. The largest of these is the BiBench package by Kumal Eren et al. [2]. They implement a package for python linux environment that supports 12 algorithms from the literature. Another very good and ready to use software is R-package biclust by Sebastian Kaiser et al. [6] witch contains 7 methods.

3 Methods

Notation was taken from the paper by Madeira and Oliveira [9], where bi-cluster is defined by a subset of rows and subset of columns from data matrix. Given the data matrix V with set of rows (X), and set of columns (Y), a bi-cluster (B) is defined by a sub-matrix (I,J), where I is a subset of X, and J is a subset of Y.

$$V = (X, Y) \tag{1}$$

$$V = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \vdots & \vdots \\ a_{M1} & a_{M2} & \cdots & a_{MN} \end{bmatrix} X_i = \begin{bmatrix} a_{i1}\, a_{i2} \cdots a_{iN} \end{bmatrix}, Y = \begin{bmatrix} a_{1j} \\ a_{2j} \\ \vdots \\ a_{Mj} \end{bmatrix} \tag{2}$$

Single bi-clustering experiment (R) outputs K bi-clusters, where K is a number which, depending on the algorithm used, can be a parameter given by the user, or the number formed as a result of executing the selected method.

3.1 Literature Methods

At the moment AspectAnalyzer has 7 different methods available. 6 algorithms from literature and 1 ensemble method developed by author. The system architecture joins these algorithms in the form of modules. It is open to the new algorithms, and the following describes those that work at the present time.

Non-negative Matrix Factorization. A very wide range of algorithms are algorithms based on data matrix decomposition. In such methods data matrix (A) is factorized into (usually) much smaller matrices. Such a distribution, because of the much smaller matrices is much easier to analyse, and the obtained matrices reveal previously hidden features. These algorithms are often called NMF algorithms. NMF stands for non-negative matrix factorization. Two efficient algorithms

were introduced by Seung and Lee [7]. First minimize conventional least square error distance function and second generalized Kullback-Leibler divergence. Third and last from this group is algorithm that slightly modify the second approach. Author [10] introduce smoothing matrix for achieving a high degree of sparseness, and better interpretability of the results. Data matrix in this techniques is factorized into (usually) two smaller matrices:

$$A \approx WH \tag{3}$$

Finding the exact solution is computationally very difficult task. Instead, the existing solutions focus on finding local extrema of the function describing the fit of the model to the data. AspectAnalyzer implements five algorithms based on non-negative matrix factorization:

- PLSA witch stands for Probabilistic Latent Semantic Analysis. Introduced by Thomas Hoffman [4], and based on maximizing log-likelihood function. For this purpose author use Expectation-Maximization algorithm [1].
- Based on minimization of Least Square Error distance function

$$\|A - WH\|^2 = \sum_{ij}(A_{ij} - WH_{ij})^2 \tag{4}$$

- Based on minimization of Kullback-Leibler divergence

$$D(A \,||\, WH) = \sum_{ij}(A_{ij} log \frac{A_{ij}}{WH_{ij}} - A_{ij} + WH_{ij}) \tag{5}$$

- Based on minimization of non-smooth Kullback-Leibler divergence.

$$D(A \,||\, WSH) = \sum_{ij}(A_{ij} log \frac{A_{ij}}{WSH_{ij}} - A_{ij} + WSH_{ij}) \tag{6}$$

$$S = (1 - \theta)\mathbf{I} + \frac{\theta}{q}\mathbf{1}\mathbf{1}^T \tag{7}$$

- FABIA [3] which is like PLSA variation on Expectation-Maximization algorithm.

Graph algorithms QUBIC stands for QUalitative BIClustering algorithm. It was proposed by Guojun Li et al. [8] as very efficient algorithm for analysis of gene expression data. Authors proposed weighted graph representation of discretized expression data. The expression levels are discretized to the ranks. Their number is determined by the user through the parameters of the algorithm. Number of ranks is essential and strongly affects the results. The algorithm allows two types of ranks. The positive (for up-regulating genes) and negative sign (for down-regulating genes). The vertices of the graph represent genes. The edges between them have weight to reflect the number of conditions for which they have the same rank. After building

the graph bi-clusters are find one-by-one. Starting from single heaviest and unused edge as seed, algorithm iteratively add additional edges until its violates pre-specified consistency level.

3.2 Consensus Method

For the purpose of ensemble methods there is need for algorithm to finding corresponding bi-clusters between different results. Jaccard Index can be applied to comparison of single bi-clusters. When combined with the Hungarian algorithm (also known as Munkres algorithm) can be expanded to use for comparing different results or methods. This quality index called "consensus score" was taken from paper by S. Hochreiter et al. 2010 [3]. Algorithm is as follows:

- Compute similarities between obtained bi-clusters and known bi-clusters from original set (assuming that the bi-clusters are known), or similarities between clusters from first and second result sets.
- Using Munkers algorithm assign bi-clusters of the one set to the bi-clusters from the other one.
- Divide the sum of similarities of the assigned bi-clusters as emphasized number of bi-clusters of the larger set.

Such approach finds assignments witch maximize following function S:

$$S(R_1, R_2) = \sum_{l=1}^{K} S_{Jac}(B_l^1, B_{l'}^2) \tag{8}$$

where R_1 and R_2 are two independent bi-clustering experiments and B_l^1 and $B_{l'}^2$ are pairs of bi-clusters such that B_l^1 is l'th bi-cluster from result R_1 and $B_{l'}^2$ is bi-cluster corresponding to it from result R_2.

This algorithm described above applies to matching bi-clusters of two results. In the general case, there is need to be able to match the bi-clusters between N results. Finding an optimal solution in matching N results comes down to the analysis of in N-dimensional space. But it can be safely assumed that bi-clustering experiments which are carried out on the same data with the similar number of bi-clusters should be similar to each other. Therefore, in order to minimize the computational complexity, the problem can be reduced to a two dimensional space. Rather than representing the cost matrix as a cube in three dimensional space ($\mathbf{R^3}$) or hypercube in general case in n-dimensional space ($\mathbf{R^n}$) more reasonable from complexity points of view will be putting results in a series. In this method, data is presented as $N - 1$ connected bipartite graphs and $N - 1$ Munkres assignments are performed. Function which it minimizes simplifies a little and looks like this:

$$S_{2D}(R_1, \ldots, R_N) = \sum_{l=1}^{K} (S_{Jac}(B_l^1, B_{l'}^2) + S_{Jac}(B_{l'}^2, B_{l''}^3) + \cdots + S_{Jac}(B_{l^{N-2}}^{N-1}, B_{l^{N-1}}^N)) \tag{9}$$

where B_l^1 is l'th bi-cluster from result R_1 and $B_{l'}^2$ is bi-cluster corresponding to it from result R_2. Next $B_{l''}^3$ is a bi-cluster from result R_3 corresponding to bi-cluster $B_{l'}^2$. And so on. Hungarian algorithm is performed on first pair of results. Then, the third result is added, and Hungarian algorithm is performed between the second and third. The procedure is repeated until all the results will be added. Function $S_{2D}(R_1, \ldots, R_N)$ is from range:

$$0 \leqslant S_{2D}(R_1, \ldots, R_N) \leqslant K * (N - 1) \tag{10}$$

The upper values the function S_{2D} denote the number of assignments (execution of the Hungarian algorithm) that should be done to assess the quality of the overall fit. Value of $K * (N - 1)$ (bi-clusters are compared only within neighbouring results) is usually much smaller than $K * \binom{N}{2}$ (all bi-clusters in the group are compared with each other), and the quality of this approach can be a bit lower than the general approach because it search a local minimum.

After performing Hungarian algorithm on each pair of neighbouring results, K "chains" of bi-clusters are obtained. Each consisting of N bi-clusters derived from the one of N results. This final assignment is in depended mainly by placement of results—the sequence is crucial, but not always. If all the results are very much similar to each other—then the order may not be relevant, and the solution is then optimal. Consensus algorithm is as follows:

- Using a generalized Hungarian algorithm assign bi-clusters from all methods so as to form K sets, each consisting of N bi-clusters,
- Compute for each bi-cluster a ACV quality index [11],
- In each k'th set, remove bi-clusters with quality index below certain threshold T_1 (parameter set by the user or computed automatically),
- For each k'th set compute average quality index, and remove whole set if its value is below certain threshold T_2 (optional parameter set by the user or computed automatically),
- For each k'th set compute average $n_{i,k}$ (number for i'th attribute, denotes the number of bi-clusters in set k, in which attribute is present), and remove whole set if its value is below certain threshold T_3 (optional parameter set by the user or computed automatically),
- Match the weight to each attribute i from bi-cluster j taken from set k, such that:

$$W_{i,k} = \frac{n_{i,k} + \frac{Q_{i,k} - \min\limits_k Q_k}{\max\limits_k Q_k - \min\limits_k Q_k} * N}{2} \tag{11}$$

Where:

- $n_{i,k}$—number for each i'th attribute, denotes the number of bi-clusters in set k, in which attribute is present.
- $Q_{i,k}$—average value of quality index of bi-clusters in k'th set, which contains attribute i'th.
- $\min_k Q_k$—minimum value of quality index in k'th set.
- $\max_k Q_k$—maximum value of quality index in k'th set
- N—number of results/elements in sets.

- Set $P = N$,
- For every set representing single bi-cluster:
  - Select only those attributes, for witch value of $W_{i,k}$ is equal or greater than P.
  - If number of attributes in bi-cluster are equal or greater than MinC and/or quality of bi-cluster is equal or greater than MinQ, than stop, otherwise go to 3.
  - Decrease P, and go to step 1.

4 Software

Its distributed system written in C# programming language and .NET Framework. It has implemented several algorithms taken from literature and consensus methods described in this thesis. Graphical user interface is based on Windows Presentation Foundation (Fig. 2). Communication within program and within different instances of AspectAnalyzer on different nodes is based on Microsoft MSMQ queues and all mathematical computation are done using ILNumerics [5].

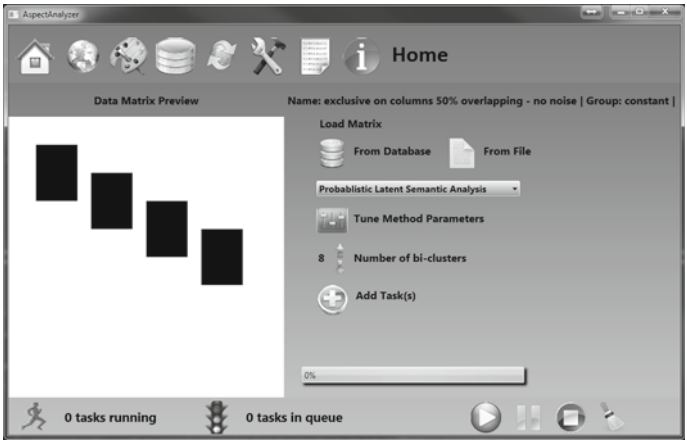


Fig. 2 AspectAnalyzer main windows

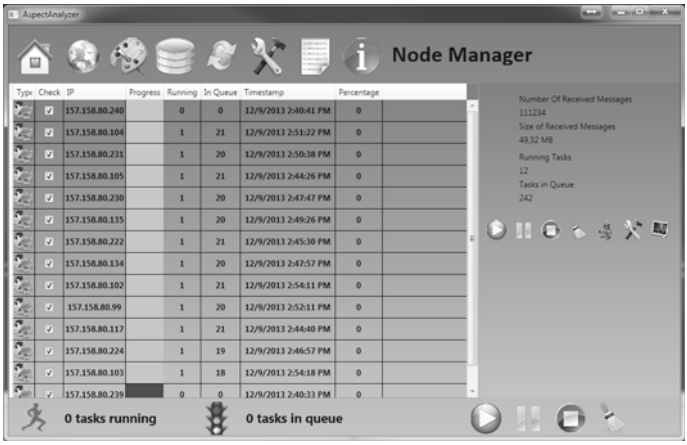


Fig. 3 AspectAnalyzer node manager panel

Thanks to the use of the database not integrated with the program, there is an opportunity to build a distributed system. It is possible to run many instances of AspectAnalyzer on a different nodes, different location etc. All instances can be set to master-slave model in which one instance is master node, and all others should be in slave mode. All nodes report to master every 5 s with information about current load, completeness of current tasks etc. Master node can manage remotely by sending specific instructions to slave-node using its IP address. Using Node Manager panel shown on Fig. 3 user can specify tasks, define experiments and system will automatically balance those jobs over running instances taking into account current load, number of cores, etc. Remote steering has the same abilities as normal one, and whole communication is done using MSMQ, so only one limitation is that ports on nodes IP should by open between every slave node and master node.

Using result view user is able to browse over results stored in database. Main window shows only general view with list of data matrices and summary number of results for it. Double click on matrix results with loading it to main screen and options with defining bi-clustering experiments. Other way is to clicking "chart and notes" icon which for the selected matrix displays in the table a more detailed summary. It contains results grouped by method and number of bi-clusters with average, minimum and maximum value of divergence function (if such function exist for selected method). The third level of nesting, available under an icon mentioning above, is a view of the individual results.

Whole system is based on free and widely available components as ready to use installer posted public on dedicated website http://AspectAnalyzer.foszner.pl/. Project site in addition to the installation version of the program itself also contains a comprehensive description and user manuals. Is organized in the form of a blog on which are published up to date information about changes and new versions.

5 Distributed Computations

AspectAnalyzer software is able to create network for distributed computations. For this purpose, on all computers that will serve as network nodes, user should install the AspectAnalyzer software. For the network to work properly there should be one master node and at least one slave nodes (the more the better). For each instance of slave, user should set the IP address of the master in its configuration. Master node will configure automatically when the slave nodes start report. Example Network with four slave nodes is shown on Fig. 4.

Slaves nodes report about their status every 5 s (how many and which jobs are in the queue). These are very short messages with information about the current load. This is handled by the MSMQ queues, and does not significantly affect the system performance. The master node allocate new tasks based on (1) the quantity of jobs in the slaves queue, and (2) the size of these tasks (data matrix size). The whole process, user can keep track via the "Node Manager" panel (shown on Fig. 3). In addition to the informational value of this panel, it also allows to exclude the node from the network and/or stop the tasks that are being performed on it.

Configuration of distributed computing is possible only with the external database (Microsoft SQL Express 2008 or grater). The results obtained by slave nodes are entered by them directly to the base (the address of which is given in the configuration of each).

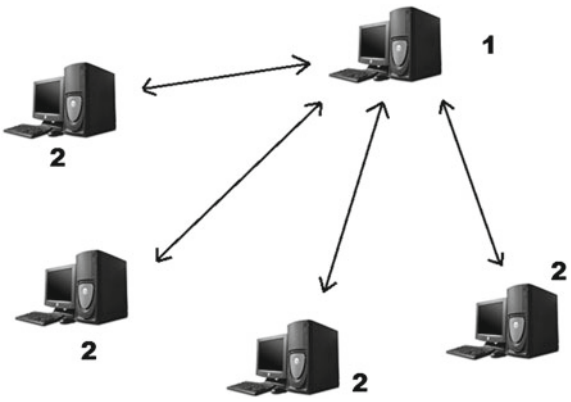


Fig. 4 (1) Master—(2) Slaves network created by AspectAnalyzer

6 Conclusions

The advantages of AspectAnalyzer software over the related work are:

- This is a ready-to-use software package witch does not require external dependencies, (such as "BiBench" where each module user must find and compile by himself, or "biclust" package where user must handle R-dependencies)
- Consensus algorithms
- Significant time optimizations by using fast libraries for mathematical calculations (ILNumerics).

The software has been released to the public on the Internet, along with extensive service organized in the form of a blog. At the address http://aspectanalyzer.foszner.pl was posted ready to use installer, along with a complete user manual. In addition, the portal allows report bugs, new features, and questions about the software. Will be published also detailed information about current and planned versions. All software is provided free of charge and will include a complete, ready-to-run package.

Acknowledgments The work was performed using the infrastructure supported by POIG.02.03.01-24-099/13 grant:"GeCONiI—Upper Silesian Center for Computational Science and Engineering". A.P. was supported by NCN Opus grant UMO-2011/01/B/ST6/06868.

References

1. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the em algorithm. J. R. Stat. Soc. **39**(1), 1–38 (1977)
2. Eren, K., Deveci, M., Kucuktunc, O., Catalyurek, U.V.: A comparative analysis of biclustering algorithms for gene expression data. Brief. Bioinform. **42**(9), 279–292 (2012)
3. Hochreiter, S., Bodenhofer, U., Heusel, M.: Fabia: factor analysis for bicluster acquisition. Bioinformatics **26**(12), 267–280 (2008)
4. Hofmann, T.: Unsupervised learning by probabilistic latent semantic analysis. Mach. Learn. J. **42**(1–2), 177–196 (2001)
5. ILNumerics GmbH: Ilnumerics—computing and visulaization engine. http://ilnumerics.net/
6. Kaiser, S., Leisch, F.: A toolbox for bicluster analysis in r. department of statistics. Technical report 28, Ludwig Maximilians Unversitat Munchen (2008)
7. Lee, D.D., Seung, H.S.: Algorithms for non-negative matrix factorization, pp. 556–562. In: NIPS 2000, Denver, USA (2000)
8. Li, G., Ma, Q., ang, A.H. Paterson, H.T., Xu, Y.: Qubic: a qualitative biclustering algorithm for analyses of gene expression data. Nucleic Acids Res. **37**(15) (2009)
9. Madeira, S.C., Oliveira, A.L.: Biclustering algorithms for biological data analysis: a survey. IEEE Trans. Comput. Biol. Bioinform. **1**(1), 24–45 (2004)
10. Pascual-Montano, A., Carazo, J.M., Kochi, K., Lehmann, D., Pascual-Marqui, R.D.: Non-smooth non-negative matrix factorization. IEEE Trans. Pattern Anal. Mach. Intell. **28**(3), 403–415 (2006)
11. Teng, L., Chan, L.: Discovering biclusters by iteratively sorting with weighted correlation coefficient in gene expression data. J. Signal Process. Syst. **50**(3), 267–280 (2008)

Algorithm for Finding Zero Factor Free Rules

Grete Lind and Rein Kuusik

Abstract Class detection rules are mostly used for classifying new objects. Another possible usage is to describe a set of objects (a class) by the rules. Determinacy Analysis (DA) is a knowledge mining method with such purpose. Sets of rules are used to answer the questions "Who are they (objects of the class)?", "How can we describe them?". Rules found by different DA methods tend to contain some redundant information called zero factors. In this paper we show how zero factors are related to closed sets and minimal generators. We propose a new algorithm that extracts zero-factor-free rules and zero factors themselves, based on finding generators. Knowing zero factors gives to the analyst important additional knowledge for understanding the essence of the described set of objects (a class).

Keywords Determinacy analysis · Rule · Zero factor · Minimal generator

1 Introduction

The problem that will be solved in this paper has arisen from the method called Determinacy Analysis (DA) which belongs to machine learning field.

There are two directions (subtasks) in machine learning. Direction 1—the main task is a *Classification task*: to find rules for classification of unknown object(s) on the basis of learning examples. Direction 2—*Data Analysis and Data Mining task*: to use the found rules for describing the class under analysis answering the questions: "Who are they (objects of the class)?", "How can we describe them?", "What distinguishes them from others?".

The best representative of direction 2 is a method called Determinacy Analysis [3, 4] that presents an original methodology for answering these questions. DA outputs the accurate and complete rule system in the form of rules "IF X THEN

G. Lind (✉) · R. Kuusik
Department of Informatics, Tallinn University of Technology, Tallinn, Estonia
e-mail: grete.lind@ttu.ee

R. Kuusik
e-mail: rein.kuusik1@ttu.ee

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_36

Class" so that each object is covered by one rule at most. All extracted rules cover only the set of objects under analysis. Now the analyst can describe the set of objects under analysis and also determine what is specific for the class and what separates different classes. If the description is not good enough he/she can change the order of attributes or add new ones.

Problem. Extracted rules consist of several factors (certain attributes with certain values) and the question is which of them are the most important and which are superfluous, giving no additional knowledge. For example, if a rule contains two factors: (an attribute) "Are you living in the countryside?" with a value "Yes", and (another attribute) "Do you have cows?" with a value "Yes", then the first attribute makes no sense, it is redundant because having cows means that one lives in the country. Consequently, it means that the first factor is an inessential factor (a zero factor) because it does not include any new knowledge for the researcher. The question is how we can avoid such (zero) factors.

DA has an approach how to ascertain zero factors, but this is hindsight. The factors are added into the rules one by one, but we cannot say at the moment of addition whether a certain factor remains essential after adding the next one(s). Therefore the analyst has to measure the influence of every factor regarding all other factors in the final rule. Different orders of factors tend to give different results consisting of different sets of rules (the approach is presented in Sect. 2.2). It is not realistic to measure the results of all different orders of attributes and it means that this approach is unusable. The task is to elaborate a simple way to avoid zero factors for extracting zero factor free rules.

Next we present a brief description of determinacy analysis in order to understand its essence and then explain the idea for extracting zero factor free rules.

2 Description of Determination Analysis

2.1 Definitions

Definitions are given according to [3, 5].

The idea behind DA is that a rule can be found based on the frequencies of joint occurrence or non-occurrence of events. Such a rule is called determinacy or determination and the mathematical theory of such rules is called determinacy analysis [4].

If it is observable that an occurrence of X is always followed by an occurrence of Y, it means that there exists a rule "If X then Y", or $X \rightarrow Y$. Such correlation between X and Y is called *determination* (from X to Y). Here X is the *determinative (determining)* and Y is the *determinable*.

The determinative (X) consists of one or more factors. A factor is an attribute with its certain value. Each attribute can have many different discrete values and gives as

many different factors as many different values it has. Factors coming from the same attribute are not contained in the same X.

Each rule has two characteristics: accuracy and completeness.

Accuracy of determination $X \rightarrow Y$ shows to what extent X determines Y. It is defined as the proportion of occurrences of Y among the occurrences of X:

$$A(X \rightarrow Y) = n(XY)/n(X) \tag{1}$$

where
$A(X \rightarrow Y)$ is the accuracy of determination,
n(X) is the number of objects having feature X and
n(X Y) is the number of objects having both features X and Y.

Completeness of determination $X \rightarrow Y$ shows which part of cases having feature Y can be explained by determination $X \rightarrow Y$. It is the percentage of occurrences of X among the occurrences of Y:

$$C(X \rightarrow Y) = n(XY)/n(Y) \tag{2}$$

where
$C(X \rightarrow Y)$ is the completeness of determination,
n(Y) is the number of objects having feature Y and
n(X Y) is the number of objects having both features X and Y.

Both accuracy and completeness can have values ranging from 0 to 1. A value of 1 shows maximum accuracy or completeness, 0 means that the rule is not accurate or complete at all. A value between 0 and 1 shows quasideterminism.

If all objects having feature X also have feature Y then the determination is (maximally) accurate. In case of accurate determination $A(X \rightarrow Y) = 1$ (100 %).

The majority of rules are not accurate. In case of inaccurate rule $A(X \rightarrow Y) < 1$.

In order to make a determination more (or less) accurate, complementary factors are added to the left part of the rule. Adding factor Z into the rule $X \rightarrow Y$, we get the rule $XZ \rightarrow Y$, adding factor W to the rule $XZ \rightarrow Y$, we get the rule $XZW \rightarrow Y$ etc.

The contribution of factor Z to the accuracy of the rule $XZ \rightarrow Y$ is measured by the increase of accuracy $\Delta A(Z)$ caused by addition of factor Z into the rule $X \rightarrow Y$:

$$\Delta A(Z) = A(XZ \rightarrow Y) - A(X \rightarrow Y) \ . \tag{3}$$

The contribution to accuracy can range from -1 to 1.

If $\Delta A(Z) > 0$ then Z is a *positive factor*. Adding a positive factor makes the rule more accurate, sometimes the resultant rule is (maximally) accurate. If $\Delta A(Z) < 0$ then Z is a *negative factor*. Adding a negative factor decreases the rule's accuracy, some-times down to zero. If $\Delta A(Z) = 0$ then Z is a *zero (or inessential) factor*. Adding a zero factor does not change the rule's accuracy. An *accurate rule* contains no negative factors, all factors are positive or zero factors. A rule consisting of positive factors only, is called a *normal rule*.

If $C(X \rightarrow Y) = 1$ (100 %) then the rule $X \rightarrow Y$ is (maximally) complete. It means that Y is always explained by X. In case of an incomplete rule $C(X \rightarrow Y) < 1$, X does not explain all occurrences of Y.

The *contribution* of factor Z *to the completeness* of the rule $XZ \rightarrow Y$ is measured by the increase of completeness $\Delta C(Z)$ by addition of factor Z into the rule $X \rightarrow Y$:

$$\Delta C(Z) = C(XZ \rightarrow Y) - C(X \rightarrow Y) \ . \tag{4}$$

The contribution of whatever factor to completeness is negative or zero.

A set of rules is called a *system of rules* and characterized by average accuracy, summarized completeness and summarized capacity (the number of objects/cases covered by rules). A *system* is called *complete* if its completeness is 1. A *system* is called *accurate* if its accuracy is 1. A system is accurate when all of its rules are accurate.

2.2 Example of DA

DA enables to find different sets of rules, depending on the order of inclusion of the attributes into the analysis. Attributes (factors) are added into (the left sides of) the rules (X) one by one in a given order. If a rule is accurate it will not be expanded by adding the next factor. At the same time non-accurate rules will acquire next factors until they become accurate (or there is no more attributes to add). This way a set of non-overlapping rules of different length (called rank) is obtained. If there are no contradictions in the data, then the result covers all objects of the observable class.

Next we give an example of two different sets of rules obtained with different orders of attributes. We use the well-known Quinlan's data set (of eight people characterized by height, hair color and eye color [10]—see Table 1) and describe (the people belonging to) the class "-" by accurate rules.

Table 1 Quinlan's table

Height	Hair	Eyes	Class
tall	dark	blue	+
short	dark	blue	+
tall	blond	blue	−
tall	red	blue	−
tall	blond	brown	+
short	blond	blue	−
short	blond	brown	+
tall	dark	brown	+

If the order is: (1) Hair, (2) Eyes, (3) Height; then we get:

- A1: Hair.red → Class.– (C = 1/3);
- A2: Hair.blond & Eyes.blue → Class.– (C = 2/3).

For the order (1) Height, (2) Hair, (3) Eyes; the set of rules is as follows:

- B1: Height.tall&Hair.red → Class.– (C = 1/3);
- B2: Height.short&Hair.blond&Eyes.blue → Class.– (C = 1/3);
- B3: Height.tall&Hair.blond&Eyes.blue → Class.– (C = 1/3).

An algorithm for the presented "step by step" approach is given in [7].

2.3 The Task of DA

As the author of DA Chesnokov states, the main task is to find maximally accurate and complete systems of rules [5]. If we want to get rid of redundant information (like "living in the countryside" in case of "having cows") then the left side of the rules has to contain only such factors that make a rule more accurate than it was without them, i.e. positive factors. As shown in [8] it is not easy to avoid them: "The fact that some factor has a positive impact on the accuracy at the moment it is added into the rule does not guarantee that the factor retains its positiveness" after the addition of the next one(s).

It means that in case of another order a certain factor may be left out of nearly the same rule because actually it is a zero (inessential) factor in the itemset (conjunction of factors) that defines a class.

If we analyze the presented accurate rule systems extracted by DA (see Sect. 2.2), we can see that the first system is free of zero factors: A1 consists of one factor only; in A2 both factors are necessary for detecting a class, neither factor alone gives an accurate rule. In the second rule system all 3 rules contain zero factors: Height.tall in B1; Height.short in B2; and Height.tall in B3. As we can see the attribute Height is not needed for determining class "-", but we do not know it in advance.

DA cannot automatically identify which factors in the rule are zero factors and there is no possibility to generate all rule systems based on the entered set of attributes—these are the main weaknesses of the method. We try to solve both problems.

Identification of zero factors would give a possibility to foreshorten rules. Identified zero factors that have to be left out from the left side of the rule can be moved to the right side—the conclusion part.

Instead of generating all possible different systems of rules our solution is to find all zero-factor-free rules that is a suitable basis for forming different (accurate and complete) systems of rules.

Next we will show that it is possible to recognize and avoid zero factors (from the left sides of the rules) and explain how to do it.

3 Theoretical Foundations

In this section we introduce different types of zero factors, the concepts of generator and closed set and show their relations to DA rules.

3.1 Different Types of Zero Factors

We have found that there are two types of zero factors:

1. the ones with zero contribution to the completeness ($\Delta C = 0$) that do not change the rule's coverage (set of covered objects) and frequency (the number of objects it covers) and
2. the ones with negative contribution to the completeness ($\Delta C < 0$) that decrease the rule's frequency.

We will call them *zero-zero factors* and *zero-negative factors*, accordingly. Recall that zero factor means a factor with zero contribution to the accuracy ($\Delta A = 0$), so the accuracy of the rule does not change in either case.

"Living in the countryside" in case of "having cows" is an example of a zero-zero factor (if everyone who has cows lives in the country).

Also Height.tall in the rule B1 (with completeness 1/3) is a zero-zero factor, because the rule without it (A1) has the same completeness (both rules cover exactly the same objects). Height.short in B2 and Height.tall in B3 are zero-negative factors. If either of them is added into the rule A2 then the completeness of the rule decreases from 2/3 to 1/3. This negative difference (1/3 − 2/3) is the factor's contribution to the rule's completeness.

3.2 Closed Sets and Generators

In frequent itemset mining an item is a binary attribute that can be either present or not in a transaction (a database record). For example, in market basket databases the items represent purchased goods. Extending the concept to multi-valued attributes, an item is a certain attribute with a certain value from the set of different possible values for that attribute. For example, in case of a market basket database, instead of "bread" there can be either "black bread" or "white bread" (i.e. attribute "bread" with either value). Such an item corresponds to a DA factor.

A *closed (item)set* is the maximal set of items common to a set of objects [9], it has no superset with the same support (i.e. frequency) [13]. Adding whichever item decreases its coverage and frequency. For example, one of the closed sets in Table 1 is Hair.blond&Eyes.blue&Class.- with frequency 2. If we add Height.short or Height.tall to this itemset, then the frequency changes and the resultant itemset is not the same closed set anymore.

A *closure* is the smallest (minimal) closed itemset containing the given itemset [2] i.e. the itemset's maximal superset with the same frequency. For example, the closure of Hair.red (frequency = 1) is Height.tall&Hair.red&Eyes.blue&Class.- (frequency = 1). A closed set is the same as its closure.

A *(minimal) generator* of a closed set is an itemset with the same closure and with no proper subsets with the same closure [1]. Taking away whichever item increases its coverage (and frequency). For example, Hair.blond&Eyes.blue with frequency 2 is a generator of the closed set Hair.blond&Eyes.blue&Class.- with a frequency of 2. Taking away either Hair.blond or Eyes.blue from the generator gives us an itemset with bigger frequency and thus with different closure.

If the number of items in a closed set and its generator differs more than by one then also the sets between the minimal generator and the closed set can be used for generating a closed set and can be called generators (for example, itemsets between Hair.red and Height.tall&Hair.red&Eyes.blue&Class.-). However, mostly "generator" means the minimal generator.

A closed set can have more than one minimal generator. For example, the closed set Hair.blond&Eyes.blue&Class.- has two (minimal) generators: Hair.blond& Eyes.blue and Hair.blond&Class.-.

A closed set or a generator is said to be frequent if its frequency is more than or equal to a given threshold. If the frequency threshold is 2, then Height.tall& Hair.red&Eyes.blue&Class.- and its generators are infrequent (frequency = 1); Hair.blond&Eyes.blue&Class.- and its generators are frequent (frequency = 2).

3.3 Relations

A closed set is the maximal set of items common to a set of objects and its (minimal) generator is a minimal set of items common to that set of objects. Between the closed set and its generator there are such items, the addition or removal of which does not change the coverage and frequency of the itemset. Those items are similar to zero-zero factors that do not change either the accuracy or the completeness of the DA rule. Just the class-belonging usually is not observed in case of closed sets. Consequently, in order to avoid zero-zero factors the left side of the rule has to be a minimal generator.

Minimal generators do not contain zero-zero factors, but they can contain zero-negative factors. For example, the generator Height.tall&Hair.blond&Eyes. blue determines Class.- (i.e. rule B3: Height.tall&Hair.blond&Eyes.blue → Class.-), but Height.tall is a zero-negative factor, because Hair.blond&Eyes.blue is enough to determine Class.- (rule A2: Hair.blond&Eyes.blue → Class.-). Height. tall decreases the rule's completeness by 1/3 (from 2/3 to 1/3). Thus, if a generator produces a rule (generator → class) then the rules with super-generators of that generator contain zero-negative factors and are redundant.

Therefore, for class detection we need such (minimal) generators that define a class and have no such subset that defines a class.

4 Zero Factor Free DA

From minimal generators that define a class, we can build rules IF minimal-generator THEN class (min-gen → class). In this case we get rules with zero-factor-free (ZFF) left sides and therefore we call our approach Zero Factor Free DA (ZFF DA).

Next we present an algorithm for producing zero-factor-free rules—a set of accurate normal rules. Additionally, it can output rules where zero-zero factors are on the right side (min-gen → zero-factors)—association rules. The algorithm is based on finding generators. For each generator it is possible to make sure whether it defines a class and also to detect the difference with its corresponding closed set (i.e. zero-zero factors). Finding of all needed generators is guaranteed. The majority of their unwanted supergenerators (containing zero factors) can be avoided, the remaining part is excluded by compression of the initial result (after the main algorithm).

4.1 Description of the Algorithm

This is a depth-first search algorithm that makes subsequent extracts of objects containing certain factors. From the root to the leaves (of search tree), the frequencies of extracts always decrease. Each extract is determined by a generator. Each generator is found only once.

The algorithm uses frequency tables that show for each attribute the frequencies of all its possible values (in the set of objects for which it is found).

Frequencies (in the frequency table) can be equal to or smaller than the current ("leading") frequency (the number of the objects in the current extract). Equal frequency shows that all objects of the extract contain that factor. For each attribute, there can be at most one frequency equal to the leading one, in such case all other frequencies for that attribute are zeroes. Factors with such frequency are zero-zero factors (in the current extract).

Detecting whether the generator determines a class is analogous. If for the class attribute one value has a frequency equal to the leading one (and others are zeroes), then all objects of the extract belong to that class.

In order to prevent finding supergenerators (subrules) of the current generator (rule) the algorithm backtracks after detecting a class. Only such supergenerators can be avoided that are not found yet.

If no class was detected (objects of the extract belong to different classes) then the next factor to be included into the generator (left side of the rule) is selected by the frequency (from the frequency table). Its frequency has to be smaller than the frequency of the current extract and bigger than or equal to the given frequency threshold. The first condition prevents the inclusion of zero-zero factors (of the current extract), the second one is usual in mining frequent sets and rules. In order to find minimal generators only (not the ones between a minimal generator and closed set), the minimal one of suitable frequencies is chosen. However this condition does not guarantee that the next generator is always minimal, but usually it is. If there is more than one factor with such frequency, just one of them is selected. The chosen factor together with the previously selected factors of the same branch forms a generator and determines a narrower (than the current) set of objects.

In order to avoid repeatedly finding already found generators, the frequency of the selected factor (the "leading" factor) is set to zero in the current frequency table. Before selecting the next leading factor, those zeroes are "brought down" from the frequency table of the previous level to the current level (except for the initial level).

The following notation is used in pseudocode of the algorithm:
`attr`—number of attributes (excluding class);
X_0—initial data table (objects*(`attr`+class));
`t`—number of the step (or level) of the recursion;
X_t—set of objects (extract) at level `t`;
FT_t—frequency table for a set X_t;
gen_t—generator at level `t`;
`zf`—zero factors (regarding gen_t);
`noclass`—the truth-value of whether the class is detected for gen_t;
`gclass`—class value of gen_t;
`V`—"leading" frequency i.e. frequency of extract;
`minfr`—frequency threshold (minimal allowed number of covered objects);
Factors are given as $value_{attribute}$;
Assignments are indicated by "←" ("=" is for comparison).
The pseudocode of the algorithm is given below (Algorithm 1).

Algorithm 1

```
Given: X0 , minfr>0
A1. t←0 ; gen0 ←{}
A2. find FT0
A3. FOR EACH factor hf=1,⋯,attr ∈FT0 with frequency V=min FT0[hf]≥minfr
DO
A4.  FT0[hf]←0
A5.  make_extract(t+1; hf; V)
    NEXT
End of Algorithm
PROCEDURE make_extract(t; hf; V)
B1. gent ←gent−1∪ hf
B2. zf←{} ; gclass←0 ; noclass←true
B3. separate submatrix Xt ⊂Xt−1 such that Xt={Xij∈Xt−1| X.f=hf}
B4. find FTt
B5. FOR EACH empty position p (p∈1,⋯,attr) in gent DO
B6.  IF exists value h such that FTt[hp]= V THEN
B7.   zf←zf∪hp
     ENDIF
    NEXT
B8. IF exists value clv such that FTt[clvcl]= V THEN
B9.  gclass←clv ; noclass←false
    ENDIF
B10. output gent, zf, gclass, V
B11. IF noclass AND V>minfr THEN
B12.  ZeroesDown(t)
B13.  FOR EACH hu=1,⋯,attr  ∈FTt with frequency V2=min FTt[hu]≥minfr and
V2<V DO
B14.   FTt[hu]←0
B15.   make_extract(t+1; hu; V2)
      NEXT
     ENDIF
END PROCEDURE
PROCEDURE ZeroesDown(t)
C1. FOR EACH factor hu=1,⋯,attr ∈FTt with frequency > 0 DO
C2.  IF FTt−1[hu]=0 THEN FTt[hu]←0
    NEXT
END PROCEDURE
```

The initial data table X_0 and the frequency threshold minfr are given. The main program starts with initial assignments for a level of recursion t and the empty generator gen_0 (step A1). Next the frequency table FT_0 for X_0 is found (A2). In step A3 each factor with a suitable frequency ($\geq$minfr) is chosen as a leading factor (for inclusion into generator) in ascending order (by frequencies). The frequency of the leading factor h_f is set to zero in the frequency table FT_0 (A4) and an extract by h_f is made (A5).

While the main program makes extracts from initial data, the recursive procedure make_extract handles all deeper levels. It starts with evaluating the current generator gen_t (B1) and giving initial values for the set of zero factors zf, class value gclass of current generator and truth-value noclass for indicating whether the class is

found (B2). Next the subset of objects X_t is extracted by the leading factor h_f (B3) and the corresponding frequency table FT_t is found (B4). Step B5 goes through all empty positions (attributes without value) in current generator gen_t (as a vector) and B6 searches for the value (of that attribute) with frequency equal to the leading one V. If one exists, it is a zero-zero factor (regarding gen_t) and it is included into the set of zero factors zf (B7). In B8 a similar check is made for class attribute. If equal frequency is found, then the generator gen_t determines a class and in B9 its class value gclass and indicator noclass are evaluated accordingly. In step B10 the generator is outputted together with its frequency, possible zero factors and class. Several conditions may be applied to decide whether to output the current generator or not—this is a possibility to leave out generators without a class and/or without zero factors (or without new zero factors at that level). If zf is not empty, the rule IF gen_t THEN zf can be produced. If class was detected then the rule IF gen_t THEN gclass can be produced.

Step B11 checks the suitability of making a subsequent extract. If a class is not found (noclass=true), then there is hope to detect it by a longer generator(s). If the frequency V is above the threshold minfr, then there is a possibility to find frequency that is $<$V and $\geq$minfr. If that check gives a positive result, then the zeroes from the frequency table of the previous level are "brought down" (B12). The procedure ZeroesDown goes through the current frequency table and for each factor with a frequency over zero (C1) its frequency at the previous level is checked (C2). If the latter is zero, then the factor gets a zero frequency at the current level as well (C2).

Step B13 goes through all factors that are suitable for subsequent extract i.e. with frequency smaller than the leading one (in order to prevent including zero-zero factors) and greater than or equal to the given frequency threshold minfr. Again the order is ascending. The frequency of selected next factor h_u is set to zero (B14) and recursive call to procedure make_extract is made with new leading factor h_u and its frequency V2 (B15).

4.2 Example

In the following example we use data from [11], given in Table 2. The data set consists of 14 objects described by four attributes and class.

With the frequency threshold 2 the algorithm finds 39 generators. The result contains 11 generators with class (suitable for producing rules IF generator THEN class) and 6 generators with zero factor(s) (suitable for producing rules IF generator THEN zero-factors); 2 generators have both. 24 generators have neither. Of course, it is possible not to output them. Also it is possible filter out (or not to output) generators without class or generators without zero factors.

After the compression 6 generators are left for Class "P", listed in Table 3.

For each listed (minimal) generator (in Table 3) we get a zero-factor-free class detection rule (IF minimal generator THEN class). For example, from G5 we can

Table 2 Initial data table

Obj	Ou(tlook)	Te(mperature)	Hu(midity)	Wi(ndy)	Cl(ass)
1	sunny	hot	high	false	N
2	sunny	hot	high	true	N
3	overcast	hot	high	false	P
4	rain	mild	high	false	P
5	rain	cool	normal	false	P
6	rain	cool	normal	true	N
7	overcast	cool	normal	true	P
8	sunny	mild	high	false	N
9	sunny	cool	normal	false	P
10	rain	mild	normal	false	P
11	sunny	mild	normal	true	P
12	overcast	mild	high	true	P
13	overcast	hot	normal	false	P
14	rain	mild	high	true	N

Table 3 Minimal generators of class "P" and corresponding generator-based rules

	Minimal generator	Rule IF minimal generator THEN class
		Rule IF minimal generator THEN zero-factor(s)
G1	Ou.overcast	IF Outlook.overcast THEN Class.P
G5	Te.cool&Wi.false	IF Temperature.cool&Windy.false THEN Class.P
		IF Temperature.cool&Windy.false THEN Hu.normal
G13	Ou.sunny&Hu.normal	IF Outlook.sunny&Humidity.normal THEN Class.P
G24	Ou.rain&Wi.false	IF Outlook.rain&Windy.false THEN Class.P
G26	Te.mild&Hu.normal	IF Temperature.mild&Humidity.normal THEN Class.P
G38	Hu.normal&Wi.false	IF Humidity.normal&Windy.false THEN Class.P

conclude: IF Temperature.cool&Windy.false THEN Class.P. Such a conclusion holds in the given data set (Table 2).

Generator G5 has a zero factor also, this gives an association rule (IF minimal generator THEN zero-zero factors): IF Temperature.cool&Windy.false THEN Humidity.normal. It means that Temperature.cool&Windy.false is always accompanied by Humidity.normal. From the viewpoint of the data analysis task it shows that Class is "P" in case of objects which contain factors Temperature.cool&Windy.false& Humidity.normal, their essence is determined by the factors Temperature.cool& Windy.false, but not by Humidity.normal. It is important additional information for the data analyst.

A generator together with its zero-zero factors forms a closed set—the set of all the common factors of the covered objects. No other object in the given

data set contains all those factors. The two objects covered by the minimal generator Temperature.cool&Windy.false (G5) have 3 factors in common: Temperature.cool&Windy.false&Humidity.normal.

Thus from the minimal generator Temperature.cool&Windy.false (G5) we can conclude both Class.P and the presence of Humidity.normal. For the analysis task it is important to know which feature (Humidity.normal) is elicited by the other one(s).

4.3 Experiments

The method described in this paper is implemented by Jõgiste [6]. Experiments for showing dependency of execution time on number of objects (rows) or attributes (columns) were carried out using Nursery data set from UCI Machine Learning Repository [12]. The dataset contains 12000 rows and 12 columns, 2 of which were added for testing purposes. Attribute value ranges from 1 to 5. Besides execution time the number of extracts made during the work ("steps") was measured.

Dependency on number of rows was tested with 2000, 4000, 6000, 8000, 10000 and 12000 rows. The number of columns was 12. The result is given in Fig. 1.

Dependency on number of columns (Fig. 2) was tested with 2–12 columns with step 2.

As we can see, the number of columns has a strong impact on the execution time and number of extracts, while the number of rows influences them much less.

Often processing of found rules takes more time than finding the rules. The time for rule processing includes adding rules to specific rule classes, sorting or grouping if necessary and writing them to text file. Rule processing time, number of generators and file size was measured in case of different frequency threshold values, ranging from 1 to 100. Those dependencies are presented in Fig. 3.

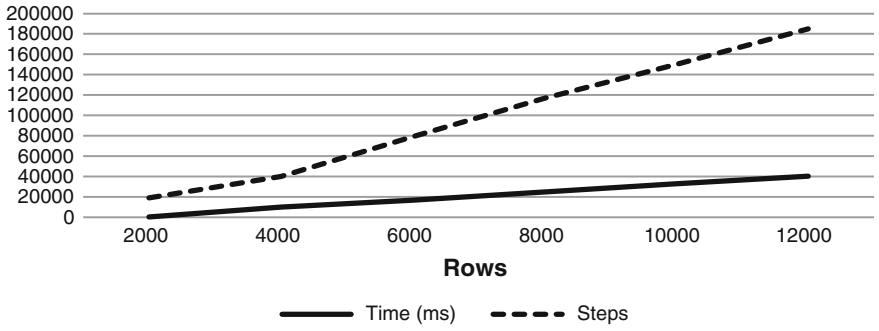


Fig. 1 Execution time dependency on number of rows [6]

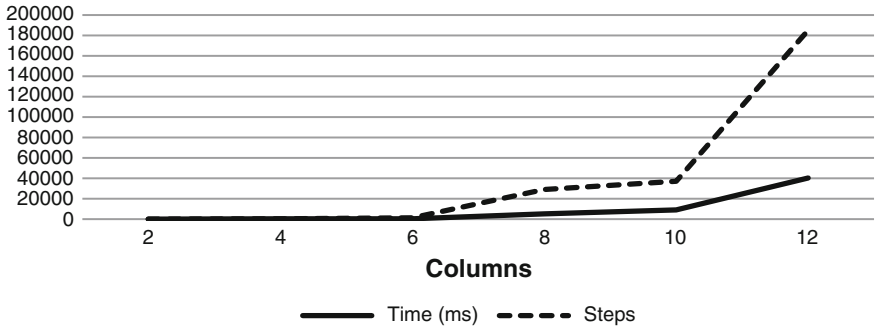


Fig. 2 Execution time dependency on number of columns [6]

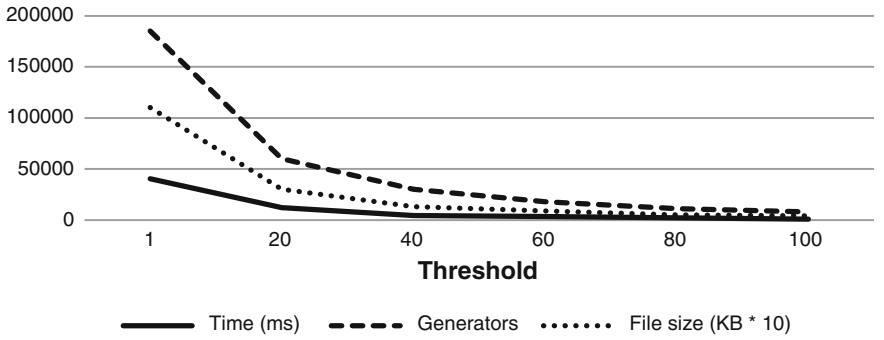


Fig. 3 Rule processing time dependency on frequency threshold [6]

5 Conclusions

In this paper an overview of a rule mining method called determinacy analysis (DA) is given and its main weakness—DA cannot mine rules free from inessential factors—is pointed out. For us the research question arose: how we can recognize such (zero) factors and how DA can mine zero factor free rules. We specified the types of zero factors, showed their connections to the generators and closed sets and found out that desirable DA rules should have a minimal generator in their left side. Based on that, we proposed an algorithm for finding rules where the left side is a minimal generator. The algorithm produces two types of rules: (1) IF generator THEN class; (2) IF generator THEN zero factor(s). The first type is a class detection rule; the second—an association rule. We also show how to interpret these rules for data analysis purposes.

Next we must improve a methodology for exploitation of the method and develop a user friendly solution for querying from the rule base and improve software.

References

1. Bastide, Y., Pasquier, N., Taouil, R., Stumme, G., Lakhal, L.: Mining minimal non-redundant association rules using frequent closed itemsets. In: Lloyd, J., Dahl, V., Furbach, U., Kerber, M., Lau, K.K., Palamidessi, C., Pereira, L., Sagiv, Y., Stuckey, P. (eds.) Computational Logic—CL 2000, LNCS, vol. 1861, pp. 972–986. Springer, Berlin (2000)
2. Bastide, Y., Taouil, R., Pasquier, N., Stumme, G., Lakhal, L.: Mining frequent patterns with counting inference. SIGKDD Explor. Newsl. **2**(2), 66–75 (2000)
3. Chesnokov, S.: Determination-analysis of social-economic data in dialogical regime (Preprint). All-Union Institute for Systems Research, Moscow (1980)
4. Chesnokov, S.: Determinacy analysis of social-economic data. Nauka, Moscow, Russia (1982)
5. Chesnokov, S.: Determinacy analysis of socio-economic data. Illustrative materials to lectures. Lecture 2: Rules. Lecture 3: Systems of rules (2002), lomonosov Moscow State University, Faculty of Economics, Moscow, unpublished (in Russian)
6. Jõgiste, L.: Prototyping of Zero-factor based DA. Master's thesis, Tallinn University of Technology (2014)
7. Lind, G., Kuusik, R.: New developments for determinacy analysis: diclique-based approach. WSEAS Trans. Inf. Sci. Appl. **5**(10), 1458–1469 (2008)
8. Lind, G., Kuusik, R.: Some problems in determinacy analysis approaches development. In: DMIN 2008, pp. 102–108. Las Vegas, Nevada (2008)
9. Pasquier, N., Bastide, Y., Taouil, R., Lakhal, L.: Pruning closed itemset lattices for association rules. In: BDA 1998, pp. 177–196. Hammamet, Tunisie (1998)
10. Quinlan, J.R.: Learning efficient classification procedures and their application to chess end games. In: Michalski, R., Carbonell, J., Mitchell, T. (eds.) Machine learning. An Artificial Intelligence Approach, pp. 463–482. Springer, Berlin (1984)
11. Quinlan, J.R.: Induction of decision trees. Mach. Learn. **1**(1), 81–106 (1986)
12. UCI: Machine Learning Repository. http://archive.ics.uci.edu/ml/datasets/Nursery
13. Zaki, M.J., Hsiao, C.J.: CHARM: an efficient algorithm for closed itemset mining. SIAM **2002**, 457–473 (2002)

Diagnostic Model for Longwall Conveyor Engines

Marcin Michalak, Beata Sikora and Jurand Sobczyk

Abstract The paper presents a new approach of wall conveyor engines diagnosis. A wall conveyor is an essential device in coal mines. Its work is usually represented by three time series of current values of three conveyor engines. The startup of the conveyor is the phase with the maximal observed load during its work cycle. In the research, each startup is described with almost twenty variables. On the basis of 1000 real monitored startups, a set of association rules was inducted. On the basis of the further rules analysis and interpretation, a set of almost 50 rules was selected to the diagnosis system. The proposed diagnosis system compares the quality (precision) of each association rule from a selected subset—the precision evaluated on the representative data—with the precision of the same rule, evaluated on newly detected startups.

Keywords Machine diagnosis · Association analysis · Association rules

1 Introduction

Though the production of energy from renewable resources has been increasing recently, the mining is still an important part of the industry. One of the aspects of making coal mining more effective (and less polluting) is an improvement of coal mining machines efficiency and reliability. This goal is achieved with the application of data mining and machine learning methods for the purpose of machine diagnosis.

M. Michalak (✉)
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: marcin.michalak@polsl.pl

B. Sikora
Institute of Mathematics, Silesian University of Technology, Gliwice, Poland
e-mail: beata.sikora@polsl.pl

J. Sobczyk
SOMAR S.A., ul. Karoliny 4, 40-186 Katowice, Poland
e-mail: j.sobczyk@somar.com.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_37

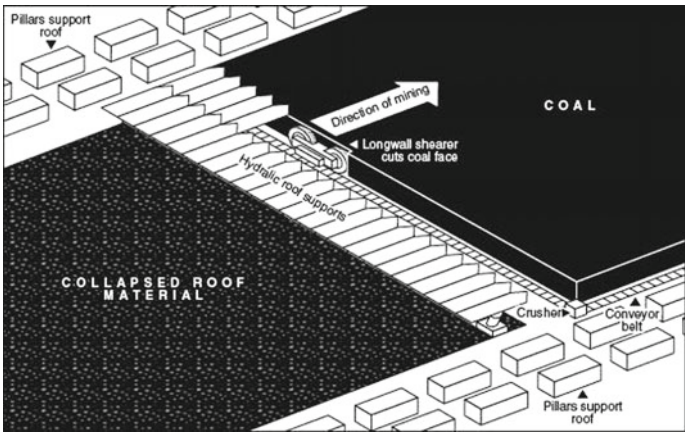


Fig. 1 A diagram of underground longwall mining (http://www.patriotcoal.com/)

The most common way of underground coal mining is longwall mining (Fig. 1). The face—a place where coal is mined—is usually up to 300 m long. The rock roof is supported by a set of powered roof support sections, which are moved with the face progress. A longwall shearer is a device that moves back and forth across the face and tears off the rock. The teared off rock is then moved by a longwall chain conveyor outside the face to the gangway.

Machine diagnostic becomes very popular in various fields of application in industry, starting from synchronous motor imminent failure conditions [5], diagnostics of direct current machine [4], rotating machines [7], to gas turbine generator systems [12].

In underground coal mining monitoring and diagnosis systems become an important part of the equipment. The applications of monitoring and diagnosing methods are widely presented in [1, 3, 6, 9].

The correct work of wall conveyors is very important for the proper operation of the longwall shearer. Its proper operation can be measured by the analysis of its startups. Each conveyor is driven by three engines: one pulling and tensing its chain and the other two just driving the conveyor [3, 8] (Fig. 2).

It is a typical situation that during a conveyor startup there is a large amount of a coal located on it. Due to this situation, the startup becomes a critical moment which implies the conveyor load and its operating point characteristics. The conveyor, whose startups were observed, was equipped with a two-speed starting system: first gear—slow start of the conveyor; second gear—basic work of the conveyor. There are also fluid coupling propelled (one gear) conveyors, offered, inter alia, by Voith, Transfluid, and Siemens.

The construction of the diagnostic models of machines/devices can be carried out as a planned experiment or can be based on the analysis of historical data. In the latter case we can have measurements which describe all states of the machine (including

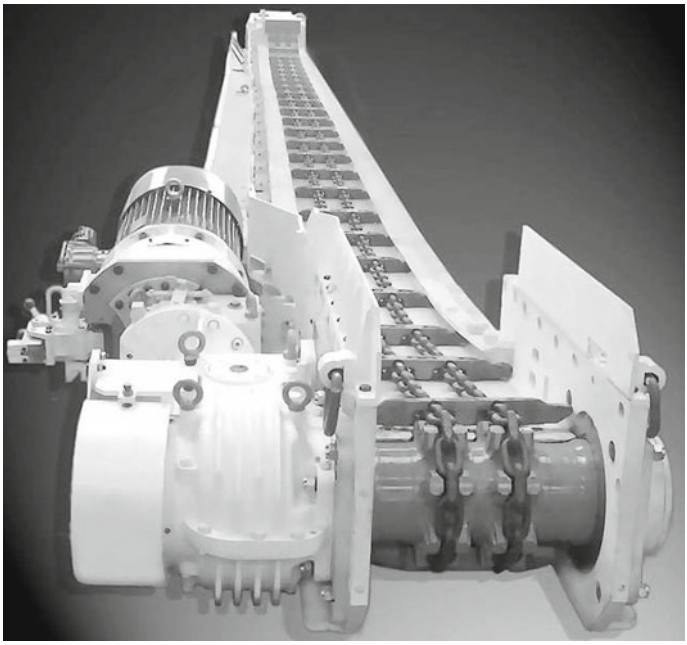


Fig. 2 A chain longwall conveyor (www.ostroj.cz)

emergency states). Alternatively, we can have a certain subset of states (e.g. the state of proper operation of the machine). When we have only measurements which illustrate the state of proper operation of the machine, we can determine the model of this state and then observe whether the successive measurements are contained within this model. Going outside the model or observing a certain trend in changes can be a motive to raise the alarm. This type of diagnostics is applied for diagnosing the work of machines and devices which work in a normal production cycle of a plant where there is no time and no permission to run the planned experiments.

This paper presents the results of research on data from one gear engines. The analysis of correct work of the conveyor was based on the observation of three engines current consumption. To be more precise: exceedances of assumed maximal levels and the total maximal current consumption. All gathered data were acquired during the "proper" longwall system work. It means that the correct starting process of the conveyor should consist of both uniformly loaded main transporting engines. Uniform loading of the engines should manifest similar courses of time series reflecting the current drawn by the engines during their work.

In the paper, the engines work was divided into two phases: startup and basic work. The startup phase is described by the maximum current value reached by the engines and the current rise time. The basic work phase is described by a time series illustrating the current drawn by the engines. The purpose of this paper is to analyse the associations between parameters reflecting the process of the conveyor startup.

The analysis used the MagnumOpus program for the induction of association rules [10, 11]. The analysis results can state a basis for the preparation of a diagnostic procedure verifying the correctness of the conveyor startups In addition, the paper presents the process of preparation, analysis and use of the achieved results.

The paper is organised as follows: it starts from the presentation of the real dataset used in the research and the definition and interpretation of variables (indices) defined for the startup description. The next part presents some association rules describing the correct diagnostic state of the conveyor work. The rules are grouped by their premises to make their interpretation easier. The goal of the paper is the presented in the following part the diagnostic procedure which may be helpful in the monitoring of the conveyor work, presented in the next section. The paper ends with some final words and a short description of the future works on the issue.

2 Background

The analysis of the conveyor work was performed on the data coming from the DEMKop monitoring system [2]. Currents were sampled every second. As the startup the period of time, between the first minimum in the current and the next maximum was defined.

From each working time the following indices were calculated:

- t—working time duration $[s]$,
- m_1, m_2, m_3—minimal value of the current during the startup $[A]$,
- M_1, M_2, M_3—maximal value of the current during the startup $[A]$,
- $\Delta t_1, \Delta t_2, \Delta t_3$—the time of increase of the current for each engine during the startup $[s]$,
- v_1, v_2, v_3—the speed of current increase for each engine $[A/s]$,
- Δ—maximal difference between currents of engines in moments t_1, t_2 and t_3 : $\max\{I_2(t_2) - I_1(t_1), I_2(t_2) - I_3(t_3)\}$ (the global difference) $[A]$,
- δ—maximal difference between currents of engines during the startup (the local difference) $[A]$,
- Δ_{12}—maximal difference between currents of engines E_1 and E_2 in moments t_1 and t_2 : $I_2(t_2) - I_1(t_1)$ (the global difference) $[A]$,
- δ_{12}—maximal difference between currents of engines E_1 and E_2 during the startup (the local difference) $[A]$,
- *est. lvl*—"established level" at $t = 50\,[s]$ (an average of all three engines).

Two groups of indices define a difference, but to distinguish their meanings, they are named as local (δ) and global (Δ). The global one is the difference between currents marking the end of the startup. For two engines it is just their difference. The local one is defined analogically way but on samples coming from the same time for each engine. The visualisation of both differences is shown on Figs. 4 and 5. Additionally, the interpretation of the current increase is shown on Fig. 3.

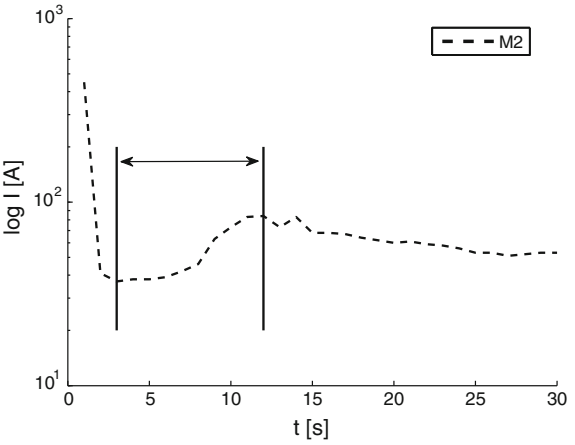


Fig. 3 The current increase time for the engine M_2

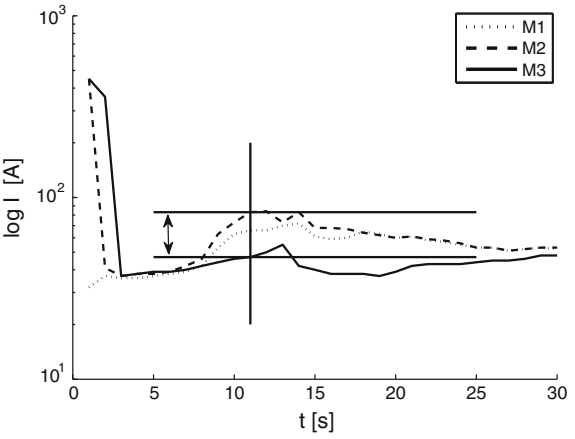


Fig. 4 The maximal global current difference between all engines

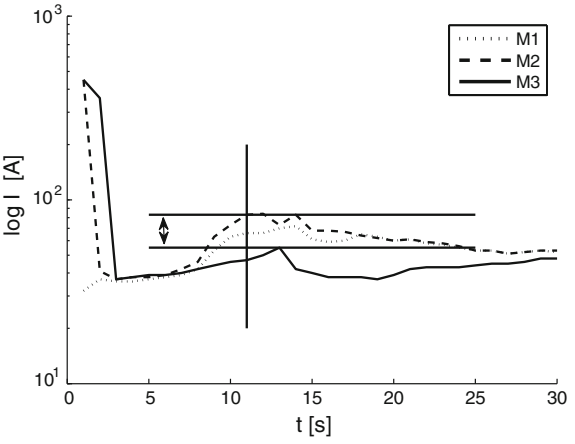


Fig. 5 The maximal global current difference between all engines

3 Association Analysis

For the association analysis 1000 startups were considered as the train-test samples. In this section a brief description of most significant association rules is presented. Each set of rules is connected with a conveyor operation aspect and is described in a separate subsection of the paper. The rules are inducted on the data, considered as coming from the proper device (the whole longwall system and the conveyor) operation.

3.1 Engine E_3—Maximal Current and Current Increase Speed

The first set of association rules (Table 1) joins the speed of the E_3 engine current increase with its maximal current consumption. It reflects a strong positive correlation between these two variables: the higher value of the current increase speed the higher maximal value of the power consumption.

The rules (here and in the further analysis) are evaluated with the fraction of a number of records that support the rule (fulfill the premises and the conclusion of a rule—column marked as *supp*) and a number of records just fulfilling the premises part of the rule (column marked as *match*). This measure is called *precision*. Another popular method of rules evaluation takes into consideration the range of the rule, defined as the fraction of supporting objects and the number of all objects in the data. Because there are 1000 of them, the column *supp* can be interpreted as the rule coverage, normalised to 1000.

As premises of all rules are disjoint, the precision of the whole set of association rules is around 70 % and can be calculated as a fraction of all supporting objects.

3.2 Engine E_1—Maximal Current and Current Increase Speed

The similar set of rules for the engine E_1 (Table 2) consists of rules that are weaker than their equivalents for the engine E_3.

This set of rules covers 68.2 % of the data.

Table 1 First set of association rules

Rule	Precision	Match	Supp
IF $v_3 > 3.70$ THEN $M_3 > 79$	0.809	329	266
IF $2.50 \leq v_3 \leq 3.70$ THEN $66 \leq M_3 \leq 79$	0.656	340	223
IF $v_3 \leq 2.50$ THEN $M_3 \leq 66$	0.755	331	250

Table 2 Second set of association rules

Rule	Precision	Left	Both
IF $v_1 \geq 6.00$ THEN $M_1 > 94$	0.786	238	187
IF $4.00 \leq v_1 \leq 6.00$ THEN $76 \leq M_1 \leq 94$	0.632	250	158
IF $2.92 \leq v_1 \leq 4.00$ THEN $64 \leq M_1 \leq 76$	0.596	260	155
IF $v_1 \leq 2.92$ THEN $M_1 \leq 64$	0.722	252	182

Table 3 Third set of association rules

Rule	Precision	Left	Both
IF $v_2 \geq 6.13$ THEN $M_2 > 95$	0.806	248	200
IF $4.25 \leq v_2 \leq 6.13$ THEN $77 \leq M_2 \leq 95$	0.628	250	157
IF $v_2 \leq 3.00$ THEN $M_2 \leq 65$	0.736	258	190

3.3 Engine E_2—Maximal Current and Current Increase Speed

An analogical set of rules for the engine E_2 (Table 3) gives more precise information but covers only 54.7 % of the data.

3.4 Minimal and Maximal Current Values for Engines E_1 and E_2

The rules describing two groups of variables are presented in Tables 4 and 5 respectively.

Table 4 Fourth set of association rules

Rule	Precision	Left	Both
IF $m_1 \leq 35$ THEN $m_2 \leq 36$	0.767	339	260
IF $m_2 \leq 36$ THEN $m_1 \leq 35$	0.583	446	260

Table 5 Fifth set of association rules

Rule	Precision	Left	Both
IF $M_1 > 94$ THEN $M_2 > 95$	0.725	247	179
IF $M_1 \leq 64$ THEN $M_2 \leq 65$	0.688	253	174
IF $M_2 > 95$ THEN $M_1 > 94$	0.740	242	179
IF $M_2 \leq 65$ THEN $M_1 \leq 64$	0.682	255	174

Table 6 Sixth set of association rules

Rule	Precision	Left	Both
IF $v_1 > 6.00$ THEN $v_2 > 6.13$	0.660	238	157
IF $v_2 > 6.13$ THEN $v_1 > 6.00$	0.633	248	157

Although these rules have rather good quality, the coverage of the first set of rules equals 26 % and the coverage of each pair of rules from the second set is 35.3 % (each pair covers the same region of the data).

3.5 Current Consumption Increase Speed—Engines E_1 and E_2

Two association rules (Table 6) join the speed of the current increase of two engines: E_1 and E_2.

It is another situation when we obtain a set of symmetric rules. This set—and the previously presented pairs—should be taken into consideration for the purpose of monitoring the two engines load uniformity.

3.6 Current Consumption Increase Speed and Maximal Current Consumption—Engines E_1 and E_2

The set of very strong rules is inducted from variables v_1, v_2, M_1 and M_2. It is presented in Table 7.

These rules are very strong but also have a smaller coverage. They are also very interpretable—the speed of current increase determines the maximal current consumption, and the maximal current consumption of one engine depends on its increase speed and the maximal current consumption of the other engine (assuming that they are loaded uniformly).

3.7 "Post Factum" Rules—All Engines

The last group of rules (Table 8) describes the association between the value of the established power consumption level, the speed of a current value increase and—what may be surprising—the maximal current value during the startup.

Table 7 Seventh set of association rules

Rule	Precision	Left	Both
IF $v_1 > 6.00 \wedge v_2 > 6.13$ THEN $M_1 > 94$	0.898	157	141
IF $v_1 > 6.00 \wedge v_2 > 6.13$ THEN $M_2 > 95$	0.892	157	140
IF $v_1 < 2.92 \wedge v_2 < 3.00$ THEN $M_2 < 65$	0.820	150	123
IF $M_1 > 94 \wedge v_2 > 6.13$ THEN $M_2 > 95$	0.915	165	151
IF $M_2 > 95 \wedge v_1 > 6.00$ THEN $M_1 > 94$	0.923	155	143
IF $M_1 < 64 \wedge v_2 < 3.00$ THEN $M_2 < 65$	0.928	138	128
IF $M_2 < 65 \wedge v_1 < 2.92$ THEN $M_1 < 64$	0.891	147	131

Table 8 Eighth set of association rules

Rule	Precision	Left	Both
IF $v_3 > 3.70 \wedge est.lvl > 70$ THEN $M_3 > 79$	0.963	135	130
IF $v_1 > 6.00 \wedge est.lvl > 70$ THEN $M_1 > 94$	0.943	123	116
IF $v_2 > 3.70 \wedge est.lvl > 70$ THEN $M_2 > 95$	0.900	130	117

As it is defined at the beginning of a paper, the value of M_x is known before the value of *est. lvl*. However, it occurs that these "post-factum" rules have a very high precision.

4 Proposition of a Diagnostic Procedure

All gathered data, which were the basis of the inducted association rules, come from the so-called proper machine operation. This means that the developed associations describes the normal work of the machine and no data could be used to develop an opposite model on improper machine operation. But, as it was presented in the previous section, even the best of them were not completely accurate.

This remark leads to the conclusion that the further proper operation of the machine should be represented with startups, whose characteristics should not differ too much from the quality of previously inducted association rules. Having only the description of a proper machine operation, we can define a soft margin between the (association rules based) model of a proper operation and a current characteristic of startups.

The diagnostic model is simple and very intuitive: as long as the precision of association rules applied on the new data (e.g. last 100 startups) does not decrease an assumed level, the machine operation is considered proper. The assumed level of 5 % precision decrease does not imply from any premises and does not reflect the nature of the process. The better estimation of this decrease level could be derived

Table 9 Subset of association rules—a "green level" of an operation correctness

Train data		Current operation data		
Rule	Precision/−5 %	Match	Supp	Precision
IF $v_3 > 3.70$ THEN $M_3 > 79$	0.809/0.769	33	26	0.788
IF $2.50 \leq v_3 \leq 3.70$ THEN $66 \leq M_3 \leq 79$	0.656/0.623	34	24	0.706
IF $v_3 \leq 2.50$ THEN $M_3 \leq 66$	0.755/0.717	33	24	0.727

from the data containing proper and improper startups, on the basis of the analysis of false positives and false negatives coming from the model.

Let us consider the first set of rules, describing the behaviour of a machine during proper startups (Table 9). Assuming a 5 % soft margin for the rule precision decrease, original and new—smaller—threshold values of rule precision are presented. During the last 100 startups a specified number of them matched (match) and supported (supp) the following association rules. On the basis of these startups, current quality of rules (precision) can be evaluated.

We can observe that no rules accuracies (evaluated on the new data) exceed the assumed minimal level. It should correspond to the situation of a proper engine E_3 operation (a "green level").

It may happen that not all rules will fulfil a margin defined criterion. This situation can be called "yellow level"—a warning for the operator. The sample data are presented in Table 10.

A fatal situation—when none of the rules achieve the minimal precision—should be reported as the "red level" alert for the operator. The sample data for this situation are presented in Table 11.

Table 10 Subset of association rules—a "yellow level" of an operation correctness

Train data		Current operation data		
Rule	Precision/−5 %	Match	Supp	Precision
IF $v_3 > 3.70$ THEN $M_3 > 79$	0.809/0.769	28	23	0.821
IF $2.50 \leq v_3 \leq 3.70$ THEN $66 \leq M_3 \leq 79$	0.656/**0.623**	35	21	**0.600**
IF $v_3 \leq 2.50$ THEN $M_3 \leq 66$	0.755/0.717	37	27	0.730

Table 11 Subset of association rules—a "red level" of an operation correctness

Train data		Current operation data		
Rule	Precision/−5 %	Match	Supp	Precision
IF $v_3 > 3.70$ THEN $M_3 > 79$	0.809/**0.769**	28	20	**0.714**
IF $2.50 \leq v_3 \leq 3.70$ THEN $66 \leq M_3 \leq 79$	0.656/**0.623**	35	21	**0.600**
IF $v_3 \leq 2.50$ THEN $M_3 \leq 66$	0.755/**0.717**	37	26	**0.703**

5 Conclusions

In the paper the authors presented the application of the correlation analysis and the association analysis for the evaluation of the conveyor diagnostic state. On the basis of over eight weeks of the proper conveyor work observation some dependencies in the data were found. These dependencies are described as high values of statistically significant Pearson correlation coefficients and association rules.

On the basis of the association rules a new way to estimate the diagnostic state of the wall conveyor was proposed. The obtained diagnostic procedure bases on the assumption that only the proper conveyor work is described and the deviation from this behaviour is considered as an improper diagnostic state.

Because the given observations of the proper work made it possible to induct quite good association rules describing this diagnostic state, our next goal is to observe the same conveyor working under the same conditions in the proper and improper way. The induction of association rules describing two mentioned diagnostic states should give more precise models of making decisions about the state of the device.

Acknowledgments The work was financially supported by POIG.02.03.01-24-099/13 grant: GeCONiI "Upper Silesian Center for Computational Science and Engineering". The participation of the second author was supported by Polish National Centre for Research and Development (NCBiR) grant PBS2/B9/20/2013 in frame of Applied Research Programmes.

References

1. Bartelmus, W.: Condition monitoring of open cast mining machinery. Wroclaw University of Technology Press, Wroclaw (2006)
2. DEMKop System: http://www.somar.com.pl/katalog-wyrobow/demkop,18,2,71
3. Gąsior, S.: Diagnosis of longwall chain conveyor. Przegląd Górniczy **57**(7–8), 33–36 (2001)
4. Głowacz, A.: Diagnostics of direct current machine based on analysis of acoustic signals with the use of symlet wavelet transform and modified classifier based on words. Eksploatacja i Niezawodność—Maint. Reliab. **16**(4), 554–558 (2014)
5. Głowacz, A., Głowacz, A., Korohoda, P.: Recognition of monochrome thermal images of synchronous motor with the application of binarization and nearest mean classifier. Arch. Metall. Mater. **59**(1), 31–34 (2014)
6. Kacprzak, M., Kulinowski, P., Wędrychowicz, D.: Computerized information system used for management of mining belt conveyors operation. Eksploatacja i Niezawodność—Maint. Reliab. **13**(2), 81–93 (2011)
7. Karabadji, N., Seridi, H., Khelf, I., Azizi, N., Boulkroune, R.: Improved decision tree construction based on attribute selection and data sampling for fault diagnosis in rotating machines. Eng. Appl. Artif. Intell. **35**, 71–83 (2014)
8. Michalak, M., Sikora, M., Sobczyk, J.: Analysis of the longwall conveyor chain based on a harmonic analysis. Eksploatacja i Niezawodność—Maint. Reliab. **15**(4), 332–336 (2013)
9. Siciński, K., Isakow, Z., Oset, K.: Monitoring of longwall coal-cutting machines' work to improve the efficiency of the wall extraction. Mech. Autom. Min. Ind. **381**(9), 113–120 (2002)

10. Webb, G.: Discovering associations with numeric variables. In: SIGKDD 2001, San Francisco, USA, pp. 383–388 (2001)
11. Webb, G.: Discovering significant patterns. Mach. Learn. **68**(1), 1–33 (2007)
12. Wong, P., Yang, Z., Vong, C., Zhong, J.: Real-time fault diagnosis for gas turbine generator systems using extreme learning machine. Neurocomputing **128**, 249–257 (2014)

Supporting the Forecast of Snow Avalanches in the Canton of Glarus in Eastern Switzerland: A Case Study

Sibylle Möhle and Christoph Beierle

Abstract Snow avalanches pose a serious threat in alpine regions. They may cause significant damage and fatal accidents. Assessing the local avalanche hazard is therefore of vital importance. This assessment is based, amongst others, on daily collected meteorological data as well as expert knowledge concerning avalanche activity. To a data set comprising meteorological and avalanche data collected for the Canton of Glarus in Eastern Switzerland over a period of 40 years, we applied different machine learning strategies aiming at modeling a decision support system in avalanche forecasting.

Keywords Decision support · Avalanche forecasting · Uncertain reasoning · Man-machine interaction

1 Introduction

Snow avalanches endanger, amongst others, traffic infrastructure and may cause significant damage and fatal accidents. Thus, assessing the local risk of snow avalanches is of vital importance. For this reason, local avalanche services have been established in alpine countries. Their task is to protect people from the impact of snow avalanches by temporary measures, like the closing of roads not protected by physical structures, ordering people to stay in buildings, evacuation, or artificial avalanche triggering [19].

Precipitation (new snow or rain), wind, air temperature, and solar radiation are the main factors influencing the formation of avalanches. Local avalanche forecasters

S. Möhle (✉)
International Center for Computational Logic, TU Dresden, Dresden, Germany
e-mail: sibylle.moehle@tu-dresden.de

S. Möhle · C. Beierle
Department of Computer Science, University of Hagen, Hagen, Germany
e-mail: beierle@fernuni-hagen.de

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_38

base their daily judgment of the avalanche danger on a careful analysis of meteorological variables and snowpack properties influencing the stability of the snowpack. This assessment relies heavily on a sound understanding of the physical processes in the snowpack, but also on experience and comparison with similar situations observed in the past since a similar avalanche activity might take place in a specific situation. However, meteorological data collected on consecutive days may be very similar and distinguishing an avalanche day from a non-avalanche day typically comes with a high level of uncertainty.

Meteorological and snow data are collected daily by automatic and manual stations. They are visualized in the Intercantonal Early Warning and Crisis Information System (IFKIS) [3] operated by the WSL Institute for Snow and Avalanche Research SLF in Davos, Switzerland. Being conceived as an information system, IFKIS provides no avalanche forecast. Decision support systems such as NXD2000 [7, 8] which are based on the method of nearest neighbors [4] help local avalanche forecasters to base their decisions on more objective criteria, in addition to their expert knowledge and experience. NXD2000 returns the ten days in the database which are most similar to the situation being assessed and the avalanche activity that occurred within the corresponding time slots. Interpretation of the presented data is left to the user, and region-specific knowledge thereby plays a decisive role. Decision support systems which are trained for a specific region provide valuable evidence with regard to the avalanche activity in cases where little experience or region-specific knowledge is available.

Classification and regression trees [2] were applied for forecasting large and infrequent snow avalanches in Eastern Switzerland [17] as well as for predicting avalanches in coastal Alaska [12]. The suitability of classification trees and Random Forests for predicting wet-snow avalanches in the region of Davos in Eastern Switzerland was investigated [15]. The suitability of Random Forests and variants thereof for avalanche prediction in the Canton of Glarus in Switzerland was assessed [16]. However, avalanche forecasting is a highly region-specific task, and to our knowledge, no other work concerning avalanche forecasting in the region of the Canton of Glarus was carried out. Our aim therefore was to model a decision support system for avalanche forecasting in the Canton of Glarus. In the data used in [16], avalanches represent rare events, making their exact characterization difficult and uncertain. Using Balanced Random Forest and Weighted Random Forest [5], feasible models were trained [16].

The purpose of this work is to report on a case study where we applied and evaluated different machine learning strategies to the data used in [16] in order to achieve avalanche forecasting for the involved region. In a first scenario, we investigate the impact of variable selection on the model performance as proposed in [16]. A wrapper method combined with a backward selection was employed. A second goal was to test whether oversampling the positive class results in a better model performance than the models developed in [16] which manifested difficulties in distinguishing avalanche days from non-avalanche days. A third objective was to investigate the suitability of a naïve Bayes classifier, despite the fact that for the given scenario, the independence assumptions inherent to naïve Bayes are clearly violated.

The rest of this paper is organized as follows: In Sect. 2, we describe the available data and their processing. Comparison of different models by means of ROC analysis is addressed in Sect. 3. The oversampling process and appropriate results are presented in Sect. 4. Variable selection is described and results obtained are presented in Sect. 5. In Sect. 6, we briefly address the main characteristics of naïve Bayes classifiers and present the appropriate results. In Sect. 7, our results are discussed. In Sect. 8, we conclude and point out further work.

2 Data

The Canton of Glarus is situated in Eastern Switzerland. It is characterized by high mountains and steep slopes. In this work, we focused on the alpine valley Kleintal situated in the southeast of the Canton of Glarus. The valley floor is gently inclined, its elevation ranging from over 600 m.a.s.l. to over 1000 m.a.s.l. The starting zone of a snow avalanche may be situated up to 1700 m above the valley floor and may therefore endanger the main road leading through the valley.

The data used in our case study consist of meteorological variables measured daily in the early morning as well as information regarding avalanches which endangered the main road. The meteorological variables are collected by two measure stations and are recorded in the database by the local avalanche service of the Canton of Glarus. The collected variables comprise the maximum and minimum air temperature in the last 24 h, actual wind speed and actual wind direction, degree of sky cover and precipitation in the last 24 h as well as snow depth and new snow depth in the last 24 h.

Meteorological factors are potentially useful for estimating snowpack instability, but interpretation is uncertain and the evidence less direct than for snowpack factors [14]. Avalanche expert knowledge was taken into account by using the derived variables which actually are used for assessing the local avalanche danger in the Canton of Glarus. They are listed in Table 1. A variable *avalanche* was introduced and assigned the value 1 if the corresponding entry represented an avalanche day and the value 0 otherwise. By this means, *avalanche* denotes the class of the corresponding entry.

There are 7 avalanche paths which pose a particular danger for the main road. For each of these, 7 to 13 avalanches endangering the main road in the period considered were recorded, thus resulting in a total of 53 events. Due to the lack of data, we did not discriminate between avalanche paths, and days with more than one avalanche triggered in these avalanche paths were considered as one event. Our final data consist of daily records over a period of more than 40 winter seasons of approximately 181 days each between January 1, 1972, and April 30, 2013. Due to missing data, small data gaps may be present in some of the seasonal series. The data contained 6889 non-avalanche days and 53 avalanche days and therefore the ratio of positive to negative examples was approximately 1:130. We divided the data into a training and

Table 1 Meteorological variables are measured daily

	Variable
1	Max. air temperature in the last 24 h
2	Max. air temperature in the last 48 to 24 h
3	Min. air temperature in the last 24 h
4	Min. air temperature in the last 48 to 24 h
5	Actual wind direction
6	Wind direction of the previous day
7	Actual wind speed
8	Wind speed of the previous day
9	Degree of sky cover
10	Precipitation in the last 24 h
11	Precipitation in the last 48 to 24 h
12	New snow fallen in the last 24 h
13	New snow fallen in the last 72 to 24 h
14	Snow depth

The definition of derived variables allows consideration of expert knowledge about avalanche activity

a test data set as follows: All records until April 30, 2002, were labeled as training data, the other records as test data. That way the ratio of positive to negative examples in the test data corresponds to the one observed over the period considered.

3 Model Comparison

Forecasting an avalanche is equivalent to predicting the value of the variable *avalanche*. Therefore, avalanche forecasting can be considered as a classification problem, and the results can be represented in a contingency table. The number of true negative forecasts is denoted by *TN*. It refers to the cases in which neither an avalanche was predicted nor an avalanche occurred. The number of false negative forecasts is denoted by *FN* and refers to the number of missed avalanche days. In these cases, an avalanche was released whereas none was predicted and therefore no measures were taken, i.e., the road was not closed. Hence, fatal accidents may occur. Not all avalanches recorded reached the road, nevertheless the number of false negative forecasts should be minimized. The number of false positive forecasts is abbreviated as *FP*. It refers to situations in which a predicted avalanche did not occur. While no fatal accidents occur in this case, the number of false positive forecasts should be minimized in order to avoid costs caused, e.g., by road closures, and to prevent loss of credibility of the local avalanche service. The number of correctly predicted avalanche days is denoted by *TP* and refers to situations in which a predicted avalanche was released.

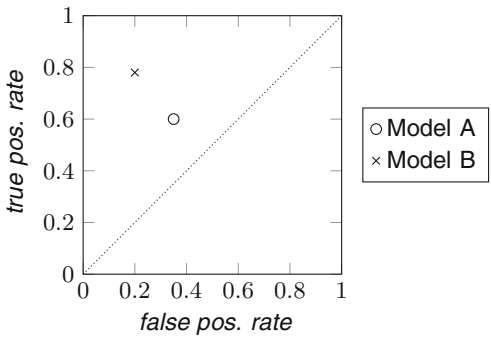


Fig. 1 In the ROC space, the performance of two classifiers may be compared visually. The point representing classifier B is located in the northwest of the point representing classifier A. Therefore classifier B performs better than classifier A

For our purposes, the true positive rate $tpr = \frac{TP}{TP+FN}$ and the false positive rate $fpr = \frac{FP}{TN+FP}$ are relevant. The true positive rate *tpr* denotes the proportion of correctly predicted avalanche days. Analogously, the false positive rate *fpr* represents the proportion of wrongly classified non-avalanche days. By means of these two values, the performance of a classifier may be assessed and visualized in a receiver operating characteristics (ROC) graph. The ROC graph is a two-dimensional graph in which *fpr* is plotted on the X axis and *tpr* is plotted on the Y axis [6]. A discrete classifier, i.e., a classifier predicting a class without issuing any probabilities, produces a point in the ROC space. For an ideal classifier, this point is situated in the upper left corner. Points situated on the diagonal line $fpr = tpr$ refer to classifiers randomly predicting the class, while points situated below the diagonale represent classifiers whose classification is inverted. If for these classifiers a positive prediction is issued instead of a negative one and vice versa, a classifier is obtained which in the ROC space is represented by a point above the diagonale and therefore performs better than a random classifier. Two classifiers may be compared visually by analyzing the position of their points in relation to each other. A classifier performs better than another, if its point is located in the northwest of the other classifier's point [6], since it has a lower false positive rate as well as a higher true positive rate, i.e., it issues less false alarms and detects more events. Hence, in Fig. 1, classifier B is better than classifier A.

4 Oversampling the Positive Class

Method The data described above was used in [16] to build a decision support system for snow avalanche warning. As pointed out in [16], a particular obstacle was that, in the given data, avalanches are rare events. The positive class gains weight during the learning phase if oversampling by means of introducing duplicates of positive examples is employed. This procedure bears the risk of learning specific examples [18]. Our aim was to investigate the performance of the model depending on the

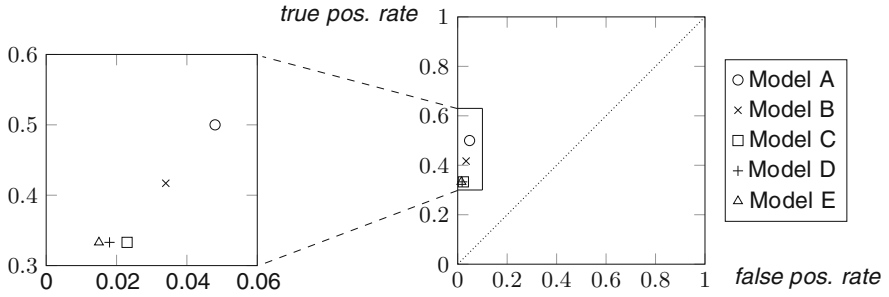


Fig. 2 Oversampling the positive class: Model A was generated without replication of the positive examples. For models B, C, D, and E the positive examples were replicated 1, 2, 3, and 4 times, respectively. On the left-hand side, a zoomed section is shown

number of replications of positive examples, i.e., the avalanche days. As a learning algorithm, we employed Balanced Random Forest [5].

From the training data, we generated four additional training data sets in which the positive examples were replicated 1, 2, 3, and 4 times, respectively. For each of these data sets, we trained Balanced Random Forests according to [16] with different parameter combinations.

Results A fundamental behaviour is observed when altering the number of replications while fixing the cutoff and the number of variables to be tested. In Fig. 2, Model A was generated without replicating positive examples and hence represents the model learned in [16]. For models B, C, D, and E the positive examples were replicated 1, 2, 3, and 4 times, respectively. An increase in the number of replications of positive examples leads to an increase in the number of false negatives as well as a decrease in the number of false positive forecasts, i.e., the more replications of positive examples are made, the less often test examples are classified positive. This behaviour is reflected in the position of the points which the various models produce in the ROC space (see Fig. 2). Replication of the positive examples did not significantly improve the performance of the model.

5 Variable Selection

Method The prediction accuracy of a decision tree based algorithm may suffer if many variables are present which are irrelevant for the prediction [13]. In [9], different variable selection strategies are discussed. The main idea is to investigate models learned using different subsets of variables and to determine the optimal subset in order to train a classifier with maximum performance. This approach may be considered equivalent to choosing the variables which most contribute to the prediction.

Alg. 1 Variable selection by means of a wrapper method using backward elimination.

1: $V \leftarrow$ set of all variables listed in Table 1
2: $M \leftarrow$ model trained using V
3: $v \leftarrow$ variable with lowest importance
4: $V' \leftarrow V \setminus \{v\}$
5: $M' \leftarrow$ model trained using V'
6: **while** $performance(M') \geq performance(M)$ **do**
7: $M \leftarrow M'$
8: $V \leftarrow V'$
9: $v \leftarrow$ variable with lowest importance
10: $V' \leftarrow V \setminus \{v\}$
11: $M' \leftarrow$ model trained using V'
12: **end while**
13: **return** V

Wrapper methods use the learning algorithm as a black box for building models for the investigated variable subset.

Considering the fact that avalanche days in our data represent rare events, a wrapper method using Balanced Random Forest as learning algorithm appeared promising. Variable importance may be assessed, therefore providing a valuable criterion for variable selection in combination with backward elimination. The procedure is illustrated in Algorithm 1. The variable set V is initialized with the set containing all variables listed in Table 1 (line 1). A model is trained, the variable v with lowest importance is determined and removed from V (lines 2–4) thus resulting in a new variable set V' which is used for generating a new model M' (line 5). This procedure is repeated iteratively until no model with a higher performance results (lines 6–12).

Results We performed variable selection according to Algorithm 1 on our models obtained with different parameter settings. The resulting variable subsets showed slight differences, and the order in which the variables were removed differed among the various tests. However, in all tests, wind and degree of sky cover proved to be least important while, with one exception, the precipitation in the last 24 h, the amount of new snow fallen in the last 24 h, the amount of new snow fallen in the last 72 to 24 h as well as the snow depth proved to be most important. The position of the resulting models in the ROC space is visualized in Fig. 3. No significant improvement in model performance was achieved using variable selection by means of Algorithm 1.

6 Naïve Bayes Classifier

Method Naïve Bayes classifiers are based on the assumption that the variables are conditionally independent given the class. For calculating the probability of each class for a given example, Bayes' theorem is adopted. Even in situations in which this assumption does not hold, naïve Bayes classifiers often outperform more sophisticated algorithms [11]; reasons for this observation are given in,

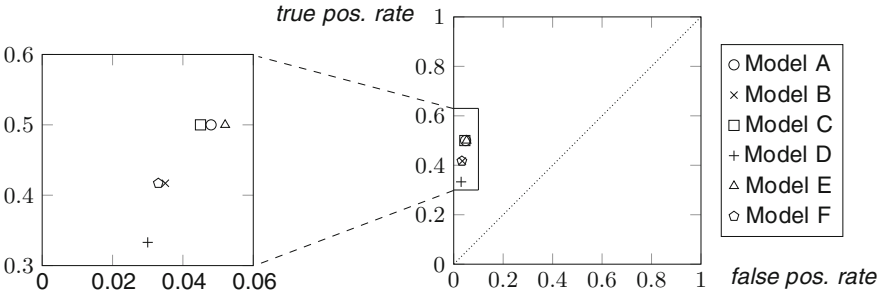


Fig. 3 Variable selection: Models A and B correspond to models learned in [16]. Models C, D, E, and F were trained using variable subsets determined by means of Algorithm 1

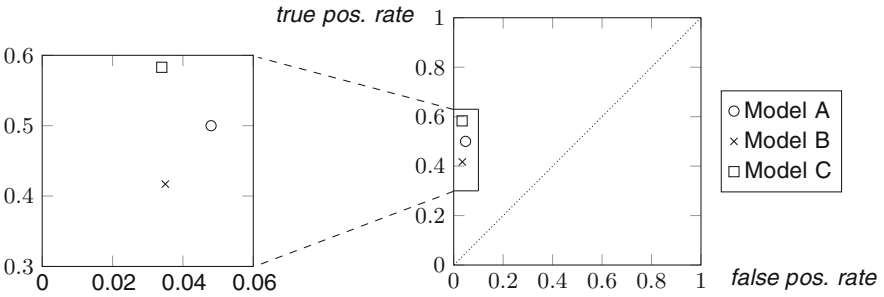


Fig. 4 Naïve Bayes classification: Models A and B correspond to original models learned in [16]. Model C is the model trained using a naïve Bayes classifier

e.g., [1, 10]. Therefore, we also applied a naïve Bayes classifier to the avalanche prediction problem.

Results The independence assumption is clearly violated, nevertheless the naïve Bayes classifier clearly outperformed the models trained in [16] (see Fig. 4).

7 Discussion

Oversampling the Positive Class In the Balanced Random Forests trained without oversampling the positive class, the true positive rate is moderate and the false positive rate is low. The number of false positive forecasts decreases considerably as the number of replications of the positive examples is increased. At the same time, a decrease in the number of recognized avalanche days is observed. Model performance is comparable to the one obtained for the models developed in [16]. Therefore oversampling the positive class provided no significant improvement with regard to model performance.

Variable Selection The performance of the models trained using a variable subset is comparable to the performance of the models developed in [16]. Hence, variable selection as described in Algorithm 1 had no considerable effect on the quality of the learned model. From our tests, it follows that taking wind and degree of sky cover into account does not noticeably contribute to the classification outcome. They do not deteriorate the results, either. Therefore, wind direction and wind speed as well as the degree of sky cover could either be removed from or be retained in the variable set.

In our data, one entry for wind direction and wind speed per day is available. For assessing the local hazard of snow avalanches, however, the time period during which the wind blows from the same direction is important because of the snow drift it may cause which might promote avalanche release. Unfortunately, this information can not be derived from our data. This might be one reason for the low importance wind speed and wind direction achieved during variable selection. When assessing the avalanche hazard, local avalanche forecasters told us that they use relevant information regarding wind data from other sources; unfortunately, these data have not been recorded in the past and thus were not available for our case study.

Analyzing the values for the degree of sky cover may explain why this variable achieved a low importance in all tests. The sky is overcast in 43 % of all data records. A clear sky appears second most in our data accounting for a total of 19 % of all data records. For the test and training data set, similar values are observed. In the training data set, 83 % of the avalanches were triggered with an overcast sky and about 2 %, i.e., exactly one, with a clear sky. In the test data set, 75 % of the avalanches were released with an overcast sky and no avalanche was triggered with a clear sky. Although most avalanches occurred with an overcast sky, this observation does not allow a clear differentiation of positive and negative examples. This may be due to the amount of negative examples with an overcast sky in general as well as the similarity of meteorological variables collected at consecutive days mentioned in Sect. 1. From the available data, however, one might conclude that with a clear sky the triggering of an avalanche is unlikely.

Naïve Bayes Classifier Our data consist of more than 40 time series of meteorological variables. Therefore, the individual entries for some variables in a time series are correlated, for example, the snow depths of two consecutive days can not differ arbitrarily. Additionally, some of the variables are not conditionally independent given the class, and therefore the independence assumption is clearly violated. Despite these facts, the naïve Bayes classifier performed better than any of the models developed in [16] and the other models presented here, and the fact that avalanche days represented rare events apparently did not affect the result.

Feasibility of the Models Considering the length of the time period comprised in the test data, the number of false forecasts is acceptable. Due to the fact that avalanche days in our data represent rare events, one unique missed avalanche results in a significant decrease in the true positive rate. An expert of the local avalanche service of the Canton of Glarus confirmed that the discussed models are feasible as a decision support in avalanche warning since the misclassification rate is comparable to that

of an human expert. As no records concerning human forecasts are available, this statement is based on the self-assessment of the avalanche forecaster in charge of avalanche warning in the Canton of Glarus.

8 Conclusions and Further Work

In alpine regions, assessing the local risk of snow avalanches is of vital importance and requires expert knowledge, intuition, and process understanding. Moreover, since avalanche forecasting is a highly region-specific task, local knowledge is essential and generalization to other locations is very difficult. Using meteorological and avalanche data collected daily over a period of 40 years for the Canton of Glarus in Switzerland, in this paper we reported on a case study where we applied and evaluated different machine learning strategies aiming at modeling a decision support system in avalanche forecasting. Oversampling the positive class resulted in a significant decrease of both the number of false positive forecasts and the number of true positive forecasts. Variable selection performed as described in Algorithm 1 did not considerably affect the performance of the model. The naïve Bayes classifier altogether performed best.

There are several directions in which the work presented here should be extended. The decision concerning the practicality of a model will merely depend on the type of avalanches missed and the specific situation, thus the misclassified examples should carefully be examined in more detail. Furthermore, alternate variable selection methods should be investigated, and the definition of more meaningful variables derived from the given data could prove beneficial. In particular, alternate data sources containing more detailed information concerning wind speed and wind direction should be looked for. Further improvements of the models could be achieved by integrating snowpack characteristics such as stratigraphy or temperature gradients in the snowpack.

As stated by avalanche professionals, models will remain only an additional criteria for decision making. They provide useful support in cases where little region-specific knowledge or experience is available. The main difficulty in avalanche forecasting is due to the similarity of two consecutive days regarding their meteorological variables. In these cases, prediction models might provide valuable support in distinguishing an avalanche day from a non-avalanche day. The final decision with respect to the forecast (e.g., to close or not to close a road) is still left to the user, but further improvements of these models will strengthen their role in the future.

Acknowledgments The authors wish to thank the avalanche service of the Canton of Glarus, Switzerland, and the WSL Institute for Snow and Avalanche Research SLF in Davos, Switzerland, for providing the data on which this work is based.

References

1. Bishop, C.M.: Pattern Recognition and Machine Learning, vol. 1. Springer, New York (2006)
2. Breiman, L., Friedman, J., Stone, C.J., Olshen, R.A.: Classification and Regression Trees. CRC Press, Boca Raton (1984)
3. Bründl, M., Etter, H.J., Steiniger, M., Klingler, C., Rhyner, J., Ammann, W.: IFKIS—a basis for managing avalanche risk in settlements and on roads in Switzerland. Nat. Hazards Earth Syst. Sci. **4**(2), 257–262 (2004)
4. Buser, O., Bütler, M., Good, W.: Avalanche forecast by the nearest neighbour method. Int. Assoc. Hydrol. Sci. **162**(2), 557–570 (1987)
5. Chen, C., Liaw, A., Breiman, L.: Using Random Forest to Learn Imbalanced Data. Technical Report, University of California, Berkeley (2004)
6. Fawcett, T.: An introduction to ROC analysis. Pattern Recognit. Lett. **27**(8), 861–874 (2006)
7. Gassner, M., Brabec, B.: Nearest neighbour models for local and regional avalanche forecasting. Nat. Hazards Earth Syst. Sci. **2**, 247–253 (2002)
8. Gassner, M., Etter, H.J., Birkeland, K., Leonard, T.: NXD2000: An improved avalanche forecasting program based on the nearest neighbor method. In: ISSW 2000. pp. 52–59. Big Sky, Montana (2000)
9. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. J. Mach. Learn. Res. **3**, 1157–1182 (2003)
10. Hand, D.J., Mannila, H., Smyth, P.: Principles of Data Mining. MIT press, Cambridge (2001)
11. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference and Prediction. Springer Series in Statistics, 2nd edn. Springer, New York (2009)
12. Hendrikx, J., Murphy, M., Onslow, T.: Classification trees as a tool for operational avalanche forecasting on the Seward Highway, Alaska. Cold Reg. Sci. Technol. **97**, 113–120 (2014)
13. Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artif. Intell. **97**(1), 273–324 (1997)
14. McClung, D., Schaerer, P.A.: The Avalanche Handbook. The Mountaineers Books, Seattle WA (2006)
15. Mitterer, C., Schweizer, J.: Analyzing the atmosphere-snow energy balance for wet-snow avalanche prediction. In: ISSW 2012. pp. 77–83. Big Sky, Montana (2012)
16. Möhle, S., Bründl, M., Beierle, C.: Modeling a system for decision support in snow avalanche warning using balanced random forest and weighted random forest. In: Agre, G., Hitzler, P., Krisnadhi, A., Kuznetsov, S. (eds.) Artificial Intelligence: Methodology, Systems, and Applications. LNCS, vol. 8722, pp. 80–91. Springer, Switzerland (2014)
17. Schweizer, J., Mitterer, C., Stoffel, L.: On forecasting large and infrequent snow avalanches. Cold Reg. Sci. Technol. **59**(2), 234–241 (2009)
18. Weiss, G.M.: Mining with rarity: a unifying framework. ACM SIGKDD Explor. Newsl. **6**(1), 7–19 (2004)
19. WSL-Institut für Schnee- und Lawinenforschung SLF, Bundesamt für Umwelt BAFU, Schweizerische Interessengemeinschaft Lawinenwarnsysteme (SILS): Praxishilfe. Arbeit im Lawinendienst: Organisation, Beurteilung lokale Gefährdung und Dokumentation. http://www.slf.ch/dienstleistungen/merkblaetter/praxishilfe_lawdienst_deutsch.pdf

Geospatial Data Integration for Criminal Analysis

Kamil Piętak, Jacek Dajda, Michał Wysokiński, Michał Idzik and Łukasz Leśniak

Abstract The aim of the paper is to discuss the problem of geospatial data integration for criminal analysis. In order to integrate and analyze various data sources the platform introduces an object-based data model for each analyzed domain. The paper focuses on the model for geospatial analysis and integration methods that allow to visualize and analyze various data on geographical map. To verify the realized concept, a simple case study is given as an example of the integration results.

Keywords Geospatial criminal analysis · Data integration

1 Introduction

Nowadays, along with the rapid technological development one can observe a growing number of devices and systems producing data with geographical context such as GPS trackers, cameras, mobile phones, tablets, street monitoring systems, mobile operator centers and others. This kind of data can be used by security authorities for various analytical purposes.

The large number of data sources with geospatial context, while promising for the quality of final results, poses new challenges related to integration of data coming from various sources, formats heterogeneity, data size, and proper visualization

K. Piętak (✉) · J. Dajda · M. Wysokiński · M. Idzik · Ł. Leśniak
AGH University of Science and Technology, Krakow, Poland
e-mail: kpietak@agh.edu.pl

J. Dajda
e-mail: dajda@agh.edu.pl

M. Wysokiński
e-mail: michal.wysokinski@agh.edu.pl

M. Idzik
e-mail: idzik@iisg.agh.edu.pl

Ł. Leśniak
e-mail: lesniak@iisg.agh.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_39

techniques. To support the analysis, different software tools and packages are utilized, however not without problems and compromises.

In most cases, the available tools (usually denoted as GIS—Geographical Information Systems) are of general purpose which requires from analysts advanced knowledge as well as manual work to be performed on data preparation and processing. Some of them develop their own dedicated applications but they are limited and not extensible. Another problem refers to the fact that every analysis and analyst is different. This requires from tools to be used in a different way and allow for further extensions which may be useful when new data sources or techniques are discovered. Finally, geospatial analysis is always a part of larger analytical process and that is why provided support must provide data integration.

The aim of this paper is to present the concept of a tool-set and a flexible and extensible framework for geo-spatial criminal analysis. The technical aspects of the concept will be illustrated with sample implementation which is *LINK Map* component built upon LINK Platform[1]—plug-in-based environment designed for supporting criminal analysis. The similar approaches are also partially implemented in shape of web application for information management and decision support system for the Government Protection Bureau.

This paper is organized as follows: next section introduces the needs of geo-spatial criminal analysis and challenges it faces. A few popular GIS analysis tools are presented as well. Section 3 describes an architecture and crucial concepts of LINK environment. In Sect. 4 a new data model for geo-spatial analysis is presented. In Sect. 5 the verification of proposed solution is presented based on an example case. Than conclusions of the study and further work is described.

2 Criminal Analysis in Concepts and Existing Solutions

There is a number of software solutions and techniques that can be used during geospatial analysis. However, they are often adjusted to strategic analysis and planning or provide limited analytical features.

The most known GIS related technique is crime-mapping [9] which assumes visualization of crime locations on a map. This technique can be further explored into hot-spot analysis [12] and crime indicators calculations and spatial statistics [5]. With the recent growth of mobile devices' popularity special analysis methods have been designed to focus on that kind of data and utilize it to full advantage [10].

Large quantities of data, which are generated every day allowed to not only analyze the past behavioral and crime patterns, but also to extrapolate them and even to try to simulate them [13].

[1]LINK Platform is developed at AGH-UST in Krakow as an environment for building software tools supporting polish criminal analysts. More about the platform can be found at https://www.fslab.agh.edu.pl/#!product/link2. LINK Map is a tool-set that provides graphical components for integration of geo-spatial data, their visualization, analytical tools and extensions for gathering data in various formats.

Some of these ideas have incorporated into commercial solutions like IBM i2 Analyst's Notebook [6] or Palantir [7]. IBM's application is a very complex tool which allows advanced data analysis and pattern discovery. Unfortunately its GIS analytical capabilities are practically non existent and limited to only data visualization on a map. Palantir on the other hand includes some spatial analytical functions, however it is not extendable and similarly to Analyst's Notebook very expensive.

All of the presented concepts require building a model that represents a specific aspect of data. This approach is well-suited for data from a specific period of time which is processed in predefined flows. This kind of analysis can be achieved in popular GIS applications such as ArcGIS [4]. However, because the fact that ArcGIS is a generic purpose tool (like many others GIS solutions), these types of analyses require from end-user some portion of specialized knowledge, manual operation as well as data in a proper format. This is a considerable drawback in a dynamic environment where each analysis differs in some way from previous ones.

Another option is to utilize more lightweight tools such as GoogleMaps or GoogleEarth; however, these tools provide only basic visualization functions and require active Internet connection to download maps, which is unacceptable in most analytical cases.

The available software fails to provide satisfactory support in terms of analytical functions dedicated to criminal analysis. It is because on contrary to strategic analysis, criminal analysis is much more closer to operation activities such as objects tracking and therefore more extendable environment is needed which would allow for fast introduction of new function. In addition, the analysis is more discrete and its goal is to find specific information or correlations between given objects or events. In this case visualization on maps is not the main purpose of the analysis. For example solving such cases as discovering potential meeting points between two phone numbers based on their phone billings requires a series of calculations to look through all the positions and calculate the distances. This can be easily achieved in simple programming function and the visualization aspect is only needed to show the obtained results.

Another problem which is not well-covered in existing tools is the problem of data import. Before any analysis can be performed, the analytical environment has to load data from external sources. In criminal intelligence data can come from various sources, which means they contain information from different domains and are stored in varied formats (e.g. files such as XLS, CSV; data bases, unstructured text documents). In such cases, process of data loading is not trivial and requires tools that allow user to map source data to models defined in analytical environments. This process also requires support especially in cases when the data needs to be analyzed fast, for example in cases of objects monitoring or terrorist attack.

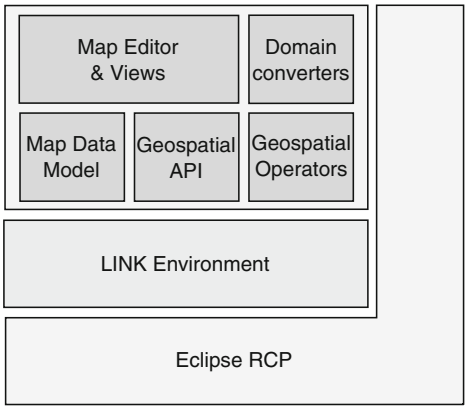


Fig. 1 Overview of LINK Map architecture

3 Framework Concept and Design

To answer the described problems, an architecture described in [2] was proposed. Further in this chapter key elements of LINK platform are shortly described. They are required to understand new concepts introduced in LINK Map that are described in the next chapter.

3.1 Architecture

LINK Map is an extensions of LINK platform—an integrated environment supporting criminal analysis [1, 2]. The platform utilizes plug-in based architecture, built on top of Eclipse RCP environment[2] as shown in Fig. 1. Besides plug-in concept, LINK utilizes *extension points* mechanism together with *OSGi services* to build integrated environment from available modules.

LINK environment provides itself a base for building analytical software tools. It provides abstract data model that is a base for domain models suitable for data coming from various sources. The main concept of the model is a data set comprised of analytical elements and operators that can create or manipulate existing data sets. This allows for building data flows, which usually begin from data import process, through analytical and visualization operations, to creating final reports (data flows are described in more details in Sect. 3.2). LINK provides also a graphical importer that is designed to lead and support a user during the process by: recognizing data types (such as geographical coordinates, dates), providing graphical representation of data models; live validation and import templates that allow to perform easily import process on the same source types.

[2] http://wiki.eclipse.org/Rich_Client_Platform.

Above LINK environment, map-related components are located. They provide a model designed for geographical data (widely described in Sect. 4) together with built-in set of analytical operators. Based on that a graphical editor is built—it presents various perspectives of geographical data on map tiles as well as results of analytical operators (e.g. showing correlations of data).

The next component in LINK Map—domain-specific operators—provides set of tools which convert data from various domains (e.g. data from GPS trackers, events data bases, phone call billings) to map diagrams showing different perspectives suitable for particular cases. Such operators allow also for integration of data coming from different domains, i.e. one map diagram can present multi-domain data and, moreover, such data can be processed by the same analytical operators (e.g. one can find time and geographical correlations between crime scenes, phone call billings (like in [8], data about POIs (Points Of Interest) and so on).

3.2 Data Processing in LINK Platform

LINK Map, containing geospatial analysis, is a part of LINK platform—it especially utilizes LINK data processing model that is widely described in [1, 2]. However to better understand how geospatial extensions work, the model is here shortly introduced.

The LINK data processing model is built on two key notions: *data models* and *operators*.

LINK data model Data models are described in a shape of data sets (*IDataSet* interface) that are collections of logically related model elements. LINK distinguishes three categories of data models shown in Fig. 2:

- *domain models* related to specific domains of source data such as GPS tracks, money transfers data, phone call billings or data about POIs,
- *intermediate/analysis models* created as results of particular analysis operators, used to show intermediate results that can be used in further analysis or visualization,

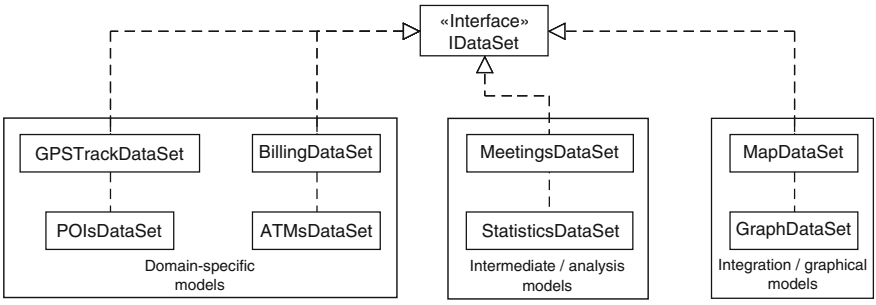


Fig. 2 Categories of data models in LINK environments

- *integration/graphical models* which allow to integrate data coming from different domains in one place and further show different perspectives such as relation between objects (graph model), geospatial or time contexts; they usually are related also to graphical editors that visualize particular aspects.

Data Flow Based on Operator Concept To allow data transformations a concept of operators was applied. Operators acts on data sets of various types and by combining them one can create data flows which can be repeated on different data or tracked for further review of performed analyses.

Operators can be represented as blocks of configurable functions in stream of data processing. Each block receives an input data sets together with optional configuration and than forms an output result (especially, another data set for further analysis).

Declarative Types In criminal analysis the flexibility of data models is crucial—analysts need to adjust models to various data they have from external sources. So the element types defined among data sets should be dynamic which means end-users can modify data types in run-time. To allow this *declarative type mechanism* has been introduced in LINK platform. It assumes that each element in a data set is bound to specific declarative type, called *c-type* (from *custom type*).

C-types equip elements with additional features, add semantic shape and give them specified meaning. Each c-type has unique text id and a set of configurable domain specific properies. Each property has a type (e.g. integer, text, date) and associated validators that assures type-safety in run-time.

4 Data Model for Geo-Spatial Analysis

The main challenge in preparing the map model was to create a solution able to link visualization aspects with domain specific information in context of criminal intelligence. The former aspects contain information about how to visualize geo-spatial data on a map, the latter describe various types of analytical objects. Their role is to integrate various domains at one map diagram. For example a diagram could have events represented as points on a map, but particular events represent different types of objects such as meeting, phone call, car accident or crime scene.

The data model assumes that visual and semantic layers are processed separately, and provides a wide range of map presentation elements that can be connected to predefined domain types. Such approach allows to integrate on one map data coming from heterogeneous sources with different set of attributes and visualize and analyze them in common way.

The model is represented in a shape of map data sets (*IMapDataSet*—a sub-interface of *IDataSet*), which contains various elements related to analytical objects or helpers (e.g. notes, legend, scale). Each analytical element contains the following three aspects:

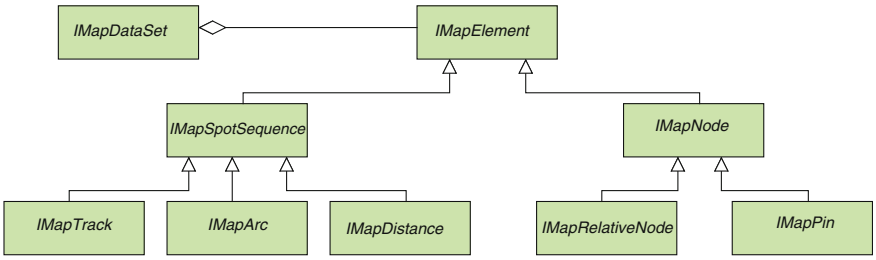


Fig. 3 Model hierarchy

- *common data* related to geospatial and time perspectives—each element has coordinates and optionally collection of date and time values which is designed to represent events;
- *domain-specific data*—each element has a custom set of attributes related to its domain type (e.g. a person, car, place, POI);
- *visual data* related to a way of visualizing an element—depending on visual type (pin, track, note), each element has a set of presentation attributes that describe a way of showing on a map (e.g. color, icon, size).

4.1 Object Model of Map Data Set

Map elements are described by object model shown in Fig. 3. They are arranged in hierarchy that can be easily extended by adding new types on suitable branch and level.

The following types of elements are defined:

- Nodes (*IMapNode*) locations on map that are defined by a pairs of coordinates (latitude and longitude), hold icon and are described by a label.
  - Pins (*IMapPin*)—map points represented as pin with icon.
  - Balloons (*IMapBalloonSlaveNode*)—rich visual tooltips, connected to map nodes and describing their content.
  - Notes (*IMapNote*)—visual annotations without geographical location.
- Spot Sequences (*IMapSpotSequence*)—various elements on map that are defined by a sequence of spots (locations on map)
  - Tracks (*IMapTrack*)—trace of some movement (e.g. obtained from a GPS tracking device), defined by a sequence of spots that can have a date associated to each of them.
  - Distances (*IMapDistance*)—distance measurements shown on map. It is a sequence of spots that is used for calculating distance.

– Directed Arcs (*IMapDirectedArc*—arcs that are defined by: spot (location on map), span angle (arc's central angle), distance (arc's radius length) and direction (an azimuth on map).

4.2 Operators for Geospatial Analysis

LINK Map provides also a set of analytical tools which supports analysts in typical steps taken during investigations that involve geospatial analysis. Two groups of operators can be distinguished:

- general—only geographical and (optionally) date and time context is utilized, e.g. finding correlation between localization of objects ("meeting-search" operator),
- domain-specific—domain-related attributes are used to perform specific operations, e.g. showing BTS (Base Transceiver Station) range based on its attributes such as direction, range and angle.

Furthermore, LINK Map supports conversions of map data sets to external formats such as JSON or *shapefile*, which can be processed in other geospatial analysis tools such as ArcGIS or Google Earth.

The list of analytical operators can be easily extended by providing custom components by third-parties.

5 A Case Study for Geo-Spatial Analysis of Various Data

In order to verify the approach described in this paper, we present a sample case, which illustrates how the LINK Map tool-set can be used in criminal analysis. The goal of the sample analysis is to verify hypothesis stating that two suspects have something in common with a committed crime. As an input data we have one phone call billing, data from GPS tracker attached to a car of the second suspect and crime location.

1. The first step will be marking the crime scene on the map.
2. Next, the analyst imports phone billings obtained for the suspects. Based on the phone call records, it occurs that one of the suspect was in the area when crime was committed. While the BTS stations cannot show the direction that the suspect was heading towards, by applying proper operator it is possible to calculate (based on the time of the calls) the order in which the number logged onto the displayed BTS.
3. As for the second suspect, let's assume that police installed a GPS tracking device which, when imported onto map, shows that he was heading north, nearby the crime zone.

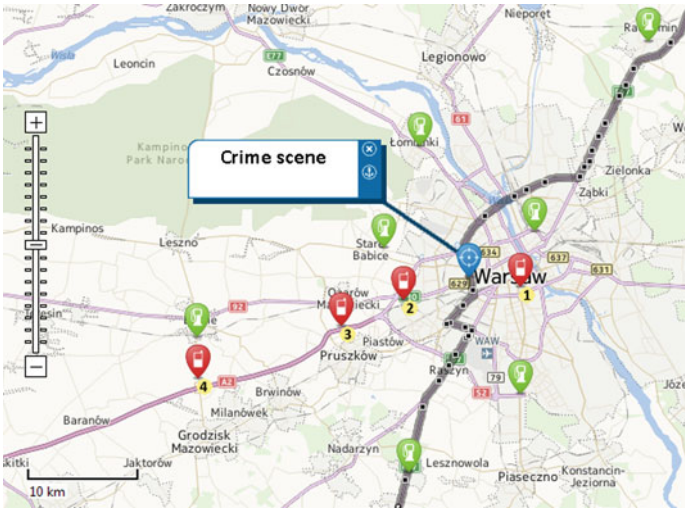


Fig. 4 Data integration for geo-spatial analysis using Nokia maps

4. In addition, for further analysis, it possible to add to the map other points, such as gas stations, in order to verify the statements of the suspects concerning the evening of the crime.

The final map diagram created in the above scenario is shown in Fig. 4. This scenario shows the capabilities of the presented tool-set which allows for integrating data from multiple domains on the geographical map in order to perform spatial-oriented analysis.

However, if one has large volume of data, visualization is not enough to perform effective analysis. Than it's time for using built-in analysis operators which can point interesting places and facts based on some hypothesis. The current prototype delivers operations listed below, but also custom analytical operators can be developed.

- finding potential meeting points between one or more objects based on given time window and allowable distance;
- searching for POI points nearby location of chosen object;
- finding objects that possibly moved together;
- setting centroids—locations that point central point of object activity;

6 Conclusion and Further Research

In the paper a concept of flexible framework for geospatial analysis is presented. The main idea behind the concept is to provide users with extensible data model and operators layer which provides flexibility in defining new analytical functions and

handling new domain data. A prototype of the framework is built based on the LINK analytical platform and its evaluation on a sample case indicates positive results.

The presented concept can be further extended towards the realization of analytical functions and descriptive data models. In the first area the following GIS analyses are considered:

- hot spot analysis [11]—calculates the gradient of investigated phenomenons occurrence; the method investigates how often certain phenomenons occur in the area of interest and creates so called hot spot maps, which are important or when investigating relations between crimes in the certain neighborhood or analyzing suspect's appearance patterns;
- driving ranges analysis—creates zones, which could have been reached by a subject from a certain point using road network and time constraints; it might be very useful during a pursuit, in order to decrease the search range and to make it more accurate; the method is based on the Dijkstra's algorithm [3], which in this case traverses the road network and returns a subset of connected edges within the specified time constraints;
- driving belt analysis—is an extension of the previous analysis and has been designed as one of the novel GIS algorithms by authors of the LINK project (in a shape of prototype); it creates a driving belt consisting of areas and roads, which might have been used be a subject during a tracked drive.

Also, the existing models can be extended toward descriptive data model where data description is provided in a dedicated, expressive DSL (Domain-Specific Language)—then models and functions will be more readable, easily modifiable and maintainable. An interesting research can be also conducted in terms of web-oriented analysis where users cooperate based on the selected domain data and operators. In the current version, geographical data can be exported to *shapefile* format and than used in tools such as ArcGIS or Google Earth. Further work will focus on tighter integration to allow also bi-directional work with these tools.

Acknowledgments The research reported in the paper was partially supported by grants "Advanced IT techniques supporting data processing in criminal analysis" (No. 0008/R/ID1/2011/01) and "Information management and decision support system for the Government Protection Bureau" (No. DOBR-BIO4/060/13423/2013) from the Polish National Centre for Research and Development.

References

1. Dębski, R., Kisiel-Dorohinicki, M., Młoś, T., Piętak, K.: Link: a decision-support system for criminal analysis. In: MCSS 2010. pp. 110–116. Krakow, Poland (2010)
2. Dajda, J., Debski, R., Kisiel-Dorohinicki, M., Pietak, K.: Multi-domain data integration for criminal intelligence. In: Gruca, A., Czachórski, T., Kozielski, S. (eds.) Man-Machine Interactions 3. AISC, vol. 242, pp. 345–352. Springer, Switzerland (2013)
3. Dijkstra, E.: A note on two problems in connexion with graphs. Numer. Math. **1**(1), 269–271 (1959)
4. Gorr, W.L., Kurland, K.S.: GIS tutorial for crime analysis. Esri Press, New York (2011)

5. Haining, R.: Spatial data analysis: theory and practice. Cambridge University Press, Cambridge (2003)
6. IBM: IBM i2 Analyst's Notebook website, www.ibm.com/software/products/en/analysts-notebook
7. Payne, J., Solomon, J., Sankar, R., McGrew, B.: Palantir: The future of analysis. In: VAST 2008, pp. 201–202. Columbus, USA (2008)
8. Rutkin, A.: Crime suspects in sicily traced by cellphone metadata. New Sci. **222**(2966), 20 (2014)
9. Santos, R.: Crime modeling and mapping using geospatial technologies. SAGE Publications, Los Angeles (2012)
10. Saravanan, M., Thayyil, R., Narayanan, S.: Enabling real time crime intelligence using mobile gis and prediction methods. In: EISIC 2013, pp. 125–128. Uppsala, Sweden (2013)
11. Silverman, B.W.: Density estimation for statistics and data analysis. Chapman and Hall, New York (1986)
12. Smith, M., Longley, P., Goodchild, M.: Geospatial analysis: a comprehensive guide to principles. techniques and software tools. The Winchelsea Press, Leicester (2013)
13. Wang, X., Lin Liu, J.E.: Crime simulation using gis and artificial intelligent agents. In: Lin Liu, J.E. (ed.) Artificial Crime Analysis Systems, pp. 209–225. IGI GLOBAL, London (2008)

Multivariate Approach to Modularization of the Rule Knowledge Bases

Roman Simiński

Abstract This article introduces the multivariate approach to modularization of the rule knowledge bases. The main difference between proposed approach and other known modularization methods consists in the fact that proposed idea allow us to use different modularization strategies, according to the current requirements specified by the expert or knowledge engineer. This work describes the formal modularization model, an example of rule base modularization variants and short comparison with existing modularization methods.

Keywords Knowledge base · Modularization · Rules groups

1 Introduction

The rules systems are the well known solvers for specialized domains of competence, in which effective problem solving normally requires human expertise. The rules are still the most popular, important and useful tool for constructing knowledge bases. The rule bases are constantly increasing in volume, thus the knowledge stored as a set of rules is getting progressively more complex and much harder to interpret, analyze, and the large size of the rules sets has got an enormous negative impact on the time efficiency of inference. Relations and connections between literals within a single rule are clear, whereas, when the number of rules increases, the dependence between rules is less clear and its reconstruction requires lots of work and attention from a knowledge engineer.

The modularization of the rule knowledge bases that introduces structure to the possibly large knowledge base can be considered as the way to avoid maintenance problems and cause inefficiency of inference. The main goal of our works is to present the modularization approach which assumes that the rule base is decomposed into the groups of rules. The main focus of this paper is the proposed idea of modularization approach rather than the experimental results, this is due to the fact that introduced

R. Simiński (✉)
Institute of Computer Science, University of Silesia, Katowice, Poland
e-mail: roman.siminski@us.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_40

modularization approach is relatively new and not well known. The main difference between proposed approach and other known modularization approaches consists in the fact that proposed idea allows us to use different modularization strategies, according to the current requirements specified by the expert or knowledge engineer. We propose a single, coherent modularization method, which is able to generate different modular models for the rule knowledge base. Proposed approach is in this a way multivariate method, obtained modular models of rule knowledge base are not limited to the one particular decomposition method. This work describes the formal modularization model, an example of modularization variants and a short comparison with existing modularization methods.

The proposed solution represents only a selected portion of a larger project. The practical goal of the project is to create a Web based [15] expert system shell with extension in the form of desktop application. Multivariate modularization method is used in such system as a tool for optimization of classical the inference methods efficiency [11, 14], creation of the new inference methods (e.g. for incomplete data) [6, 10] an other knowledge engineering tasks.

2 Related Works

The first well known attempt to define the knowledge base decomposition is the blackboard architecture. The blackboard system consists of knowledge modules which are independent modules that contain the knowledge needed to solve the problem. Each knowledge module specializes in the solving certain aspects of the overall problem [3]. The knowledge about the problem is split into a number of small knowledge bases called knowledge sources, and controlled through the blackboard control mechanism.

In this way we can consider a blackboard architecture as an inspiration for knowledge bases modularization [4]. Some expert system shells provide blackboard architecture inspired knowledge sources. `PC-Shell` expert system shell can be an example of such system [16]. Whole rule knowledge base can be divided into groups of rules, arbitrarily established by the knowledge engineer. The system does not provide any consistent method of rule base decomposition. However, system `CAKE` can be used as a tool for building multi-source knowledge bases. Decomposition methods of knowledge bases can be found in other well known systems, like `Drools`, `JESS`, `CLIPS` or `XTT2`. `CLIPS` system can divide rules into modules that allow controlling access to rules from other modules. Modularization of knowledge base helps with managing the rules, and improves efficiency of inference. Each module has its own pattern-matching network for its rules and its own agenda. Only the module that has focus set is processed by the inference engine. `CLIPS` modules do not provide any method determining which rule can be placed in a module, any rule in particular can be placed in any module [5, 8].

`Drools` system allows us to define the structure of the rule knowledge base. The rules can be grouped in a ruleflow groups, which have a graphical representation

as the nodes on the ruleflow diagram. Ruleflow allows us to specify the order in which rule sets should be evaluated by using a flow chart. This allows the definition which rule sets should be evaluated in sequence or in parallel, to specify conditions under which rule sets should be evaluated. A rule flow is a graphical description of a sequence of steps that the rule engine needs to take, where the order is important. Ruleflows can only be created by using the graphical ruleflow editor, which is part of the `Drools` plugin for Eclipse. However, there is also no consistent method of rule base decomposition [2].

`XTT2` system also provides modularized rule bases. Contrary to the majority of other systems, where a basic knowledge item is a single rule, in the `XTT2` formalism the basic component, displayed, edited and managed at a time, is a single context. A single context corresponds to a single decision table. Thus, only those rules which have the same conditional and decisions attributes can be placed in one context i.e. each rule in a decision table determines values of the same set of attributes [8].

Several efforts have been deployed to tackle the problem of the huge number of association rules. The number of rules has to be reduced significantly by techniques such as pruning or grouping. A clustering algorithms are used to build related groups of the rules [17], in [1] the information provided by the metarules is used to reorganize and group related rules, the author in [7] presents a solution for reducing the number of rules by joining them from some clusters. In the rule knowledge bases we can observe the tendencies to utilize the concepts from decision tables [19] and data bases theory, like distribution of the data bases for example and other methods described in papers [12, 18] and also approach described in [13].

The methods of modularization briefly described in this chapter are dedicated for specific systems and specific tasks. In many cases, the methods above are dependent on the specific tools and they cannot directly be applied in other systems and for other tasks different than established by the authors. The specialization and limitation of existing modularization methods is the motivation for this work.

3 Methods

Multivariate modularization method is based on the proposed method of rule knowledge base partitioning. A significant part of this work contains a detailed description of proposed approach. Introduced approach differs from other methods of the modularization and actually is described in a small number of publications.

3.1 The Knowledge Base and Rules Partitions

The knowledge base is a pair $\mathcal{KB} = (\mathcal{R}, \mathcal{F})$ where $\mathcal{R}$ is a non-empty finite set of rules and $\mathcal{F}$ is a finite set of facts. $\mathcal{R} = \{r_1, r_2, \ldots, r_i, \ldots, r_n\}$, each rule $r \in \mathcal{R}$ will have a form of Horn's clause: $r : p_1 \wedge p_2 \wedge \cdots \wedge p_m \rightarrow c$, where m—the number of literals

in the conditional part of rule r, and $m \geq 0$, p_i—ith literal in the conditional part of rule r, $i = 1 \ldots m$, c—literal of the decisional part of rule r. For each rule $r \in \mathcal{R}$ we define following the functions: $concl(r)$—the value of this function is the conclusion literal of rule r: $concl(r) = c$; $cond(r)$—the value of this function is the set of conditional literals of rule r: $cond(r) = \{p_1, p_2, \ldots p_m\}$, $literals(r)$—the value of this function is the set of all literals of rule r: $literals(r) = cond(r) \cup \{concl(r)\}$, $csizeof(r)$—conditional size of rule r, equal to the number of conditional literals of rule r ($csizeof(r) = m$): $csizeof(r) = |cond(r)|$, $sizeof(r)$—whole size of rule r, equal to the number of conditional literals of rule r increased by the 1 for single conclusion literal, for rules in the form of Horn's clause: $sizeof(r) = csizeof(r)+1$. We will also consider the *facts* as clauses without any conditional literals. The set of all such clauses f will be called *set of facts* and will be denoted by $\mathcal{F}$: $\mathcal{F} = \{f : \forall_{f \in \mathcal{F}}\, cond(f) = \{\} \wedge f = concl(f)\}$.

For each rule set $\mathcal{R}$ with n rules, there is a finite power set $2^{\mathcal{R}}$ with cardinality 2^n. Any arbitrarily created subset of rules $R \in 2^{\mathcal{R}}$ will be called *a group of rules*. In this work we will discuss specific subset $PR \subseteq 2^{\mathcal{R}}$ called *partition of rules*. Any partition PR is created by *partitioning strategy*, denoted by PS, which defines specific content of groups of rules $R \in 2^{\mathcal{R}}$ creating a specific *partition of rules* PR. We may consider many partitioning strategies for a single rule base, in this work we will only present a few selected strategies. Each partitioning strategy PS for rules set $\mathcal{R}$ generates the partition of rules $PR \subseteq 2^{\mathcal{R}}$: $PR = \{R_1, R_2, \ldots, R_k\}$, where: k—the number of groups of rules creating the partition PR, R_i—ith group of rules, $R \in 2^{\mathcal{R}}$ and $i = 1, \ldots, k$.

Rules partitions terminologically correspond to the mathematical definition of the partition as a division of a given set into the non-overlapping and non-empty subset. The groups of rules which create partition are pairwise disjoint and utilize all rules from $\mathcal{R}$. The partition strategies for rule based knowledge bases are divided into two categories:

- *Simple strategies*—the *membership criterion* decides about the membership of rule r in a particular group $R \subseteq PR$ according to the membership function mc. Simple strategy let us divide the rules by using the algorithm with time complexity not higher than $O(n \cdot k)$, where $n = |\mathcal{R}|$ and $k = |PR|$. Simple strategy creates final partition PR by a single search of the rules set $\mathcal{R}$ and allocation of each rule r to the proper group R, according to the value of the function $mc(r, R)$ described bellow.
- *Complex strategies*—the particular algorithm decides about the membership of the rule r in some group $R \subseteq PR$, with time complexity typically higher than any simple partition strategy. Complex strategies usually do not generate final partition in a single step. Complex partitioning strategies will not be discussed in this work. An example of a complex strategy is described in the [9, 11].

3.2 Simple Partitioning Strategy and Properties of the Partitions

The creation of *simple partition* requires the definition of the *membership criteria* which assigns particular rule $r \in \mathcal{R}$ to the given group of rules $R \subseteq PR$. Proposed approach assumes that the membership criteria will be defined by the mc function, which is defined individually for every simple partition strategy. The function: $mc : \mathcal{R} \times PR \rightarrow [0..1]$, return the value 1 if the rule $r \in \mathcal{R}$ with no doubt belongs to the group $R \subseteq PR$, 0 in the opposite case. The value of the function from the range $0 < mc < 1$ means the partial membership of the rule r to the group R. The method of *determining* its value and its *interpretation* depends on the specification of a given partition method. It is possible to achieve many different partitions of rule base using single mc function.

A specific partition strategy called *selection* is a special case of the simple strategies. The selection divides the set of rules $\mathcal{R}$ into the two subsets R and $\mathcal{R} - R$. All rules from R fulfill the membership criteria for some partition strategy PS, and all other rules do not meet such criteria. Thus, we achieve the partition $PR = \{R, \mathcal{R} - R\}$. In a practical sense, selection is the operation with linear time complexity $O(n)$ where n denotes the number of all rules in knowledge base. For each group of rules $R \in PR$ we can define: (i) $Cond(R)$—the *set of conditions* for the group of rule R, containing literals l appearing in the conditional parts of the rules r from R: $Cond(R) = \{l : \exists_{r \in R}\ l \in cond(r)\}$, (ii) $Concl(R)$—the *set of conclusions* for the group of rule R, containing literals l appearing in the conclusion parts of the rules r from R: $Concl(R) = \{l : \exists_{r \in R}\ l = concl(r)\}$, (iii) $L(R)$—*set of literals* for group of rule R: $L(R) = Cond(R) \cup Concl(R)$.

3.3 The Simple Partitioning and Selection Algorithms

The *simple strategy* partitioning algorithm is presented in the pseudo-code below. The input parameters are: the knowledge base $\mathcal{R}$, the function mc that defines the membership criteria, and the value of the threshold T. Output data is the partition PR. Time complexity of such algorithm is $O(n \cdot k)$, where $n = |\mathcal{R}|$, $k = |PR|$. For each rule $r \in \mathcal{R}$ we have to check whether the goal partition PR contains the group R with rule r (the value of the mc function has to be at least T: $mc(r, R) \geqslant T$). The *selection* algorithm is the special case of a simple partitioning algorithm, it is very simple and for this reason will be omitted.

Algorithm 1 The simple partition algorithm

Require: $\mathcal{R}, mc, T$;
Ensure: $PR = \{R_1, R_2, \ldots, R_k\}$;
  procedure ***createPartitions***($\mathcal{R}$, **var** PR, mc, T
  var R, r;
  begin
  $PR = \{\}$;
  for all $r \in \mathcal{R}$ **do**
    if $\exists R \in PR : mc(r, R) \geqslant T$ **then**
      $R = R \cup r$;
    else
      $R = \{r\}$;
      $PR = PR \cup R$;
    end if
  end for
  end procedure

4 A Simple Case Study and Comparison with Modularization Alternatives

To illustrate the conception of multivariate knowledge base modularization, we consider an example knowledge base $\mathcal{R}$. In this work we assume that rule's literals will be denoted as the pairs of attributes and their values. Due to the limited size of this work, formal description of attributes and value sets is omitted. An example of rules set is presented in the Fig. 1.

Basic decision oriented modularization. Let us discuss the following partitioning strategy PS_1 which creates groups of the rules from $\mathcal{R}$, by grouping rules with the same conclusions. The membership criteria for rule r and group R is given by the function mc defined as follows:

$$mc(r, R) = \begin{cases} 1 & \text{if } \forall_{r_i \in R}\ concl(r_i) = concl(r) \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

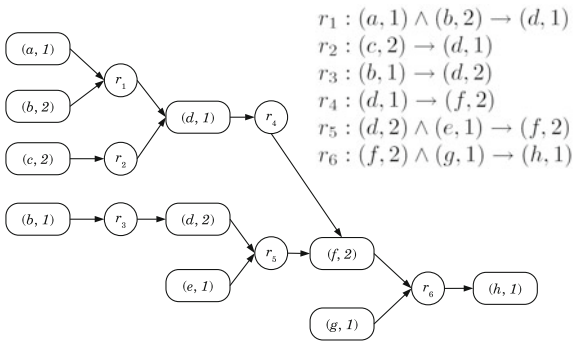


Fig. 1 An example rules as a rule-attribute diagram

By using the *simple partition algorithm* (Alg01) with the *mc* function defined in this way, we obtain *basic decision oriented partitions*. Each group of the rules generated by this algorithm may have the following form: $R = \{r \in \mathcal{R} : \forall_{r_i \in R}\, concl(r_i) = concl(r)\}$. For the previous example the created partition PR_1 has the following form: $PR_1 = \{\{r_1, r_2\}, \{r_3\}, \{r_4, r_5\}, \{r_6\}\}$. The number of groups in the partition depends on the number of *different decisions* included in conclusions of such rules. When we distinguish different decision by the different conclusions appearing in the rules, we get one group for each conclusion. All rules grouped within a rule set take part in an inference process confirming the goal described by the particular attribute-value — for each $R \in PR_1$ the conclusion set $|Concl(R)| = 1$. Ordinal decision oriented modularization strategy is used in the modified backward inference algorithm and described in [11]. The proposed modification consists of the reduction of the search space by choosing only the rules from particular rule group, according to a current structure of decision oriented rules partition and the estimation of the usefulness for each recursive call for sub-goals.

Ordinal decision oriented modularization. Let us consider the second partitioning strategy PS_2, the membership criterion for rule r and group R given by the function *mc* defined as follows:

$$mc(r, R) = \begin{cases} 1 & \text{if } \forall_{r_i \in R}\, attrib(concl(r_i)) = attrib(concl(r)), \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

When we utilize the *simple partition algorithm* (Alg01) with the *mc* function defined in such way, we obtain *ordinal decision oriented partitions*. Each group of the rules generated by this algorithm may have the following form: $R = \{r \in \mathcal{R} : \forall_{r_i \in R}\, attrib(concl(r_i)) = attrib(concl(r))\}$. For the given example the created partition PR_2 has the following form: $PR_2 = \{\{r_1, r_2, r_3\}, \{r_4, r_5\}, \{r_6\}\}$. The number of groups in the partition depends on the number of *different decisions attributes* included in conclusions. Any rule set $R \in PR_2$ can be described as $R = \{\bigcup R' \in PR_1 : \forall r_i, r_j \in R'\ attrib(concl(r_i)) = attrib(concl(r_j))\}$, it means that that decision produced by the decision partition can be constructed as the composition of the basic decision partitions.

Ordinal decision oriented modularization strategy is also used in the modified backward inference algorithm [11] and as a tool for retrieving decision model hidden in the large rules sets. Currently, both decision modularization methods are used for analysing and redesigning two real-word rule bases, counting respectively over 4000 rules over 1000. The first base, designed for WWW online expert system [6, 15], originally was created by the five experts without any verification tool. Second base was created for the mobile expert system dedicated to assessment of the performance of the merchants. Although discussed method is simple and even seems trivial, the practical usefulness and effectiveness turned out to be significant in the verification and validation of considered rule bases. Between groups of rules within selected partition we can point out possible connections. Those connections will be crucial in the many knowledge engineering task for complex rule bases. Discussed

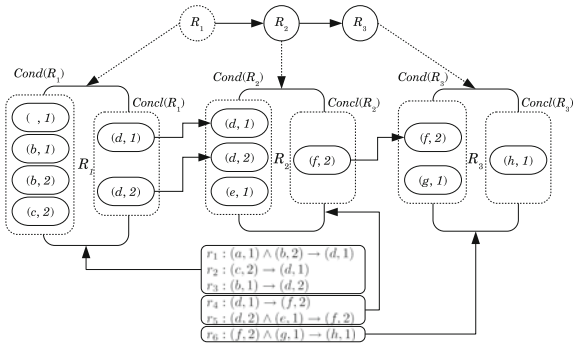


Fig. 2 Connection graph as multigraph

connection graph is in general independent of the chosen modularization variant, but it is especially useful when we consider earlier described decision oriented partitions.

To express and utilize information about global and local decision dependencies, we define *partitions connection graph* G_{PR}. For each rules set R of the partition we can examine the connection with other groups and the detailed information between literals participating in such connections. Graph G_{PR} presented in this work is a simple directed graph. We also consider multigraphs, an example of a representation of connection graph as the multigraph is presented on the bottom part of the Fig. 2.

Fact matching modularization or selection. For the third partitioning strategy PS_3, the membership criterion is defined as follows: $mc(r, R) = \frac{|cond(r) \cap \mathcal{F}|}{|cond(r)|}$. When we utilize the *simple partition algorithm* or *selection algorithm* with the mc function defined in such way, we obtain *fact matching partitions* or *fact matching selection*. When we consider threshold equal to 1, for a given set of facts: $\mathcal{F} = \{(a, 1), (b, 2), (c, 2)\}$, the strategy generates the two groups. In the first of them we obtain rules ready to fire, in the second one—rule not matching to the fact set. Each group of the rules generated by this algorithm may have the following form: $R = \{r \in \mathcal{R} : cond(r) \subseteq \mathcal{F}\}$. For the considered example the created partition PR_3 has the following form: $PR_3 = \{\{r_1, r_2\}, \{r_3, r_4, r_5, r_6\}\}$. Fact matching modularization/selection strategy is used in the modified forward inference algorithm described in [11]. We can consider: $\mathcal{F} = \{(a, 1), (d, 2)\}$ and threshold equal to 1, there are no fireable rules, but for the threshold equal to 0.5 we obtain $PR_3 = \{\{r_1, r_5\}, \{r_2, r_3, r_4, r_6\}\}$. The first subgroup contains rules half matched to the facts set. This modularization/selection strategy is used in the works considering the continuation of inference in incomplete data and knowledge described in [10].

The basic and ordinal decision partitions are similar to the association rules grouping approaches described in the works [1, 17]. The authors apply clustering algorithms for rules joining and pruning. We also propose a complex modularization strategy including the clustering strategy [9, 11], originally focused on the forward inference optimization, currently also used as a tool for building metarules. The modularization approaches proposed by the `Drools`, `CLIPS` and `PC-Shell` systems are difficult to compare with the approach proposed in this work. These systems do

not provide any method determining which rule can be placed in a module, any rule in particular can be placed in any rule subgroup, modularization is realized manually by the user or knowledge engineer and the resulting rules structure is used internally, typically for inference optimization and control. The modularization variants proposed in this work are also used in the inference optimization tasks [11, 14] and, in this context, offer similar functionality. Ordinal decision partition partially corresponds to the context of a single `XTT2` table, which contains rules with the same attributes in their conditional and conclusion parts. It is possible to do define complex partitioning strategy which joins the decision strategy and similarity strategy. This strategy is determined by using the function sim [11] based on the similarity of conditional part of rules. It allow us to cover the XTT2 modularization strategy.

The main difference between proposed approach and other known modularization approaches consists in the fact that proposed idea allows the application of different modularization strategies using single, coherent modularization method. This method is able to generate different modular models for the rule knowledge base automatically, according to the requirements specified by the expert or knowledge engineer. Proposed approach is in this way a multivariate method, obtained modular models of rule knowledge base are not limited to the one particular decomposition method, system, specific knowledge-based system's activity. We only assume that proposed approach works on any typical rule base, which utilize the horn's clause like notation and literal as an attribute-value pair, but in general, considered approach is open for other literal's formal representation. The approach proposed in this work does not assume the use of any specific tools and development methodology. The limited size of this publication does not allow us to present others variants of the modularization, the variants discussed in this work are simple and may seem trivial. However, even these simple variants are useful when we consider large rule sets without any initial structure. Proposed multivariate modularization method is used in the software project `kbExplorer`. `kbExplorer` is a Web-based expert system building tool, which provides different rules modularization methods, classical and modified inference methods. Web application is implemented using PHP/MYSQL/JS/HTML5. We also work on the desktop system implemented in the Java. Systems will be available as free software this autumn.

5 Conclusions

This article introduce the multivariate approach to modularization of the rule knowledge bases. This work describes the formal modularization model, an example of modularization variants. A general, flexible and tool independent rule modularization method was considered. Rules can be grouped in rules subsets and proposed approach reorganizes any typical rule knowledge base from a set of not related rules to groups of rules called *partitions*. It is possible to group rules using different simple partitioning strategies, using single algorithm with $O(n)$ complexity as well as complex partitioning strategies. Proposed approach is in this way multivariate method,

obtained modular models of rule knowledge base are not limited to the one particular decomposition method and system. We also assume that proposed approach works on any typical rule base, which utilize the horn's clause like notation and literal as an attribute-value pair, but in general, considered approach are open for other literal's formal representation.

Acknowledgments This work is a part of the project "Exploration of rule knowledge bases" founded by the Polish National Science Centre (NCN: 2011/03/D/ST6/03027).

References

1. Berrado, A., Runger, G.C.: Using metarules to organize and group discovered association rules. Data Min. Knowl. Disc. **14**(3), 409–431 (2007)
2. Browne, P.: JBoss drools business rules. Packt Publishing Ltd, Birmingham (2009)
3. Carver, N.: A revisionist view of blackboard systems. In: MAICS 1997. Dayton, USA (1997)
4. Corkill, D.D.: Blackboard systems. AI Expert **6**(9), 40–47 (1991)
5. Giarratano, J.C., Riley, G.: Expert systems. PWS Publishing Co., Boston (1998)
6. Jach, T., Xieski, T.: Inference in expert systems using natural language processing. In: Kozielski, S., Mrozek, D., Kasprowski, P., Małysiak-Mrozek, B., Kostrzewa, D. (eds.) Beyond Databases, Architectures and Structures, Communications in Computer and Information Science, vol. 521, pp. 288–298. Springer, Switzerland (2015)
7. Mikołajczyk, M.: Reducing number of decision rules by joining. In: Alpigini, J.J., Peters, J.F., Skowron, A., Zhong, N. (eds.) Rough Sets and Current Trends in Computing, LNCS, vol. 2475, pp. 425–432. Springer, Berlin Heidelberg, Germany (2002)
8. Nalepa, G., Ligeza, A., Kaczor, K.: Overview of knowledge formalization with XTT2 rules. In: Bassiliades, N., Governatori, G., Paschke, A. (eds.) Rule-Based Reasoning, Programming, and Applications, LNCS Volume, vol. 6826, pp. 329–336. Springer, Berlin Heidelberg, Germany (2011)
9. Nowak, A., Wakulicz-Deja, A.: The way of rules representation in composited knowledge bases. In: Cyran, K.A., Kozielski, S., Peters, J.F., Stanczyk, U., Wakulicz-Deja, A. (eds.) Man-Machine Interactions, Advances in Intelligent and Soft Computing, vol. 59, pp. 175–182. Springer, Berlin Heidelberg, Germany (2009)
10. Nowak-Brzezińska, A., Jach, T., Wakulicz-Deja, A.: Inference processes using incomplete knowledge in decision support systems chosen aspects. In: Yao, J., Yang, Y., Sowiski, R., Greco, S., Li, H., Mitra, S., Polkowski, L. (eds.) Rough Sets and Current Trends in Computing, LNCS, vol. 7413, pp. 150–155. Springer, Berlin Heidelberg, Germany (2012)
11. Nowak-Brzezińska, A., Simiński, R.: New inference algorithms based on rules partition. In: CS&P 2014. pp. 164–175. Chemnitz, Germany (2014)
12. Pedrycz, W.: Knowledge-based clustering: from data to information granules. Wiley, Hoboken (2005)
13. Sikora, M., Gudyś, A.: Chira–convex hull based iterative algorithm of rules aggregation. Fundamenta Informaticae **123**(2), 143–170 (2013)
14. Simiński, R.: Extraction of Rules Dependencies for Optimization of backward inference algorithm. In: Kozielski, S., Mrozek, D., Kasprowski, P., Małysiak-Mrozek, B., Kostrzewa, D. (eds.) Beyond Databases, Architectures, and Structures, Communications in Computer and Information Science, vol. 424, pp. 191–200. Springer, Switzerland (2014)
15. Simiński, R., Manaj, M.: Implementation of expert subsystem in the web application-selected practical issues. Studia Informatica **36**(1), 131–143 (2015)
16. Simiński, R., Michalik, K.: The hybrid architecture of the ai software package sphinx. In: CAI 1998, Colloquia in Artificial Intelligence. Poland (1998)

17. Strehl, A., Gupta, G.K., Ghosh, J.: Distance based clustering of association rules. In: ANNIE 1999. vol. 9, pp. 759–764. St. Louis, USA (1999)
18. Toivonen, H., Klemettinen, M., Ronkainen, P., Hätönen, K., Mannila, H.: Pruning and grouping discovered association rules (1995)
19. Wakulicz-Deja, A., Nowak-Brzezińska, A., Przybyła-Kasperek, M.: Complex decision systems and conflicts analysis problem. Fundamenta Informaticae **127**(1–4), 341–356 (2013)

Part VIII
Pattern Recognition

Practical Verification of Radio Communication Parameters for Object Localization Module

Karol Budniak, Krzysztof Tokarz and Damian Grzechca

Abstract The paper presents parameters verification of a mobile device (battery supplied) for object localization built with GPS module and radio front-end. At first, construction of PC device (stationary unit) and mobile device (unit) are described. Mobile unit may be attached to a movable object. More, objects in-motion requires short round-trip-time (mobile response on stationary request) and communication distance must be long enough for communication. In the paper the most important datasheet parameters have been compared with devices constructed and the number of test scenarios have been performed in order to verify parameters in real environment i.e. mobile unit lifetime on battery, communication range, output power, positioning accuracy, etc.

Keywords Radio communication · Objects localization · RTLS

1 Introduction

Real time locating systems (RTLS) became more popular in recent years. It is the result of still growing utilization of mobile devices with variety of possible applications. The radio communication modules are available at low price and allow for sending/receiving information from moving objects to the host. Current technology allows for creating ICs (Integrated Circuits) of parameters that were unable a few years ago. Thus, RF (Radio Frequency) modules became alternative for cable communication. A device with independent communication link for tracking is required

K. Budniak · D. Grzechca
Institute of Electronics, Silesian University of Technology, Gliwice, Poland
e-mail: budniak.karol@gmail.com

D. Grzechca
e-mail: damian.grzechca@polsl.pl

K. Tokarz (✉)
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: krzysztof.tokarz@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_41

if taking into account permanent protection of security guard or other stuff in open-air large area. Such system requires specific approach. The accuracy must be on acceptable level very often 10 m is enough. The operation time of the system powered with battery should be at least 12 h. For independent radio communication a new infrastructure is required so the system must be scalable but not expensive. Long distance link with high communication speed reduces cost (stationary unit should cover at least circle with radius of 1km). Fast development of objects localization systems allows to apply many ideas and techniques. Directivity of RTLS development has been presented in [1] mainly concentrate on medical healthcare of patients. In [11] localization of objects has been performed by sending SMS with actual position of the object obtained from GPS module. Accuracy in this method mainly depends on amount of sending messages. Positioning with RTLS system, presented in [6], yields only in populated areas, with prevalence of WiFi networks. Sending information about actual object position via dedicated radio channel ensures high accuracy and allows using system in rural environments. System behavior on the higher level (PC application) uses UTC time and the GPS coordinates, sent via RF module to the stationary unit, therefore reliable work of whole system depends on quality of GPS signal and retention of RF communication. Proposed device for objects positioning is composed of RF transceiver module and GPS module. RF part is build with CC1101 transceiver with CC1190EM amplifier both working at 869 MHz (Texas Instruments and Chipcon ICs). GPS unit used in the device is FGPMMOPA4.

According to data provided by ChipCon, RF module offers adjustment of the following parameters: output power, channel settings: bandwidth, spacing, data rate. Connecting power amplifier CC1190 to transceiver CC1101 should increase communication range and quality of signal received. In order to fulfill system requirements it is necessary to verify communication parameters, RF signal path loss and GPS coordinates dispersion. RF signal path loss is compared with environment dependent model presented in [7], by analysis RSSI parameter (Receive Signal Strength Index) for adequate output power implementation. Test conditions for model of propagation has been performed in suburban area, line of sight has been preserved. Test for GPS coordinates dispersion has been performed, during static position and in motion with different speed. Important part of the verification is battery lifetime with respect to output power of RF module. It should be noticed that communication algorithm plays also important role in battery lifetime [2].

2 System Construction

System consists of two devices: mobile and stationary. Stationary device initiates communication by sending request to mobile device to get its position. Then, mobile device responds with its unique identifier and actual GPS coordinates (taken from GPS module). In case of no answer received from mobile station, the communication procedure is repeated. Mobile device is supplied by Li-Ion batteries, which can be charged via USB port. Such solution requires: USB charge management

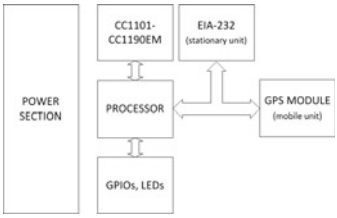


Fig. 1 Block diagram of mobile device

controller (MCP73831), an analog comparator for monitoring battery voltage (protection against damage if fully empty ADCMP361) and high efficiency buck-boost converter (for low ripple on power line). As a GPS module the FGPMMOPA4 [3] has been used. Whole system is controlled by 8-bit processor ATmega32. Block diagram of a single device has been shown in Fig. 1.

Low level firmware has been developed with the use of Atmel Studio software and operates in the following manner:

1. Configuration settings for: GPIOs, GPS module in mobile unit (UART/9600bps/8/1/even), EIA-232 in stationary unit (19200bps/8/1/none) and RF module (SPI interface). The CC1101 (RF module) and CC1190 (low noise power amplifier) initialization covers specific values to IC registers: radiated power [10] (i.e. 20 and 15 dBm, both have been verified), channel filter bandwidth (58 kHz) and spacing (200 kHz). Those parameters have been verified assuming modulation type (GFSK), deviation (14 kHz) and baud rate (1.2 kbps). Configuration settings for RF module have been created with SmartRF Studio (Texas Instruments environment) which also allows for exporting them directly to the header file. A mobile device must have a unique identifier. That ID has been read from hardware directly from input port with internal pull up (address configurable by pulling low appropriate bits).
2. Main loop. Stationary device sends 5 bytes asking frame as in Fig. 2. In order to preserve energy of the battery in mobile device after occurrence of asking frame, device turns RF module into power save state, then comparison of ID from frame with ID of device follows. When IDs are different program returns to the receive mode, otherwise last received GPS packet is analyzed, time UTC and position coordinates are extracted, then response frame is sent. This type of frame consists of 34 bytes including identifier of device and data from GPS packet as in Fig. 3. Obviously, the stationary device is in receive mode, listening and waiting for the response frame. After sending response frame mobile device waits for next asking frame, while stationary device sends received packet to the PC via UART (EIA-232).

Fig. 2 Asking Frame

Byte 0	Byte 1	Byte 2	Byte 3	Byte4
0x05	ID	QUE	0xBB	0xF5

Fig. 3 Response Frame

Byte 0	Byte 1-31	Byte 32	Byte 33
ID	data	<CR> (0x0A)	<LF> (0x0D)

3 Practical Verification of Parameters

The assumptions made for the system require verification in real environment because parameters presented in application notes [3, 10] were determined in laboratory environments. Parameters like communication range, battery lifetime, positioning dispersion while keeping on our requirements should be also correct with appropriate norm ETSI [4], which strictly defines communication parameters in short range distance, e.g. 500mW radiated power in usable radio frequency (869.4–869.65 MHz) or emission limit outside of the radio channel, which should be less than 250 nW.

Communication range

Accurate estimation of propagation path loss is a key factor for the good design of mobile systems [5]. The maximum communication range depends on output power (P_{out}) of transmitting device, sensitivity of receiving device and environment conditions. In this section Hata-Okumura model of path loss [7] is investigated. It takes into account the following parameters: frequency of radio signal, antenna height and also distance between a transmitter and a receiver. The original model has some limitations. The most restrictive is that Okumura's measurements were made at 1920 MHz, and Hata's formulas cover only frequencies range from 150 to 1500 MHz. Also antennas have been over average rooftop level. Path loss (PL_{th}) is calculated from Eq. (1).

$$PL_{th} = (A + B * log(d) + C) \tag{1}$$

A, B, C are related with radio frequency (fc) and antennas height (h_b, h_m), when d value is distance from transmitter to receiver in km. A, B, C can be obtained from (2):

$$A = 69.55 + 26.16 * log(f_c) - 13.82 * log(h_b) - a(h_m) \tag{2}$$

where $a(h_m)$ function is defined in urban (small and-medium size cities), suburban, rural environments as:

$$a(h_m) = (1.1 * log(f_c) - 0.7) * h_m - (1.56 * log(f_c) - 0.8) \tag{3}$$

and for metropolitan areas:

$$a(h_m) = \begin{cases} 8.29 * (log(1.54 * h_m))^2 - 1.1 & for\ f \leq 200\,MHz \\ 3.2 * (log(11.75 * h_m))^2 - 4.97 & for\ f \geq 400\,MHz \end{cases} \tag{4}$$

Table 1 Communication range

Pout (dBm)	9.5	13.5	17.5
Communication range (m)	800	1000	12000

B value depends only on h_b

$$B = 44.9 - 6.55(log(h_b)) \tag{5}$$

and C in urban and metropolitan is 0, in suburban environments:

$$C = -2 * [log(f_c/28)]^2 - 5.4 \tag{6}$$

and in rural areas:

$$C = -4.78 * [log(f_c)]^2 + 18.33 * log(f_c) - 40.98 \tag{7}$$

According to [7] path loss in dB has been calculated by substituting to the Eq. (1) distance values from 100 m to a maximum communication range (state, where RSSI parameter rise above maximum sensitivity level presented in Table 1) every 100m. Moreover, A, B, C values for four models of areas: rural, suburban environments, metropolitan areas, small and medium-size cities have been considered by substituting: $f_c = 869$ MHz (operating frequency), $h_m = 1$ m (mobile device is attached to the human belt), $h_b = 35$ m (antenna is on the building). Assumed data have been utilized in equations (2) to (7). Results have been presented in Table 2.

Communication range (distance) where RSSI is equal to sensitivity of receiver mainly depends on model of environment. In practice, increasing output power or antenna gain of 1dB leads to higher communication range of 106 % with respect to initial value. The real path loss value (PL_{real}) can be calculated by subtracting output power (P_{out}) from the RSSI indicator

$$PL_{real}[dB] = P_{out}[dBm] - RSSI[dB] \tag{8}$$

RSSI parameter must be read just after receiving data packet and then the RF module idle state can (should) be applied, otherwise RSSI parameter is calculated incorrect. RSSI value (dependent on the distance) is arithmetical average of one hundred samples collected for particular point (distance from stationary device). Comparison of measured path loss plots with Hata-Okumura modeled plots is presented in Fig. 4. Measured path loss have been calculated from the average RSSI for $P_{\text{out}} = 9.5/13.5/17.5$ dBm. The most upper and the most lower plots presents Hata-Okumura model for suburban and rural area, respectively. It can be noticed that measurements in real environment are comparable with model and place between upper and lower path loss. Measured results also depends on antenna, which gain could increase communication range or change the path loss. Our results has been

Table 2 Comparison of Hata-Okumara Path Loss with Measured

Distance (km)	Suburban environments	9.5 dBm	13.5 dBm	17.5 dBm	Rural area
0.1	103,05	90,65	100,30	77,00	63,16
0.2	113,52	101,37	103,38	87,92	73,64
0.3	119,65	100,47	102,00	94,51	79,76
0.4	123,99	102,43	104,68	100,08	84,11
0.5	127,36	110,89	115,10	97,81	87,48
0.6	130,12	117,16	113,59	95,07	90,23
0.7	132,45	109,42	114,30	97,77	92,56
0.8	134,46	122,30	114,48	94,58	94,58
0.9	136,24		117,73	89,87	96,36
0.10	137,84		125,09	113,82	97,95
0.11	139,28			116,02	99,39
0.12	140,59			108,50	100,71

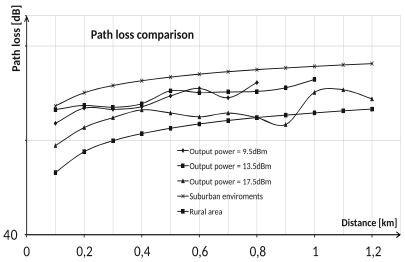


Fig. 4 Path loss comparison

obtained while using built-in RF module quarter wave antenna with 0 dBm gain, so total output power could be assumed as RF module output power.

According to the Table 2 and Fig. 4 the model of environment is placed parallelly between rural areas and suburban environments, however confirmation of the path loss curve is not preserved in all section of distance. In case of $P_{\text{out}} = 17.5\,\text{dBm}$ model of the environment remains closer to the rural area. In case of $P_{\text{out}} = 9.5\,\text{dBm}$ model of area remains nearly suburban environment. However calculated PL values were different than in case of various output power. Implementation of higher output power reduces path loss value, approximates area closer to the rural environment and finally confirms that communication range increased. Specific optimization of Hata model [5] gives high accuracy and it is able to predict path loss value, the results have been obtained for 800 MHz band. All transmitted information incurs path loss as electromagnetic waves propagate from source to destination (due to e.g., reflection, diffraction, and scattering). Specifically, it has been reported in many academic literature (e.g. [9]) that the propagation models applied for macrocell mobile systems have built-in-error (generally of the order of 7–10 dB standard deviation- a factor of ten in signal power) accounted for during the network design through a margin added

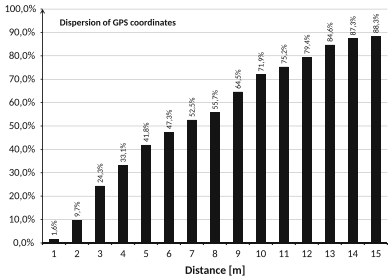


Fig. 5 Dispersion of GPS coordinates

to the overall signal strength calculations to take account of the natural signal fading phenomenon.

Static Object Localization

Localization accuracy of whole the system is based on GPS module. Therefore, GPS coordinates dispersion for static object is important. It has been calculated as average of 1000 samples measured by the mobile device. The histogram of cumulative sum of positions within a circle of the radius from 1 m up to 15 m is presented in Fig. 5. Based on the research: the first quartile equals to 3.06 m, the second quartile is 6.59 m and the third one is 10.98 m. It means 50 % of information received should be within a circle of radius smaller than 6.6 m.

Localization of an Object In-motion

The accuracy of a system in static position has been compared with accuracy measured during movement of the mobile device. In-motion accuracy is strictly connected with the cross tracking error which causes changes in positioning of following points. The cross tracking test has been performed for constant speed of the object, i.e. $v_1 = 0.8$ m/s, $v_2 = 1$ m/s, $v_3 = 1.5$ m/s and $v_4 = 3$ m/s for low, normal, high speed walking and for biking accordingly. It leads to distance shift of $d_1 = 0.8$ m, $d_2 = 1$ m, $d_3 = 1.5$ m and $d_4 = 3$ m, respectively. GPS module receives its current position and the distance shift between two points must fulfill the following equation:

$$x = \sqrt{(\lambda_{(N-1)} + \lambda_N)^2 + (\varphi_{(N-1)} + \varphi_N)^2} \tag{9}$$

where λ value means latitude, and the φ longitude. Obviously, Eq. 9 does not take into account the curvature of the Earth but for low speed and short displacement it can be neglected. If distance between two following points is greater than di then current position is discarded and next point is evaluated. In order to keep constant velocity, the distance is multiplied by two. If such point does not fulfill assumed speed again, the distance is tripled for next point. Table 3 indicates amount of acquired positions (in percent), which have been denied for constant velocity in area around 400m.

For walking speed every second coordinate is valuable. Velocity for which actual position has been estimated properly for more than 90 % is 3.5m/s. Coordinates dispersion depends on type of GPS module and GPS antenna placement. During

Table 3 In-motion dispersion of GPS coordinates

Velocity	0.8	1	1.5	3
Amount of probes	13.47 %	19.90 %	38.69 %	84.05 %

movement signal strength conditions are changing because of various satellites visibility.

Radio Channels Spacing and SNR

The localization system operates in ISM (Industrial, Science, Medical) and SRD (Short Range Device) band at 868.4 869.65 MHz frequency sub band. The band is divided into 16 channels of 58–812 kHz bandwidth (user programmable).

The power for a single channel cannot exceed 500mW in channel 0 and 25 mW in channel 1 etc. and the channel spacing must be at least 50 kHz. For frequencies lower than 869.7 MHz the spectrum access should be less than 10 %. The measured voltage is reduced by 3dB [10] for impedance matching (50Ω) between antenna and spectrum analyzer. According to [4] the maximum power emission value for frequencies higher than 869.650 MHz is 25mW. The power for a single channel cannot exceed 500mW in channel 0 and 25mW in channel 1 etc. and the channel spacing must be at least 50kHz. The maximum power rating is 500mW for all channels but the channel, which frequencies are lower than 869.7 MHz the spectrum access should be less than 10 %. For impedance matching (50Ω) between antenna and spectrum analyzer the measured voltage is reduced by 3dB [3]. According to [10] the maximum power emission value for frequencies bigger than 869.650 MHz is 25 mW. Comparison of power spectrum for channels 0 and 1 has been presented in Fig. 6. The radiated power should be less than 25 mW (13.97 dBm). The whole stated frequency band may be used as 1 wideband channel for high speed data transmission. For frequencies higher than 869.7 MHz maximum radiated power couldnt be greater than 5 mW. Channel separation is the difference between envelopes of channels at -6 dB lower than maximum output power. In our case equals to 134 kHz. A single channel bandwidth measured also at -6 dB level equals to 76 kHz. Both parameters fulfill ETSI norm requirements.

Battery Lifetime and Current Consumption

Presented system is composed of stationary device and mobile devices powered with battery. The system usability depends strongly on mobile unit lifetime while

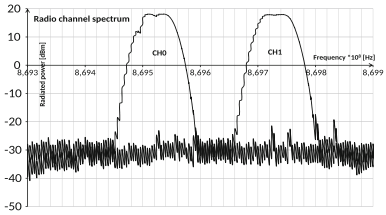


Fig. 6 Radio channel spectrum

the main energy consuming unit is the RF transmitter. For this reason, the settings for communication were determined taking into consideration energy preservation. Normal settings are: 40 question per minute, modulation GFSK, deviation 14 kHz, channel filter bandwidth equals 58 kHz, spacing 200 kHz. and 1.2 kbps baud rate. The mobile device is equipped with the Li-Ion battery of capacity equals to 1900 mAh. Using normal parameters the lifetime tests with different output power have been performed. Initial conditions for one test: battery is fully charged, stationary device enquire mobile unit 40 times for minute; a mobile device responses with data frame presented in Fig. 3, none of power saving attributes have been implemented. Battery voltage has been measured by ADC built in microcontroller. The test is completed when the battery voltage falls assumed minimum level, i.e. 2.75 V (determined on the battery application note). In such a case the onboard analog comparator turns off mobile device.

Battery lifetime mainly depends on RF module's power settings. Current consumption in transmit mode is 185.3, 141.2, 119.8 mA, for output power 17.5, 13.5, 9.5 dBm, respectively. Figure 7 presents battery voltage level versus time (in hours) for the constant number of enquires, i.e. 40 per minute. Device lifetime according to [8] can be calculated with:

$$T = C/I \tag{10}$$

where T means battery lifetime [in hours], C—maximum battery capacity in Ah, and I as discharge current. Battery voltage and current plots for output power equal to 13.5 dBm is presented in Fig. 8. Interval between following responses is 1.5 s Mobile device current consumption in transmit mode is 185.36 and 141.241 mA for 17.5 and 13.5 dBm, respectively. Obviously in case of none power saving attributes, current level depends mainly on RF module output power. Difference is result of current

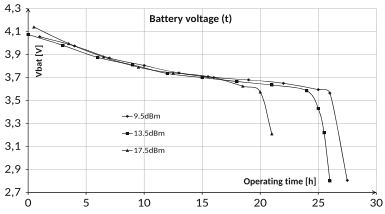


Fig. 7 Battery voltage level versus time

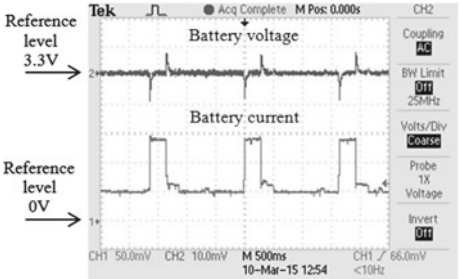


Fig. 8 Current consumption and battery voltage for 13.5 dBm

Table 4 Comparison of current consumption

Output power (dBm)	Current transmit (mA)	Current idle (mA)	Difference (mA)	Current theoretically (mA)
17.5 dBm	185.36	54,03	131.33	138
13.5 dBm	141.241	53,77	87.471	99

in non-transmitting state, which is on the same level regardless of output power. In application note [10] current consumption of RF module for considered output power settings should be 138 and 99 mA respectively. After subtracting idle current from transmit current we obtain proper values 131.33 and 87.471 mA for 17.5 and 13.5 dBm, respectively. Comparison of current consumption in transmit mode has been presented in Table 4. Average current consumed by the device can be calculated from the following formula:

$$I = \frac{1}{T}\int_{t_0}^{t_0+T} \frac{U_R(t)}{R} dt \tag{11}$$

where $U_R(t)$ means value of voltage waveform in a period T on resistor $R = 1\Omega$—which is connected in series with battery during test. Operating time (Eq. 11) is calculated based on the average current determined from the oscilloscope (Eq. 12) and capacity of the battery. Example of calculation of battery lifetime has been presented in Eq. 12.

$$T = \frac{C}{\int_t^{t_1} I_{transmission}(t)dt + \int_{t_1}^{t+T} I_{idle}} \tag{12}$$

where $I_{transmission}$ is current consumption in transmit mode, I_{idle} in idle mode, respectively. Period T is dependent on amount of sending frames in time unit. Comparison of theory calculated and measured lifetime is presented in Table 5. As can be noticed, the real time that device operates on fully charged battery is always less than theoretically calculated for all considered output powers.

Table 5 Relationship between output power and lifetime of battery

Output power (dBm)	17.5	13.5	9.5
Average current consumption (mA)	77,37	69.25	68.54
Operating time (theoretically) (h)	24.56	27.35	27.72
Operating time—measured (h)	21	26	27.5

4 Conclusion

Practical verification of mobile device parameters for system localization purpose have been presented in this paper. All tests confirmed usefulness because the parameters were comparable with those presented in the datasheet or application notes of elements. The Hata-Okumura model has been compared with real path loss within specific application area. Model calculations and real measurements of area mapping are different in case of various output power implementation. It is related to specific area under test. However all verification measurements are within rural and suburban area what agrees with reality. GPS localization for objects in-motion for various speed clearly shows decreasing location error with increasing velocity. For normal walking speed much more precise DGPS or AGPS should be applied. Usability of mobile unit is related to battery lifetime. Current consumption for the high power mode and 40 enquires per minute is low enough to give 21 h of lifetime. The minimum operating time should cover one worker shift. Time can be extended by power saving modes implementation. Implementation Wake-On-Radio function in RF modules should reduce current consumption in idle state. It would cause longer device usage especially when the number of enquires is low. BER (bit error rate) and PER (packet error rate) have not been tested at this stage but in this system radio communication link is crucial. For long distances and urban area (many obstacles) the radio link may disappear. Advanced algorithms for object localization should increase communication range by increasing output power only if it is necessary and also save more power for longer device usage.

Acknowledgments This material is based upon work supported by Silesian University of Technology under the project BK-266/RAu2/2014/502.

References

1. Boulos, K.B.G.: Real-time locating systems (rtls) in healthcare: a condensed primer. Int. J. Health Geogr. **11**(1), 2 (2012)
2. Brachman, A.: Simulation comparison of leach-based routing protocols for wireless sensor networks. In: Kwiecien, A., Gaj, P., Stera, P. (eds.) Computer Networks, CCIS, vol. 370, pp. 105–113. Springer, Berlin Heidelberg (2013)
3. GlobalTop Tech: FGPMMOPA4 datasheet, http://www.propox.com/download/docs/GPS_FGPMMOPA4.pdf
4. Institute, E.T.S.: ETSI EN 300 220 V2.3.1, Electromagnetic compatibility and Radio spectrum Matters (ERM); Short Range Devices (SRD); Radio equipment to be used in the 25 MHz to 1000 MHz frequency range with power levels ranging up to 500 mW
5. Isabona, J.K.C.: Urban area path loss propagation prediction and optimisation using hata model at 800 MHZ. IOSR J. Appl. Phys. **3**(4), 08–18 (2013)
6. Lewter, B., Moshell, J.M.W.J.: Inexpencive real time location systems (rtls) for event tracking using existing wifi infrastructures (2006)
7. Molisch, A.F.: Wireless communication, vol. 2. Wiley, New York (2011)
8. Park, S., Savvides, A.M.B.S.: Battery capacity measurement and analysis using lithium coin cell battery. In: ISPLED 2001. pp. 382–387. Huntington Beach, USA (2001)

9. Saunders, R.S., Belloul, B.: Ingenia R. Acad. Eng. Issue **12**, 36–40 (2002)
10. Seem, C., Ubostad, M.H.S.: Using the CC1190 Front End with CC1101 under EN 300 220. http://www.ti.com/lit/an/swra361a/swra361a.pdf
11. Unnikrishnan, S.A.A.: A mobile based tracking system for location prediction of moving object. Int. J. Adv. Technol. Eng. Sci. **2**(6), 19–25 (2014)

Perturbation Mappings in Polynomiography

Krzysztof Gdawiec

Abstract In the paper, a modification of rendering algorithm of polynomiograph is presented. Polynomiography is a method of visualization of complex polynomial root finding process and it has applications among other things in aesthetic pattern generation. The proposed modification is based on a perturbation mapping, which is added in the iteration process of the root finding method. The use of the perturbation mapping alters the shape of the polynomiograph, obtaining in this way new and diverse patterns. The results from the paper can further enrich the functionality of the existing polynomiography software.

Keywords Polynomiography · Perturbation · Aesthetic pattern · Computer art

1 Introduction

Today, one of the aims in computer aided design is to develop methods that make the artistic design and pattern generation much easier. Usually the most work during a design stage is carried out by a designer manually. Especially, in the cases in which the graphic design should contain some unique unrepeatable artistic features. Therefore, it is highly useful to develop an automatic method for aesthetic patterns generation. In the literature we can find many different methods, e.g., method based on Iterated Function Systems [13], method for creating stone-like decorations using marbling [11]. A very interesting method is polynomiography [6]. It is based on the root finding methods of polynomials with complex coefficients.

In this paper we present a modification of the standard rendering algorithm used in polynomiography. The modification is based on the use of perturbation mapping before the use of root finding method in the standard algorithm. The perturbation mapping disturbs the process of finding the roots of polynomial thereby obtaining new and diverse patterns comparing to the standard polynomiography.

K. Gdawiec (✉)
Institute of Computer Science, University of Silesia, Sosnowiec, Poland
e-mail: kgdawiec@ux2.math.us.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_42

The paper is organized as follows. In Sect. 2 we introduce some basic information about polynomiography and a standard algorithm for rendering polynomiographs. Then, in Sect. 3 we present perturbation mapping and its use in the polynomiography for obtaining new patterns. Some examples of polynomiographs obtained with the proposed modifications are presented in Sect. 4. Finally, in Sect. 5 we give some concluding remarks.

2 Polynomiography

The notion of polynomiography appeared in the literature about 2000 and was introduced by Kalantari. Polynomiography is defined as the art and science of visualization in approximation of the zeros of complex polynomials, via fractal and non-fractal images created using the mathematical convergence properties of iteration functions [7]. Single image created using the mentioned methods is called polynomiograph.

In polynomiography the main element is the root finding method. Many different root finding methods exist in the literature, e.g., Newton method [7], Traub-Ostrowski method [1], Harmonic Mean Newton's method [1], Steffensen method [10], and also we can find families of root finding method, e.g., Basic Family [7], Parametric Basic Family [7], Euler-Schröder Family [7], Jarratt Family [2]. Let us recall two root finding methods, that will be used in the examples presented in Sect. 4.

Let us consider a polynomial $p \in \mathbb{C}[Z]$, $\deg p \geq 2$ of the form:

$$p(z) = a_n z^n + a_{n-1} z^{n-1} + \cdots + a_1 z + a_0. \tag{1}$$

The Newton root finding method is given by the following formula:

$$N(z) = z - \frac{p(z)}{p'(z)}, \tag{2}$$

and Halley root finding method by the formula:

$$H(z) = z - \frac{2p'(z)p(z)}{2p'(z)^2 - p''(z)p(z)}. \tag{3}$$

To render a single polynomiograph we can use Algorithm 1. It is a basic rendering algorithm. In the literature we can find other methods of rendering polynomiographs, which are based on the ideas taken from the Mandelbrot and Julia set rendering algorithms [4]. Moreover, we can replace the Picard iteration used in the algorithm with other iteration methods [5], e.g., Mann, Ishikawa, Noor. In the algorithm we use the so-called iteration colouring, i.e., colour is determined according to the number of iteration in which we have left the while loop. Other colouring methods exist in the literature, e.g., basins of attraction, mixed colouring [7].

Algorithm 1 Rendering of polynomiograph

Input: $p \in \mathbb{C}[Z]$, $\deg p \geq 2$ – polynomial, $A \subset \mathbb{C}$ – area, M – number of iterations, ε – accuracy, $R : \mathbb{C} \to \mathbb{C}$ – root finding method, $colours[0..k]$ – colourmap.
Output: Polynomiograph for the area A.

```
for z_0 ∈ A do
    i = 0
    while i ≤ M do
        z_{i+1} = R(z_i)
        if |z_{i+1} − z_i| < ε then
            break
        i = i + 1
    Print z_0 with colours[i] colour
```

3 Perturbation Mappings in Polynomiography

In Algorithm 1 for any z_0 we can treat the sequence $\{z_0, z_1, z_2, \ldots\}$ as the orbit of z_0. For different starting points z_0 using the same root finding method we obtain different orbits. So, if we change some point in the orbit of a given starting point, then the orbit changes starting from the altered point. In this way we can obtain alternation of the polynomiographs shape.

Let us modify line 4 in Algorithm 1 in a following way:

$$z_{i+1} = (R \circ \rho)(z_i, i+1) = R(\rho(z_i, i+1)), \tag{4}$$

where $\rho : \mathbb{C} \times \mathbb{N} \to \mathbb{C}$ is a mapping. Moreover, we modify the convergence test in line 5 in a following way:

$$|z_{i+1} - \rho(z_i, i+1)| < \varepsilon. \tag{5}$$

The mapping ρ is called perturbation mapping and its aim is to alter (perturb) the orbit during the iteration process. Because we can alter the orbit in very different ways, so we do not make any assumptions about the perturbation mapping. Let us notice that when $\rho(z, i) = z$ for all $z \in \mathbb{C}$ and $i \in \mathbb{N}$, then (4) and (5) reduce to the standard iteration and convergence test used in the polynomiography.

The simplest perturbation mapping that alters the orbit is addition of a fixed complex number v, i.e.,

$$\rho_v(z, i) = z + v. \tag{6}$$

The value of v cannot be arbitrary, because we will lose the convergence of the root finding method and the resulting polynomiograph will be a rectangle filled with one colour. From the conducted research it turns out that the value v is highly dependent on ε. The modulus of v can be greater than ε only by a small value. Taking into

Algorithm 2 Rendering of polynomiograph with combined iteration

Input: $p \in \mathbb{C}[Z]$, $\deg p \geq 2$ – polynomial, $A \subset \mathbb{C}$ – area, M – number of iterations, ε – accuracy, $R : \mathbb{C} \to \mathbb{C}$ – root finding method, $\rho : \mathbb{C} \times \mathbb{N} \to \mathbb{C}$ – perturbation mapping, $\alpha \in \mathbb{C}$ – parameter, $colours[0..k]$ – colourmap.

Output: Polynomiograph for the area A.

```
for z_0 ∈ A do
    i = 0
    while i ≤ M do
        w = ρ(z_i, i + 1)
        z_{i+1} = αR(z_i) + (1 − α)R(w)
        if |z_{i+1} − w| < ε then
            break
        i = i + 1
    Print z_0 with colours[i] colour
```

account this observation it is very comfortable to represent v in the trigonometric form:

$$v = r\varepsilon(\cos\theta + \mathbf{i}\sin\theta), \tag{7}$$

where $r \in [0, 1.1]$ and $\theta \in [0, 2\pi)$.

Another example of perturbation mapping is mapping that uses different values of v in subsequent iterations, e.g.,

$$\rho_m(z, i) = \begin{cases} z + v_1, & \text{if } i \mod m = 0, \\ z + v_2, & \text{if } i \mod m = 1, \\ \dots & \\ z + v_m, & \text{if } i \mod m = m - 1, \end{cases} \tag{8}$$

where $m \in \mathbb{N}$ and $v_1, v_2, \dots, v_m \in \mathbb{C}$.

The examples of perturbation mappings presented so far are all deterministic. It is tempting to use randomness to obtain random patterns. But it turns out that the polynomiographs generated using random value of v in each iteration does not give a random pattern. We obtain a very similar noisy patterns, so their appearance is not aesthetic. Instead of using pure randomness we can use the random number generator of computer graphics [8], i.e., a noise function.

Besides the use of perturbation mapping we can also take combination of the standard iteration and the perturbed one. Let ρ be a given perturbation mapping and R a root finding method. We define new iteration process in the following way:

$$z_{i+1} = \alpha R(z_i) + (1 - \alpha)R(\rho(z_i, i + 1)), \tag{9}$$

where $\alpha \in \mathbb{C}$. Let us notice that for $\alpha = 1$ iteration (9) reduces to the standard iteration used in the polynomiography, and for $\alpha = 0$ it reduces to (4). So the combined iteration process is more general than the iteration with perturbation mapping.

Algorithm 2 presents method for rendering polynomiograph using the combined iteration process.

4 Examples

In this section, we present some examples of polynomiographs obtained using the proposed modifications from Sect. 3. To visually compare the obtained patterns with the originals ones we start by presenting the patterns obtained with the standard rendering algorithm (Algorithm 1). The patterns are presented in Fig. 1, and the parameters used to generate them were the following:

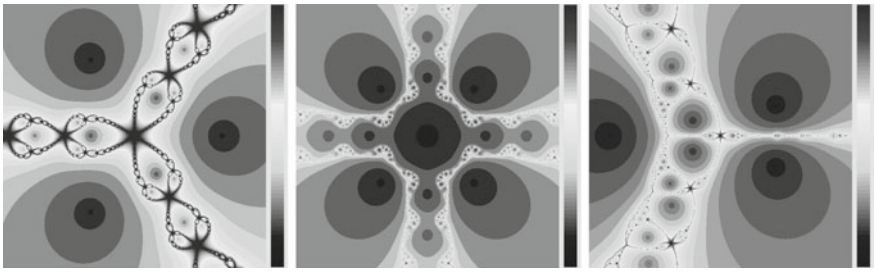


Fig. 1 Polynomiographs generated with the standard rendering algorithm

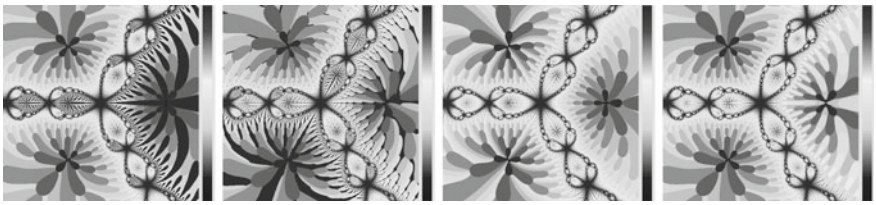


Fig. 2 Polynomiographs obtained using the perturbation mapping (6)—different arguments

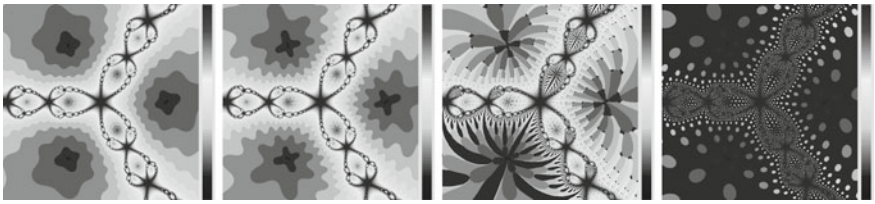


Fig. 3 Polynomiographs obtained using the perturbation mapping (6)—different moduli

(a) $p(z) = z^3 - 1$, $A = [-1.5, 1.5]^2$, $M = 15$, $\varepsilon = 0.001$, Newton's root finding method,
(b) $p(z) = z^5 + z$, $A = [-2.0, 2.0]^2$, $M = 15$, $\varepsilon = 0.001$, Halley's root finding method,
(c) $p(z) = z^3 - 3z + 3$, $A = [-2.5, 2.5]^2$, $M = 20$, $\varepsilon = 0.001$, Newton's root finding method.

The first example presents the use of perturbation mapping with the addition of a fixed complex number (6). The parameters to generate the polynomiographs were the same as in Fig. 1a, and the complex numbers used in the perturbation mapping had modulus equal to ε and their arguments were the following: (a) $\theta = 0.00$, (b) $\theta = 0.22$, (c) $\theta = 0.50$, (d) $\theta = 0.99$. The obtained polynomiographs are presented in Fig. 2.

The second example presents the influence of the modulus of the fixed complex number used in the perturbation mapping on the polynomiograph. The parameters to generate the polynomiographs were the same as in Fig. 1a, and the complex numbers used in the perturbation mapping had argument equal to 0.6π and their moduli were the following: (a) 0.6ε, (b) 0.9ε, (c) 1.0ε, (d) 1.1ε. The obtained polynomiographs are presented in Fig. 3.

The next example presents the use of perturbation mapping given by (8). The parameters to generate the polynomiographs were the same as in Fig. 1b, and the parameters of the complex numbers used in the perturbation mapping were the following (Fig. 4):

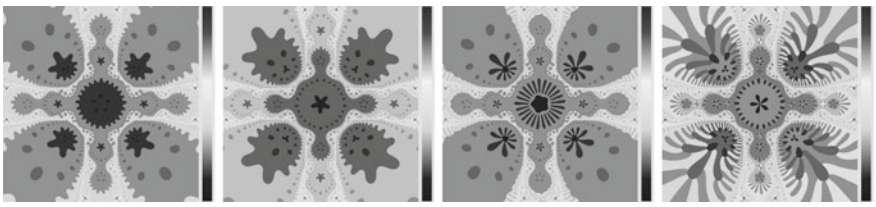


Fig. 4 Polynomiographs obtained using the perturbation mapping (8)

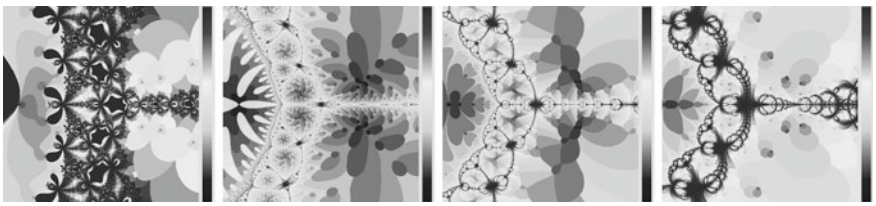


Fig. 5 Polynomiographs obtained using the combined iteration process (9)

(a)

$$\begin{cases} r = 0.95, \theta = 0.6\pi, & \text{if } i \mod 2 = 0, \\ r = 1.1, \theta = 1.5\pi, & \text{if } i \mod 2 = 1, \end{cases} \tag{10}$$

(b)

$$\begin{cases} r = 1.1, \theta = 1.5\pi, & \text{if } i \mod 2 = 0, \\ r = 0.95, \theta = 0.6\pi, & \text{if } i \mod 2 = 1, \end{cases} \tag{11}$$

(c)

$$\begin{cases} r = 1.1, \theta = 1.62\pi, & \text{if } i \mod 3 = 0, \\ r = 0.5, \theta = 0.98\pi, & \text{if } i \mod 3 = 1, \\ r = 1.0, \theta = 0.20\pi, & \text{if } i \mod 3 = 2, \end{cases} \tag{12}$$

(d)

$$\begin{cases} r = 1.0, \theta = 1.62\pi, & \text{if } i \mod 3 = 0, \\ r = 1.0, \theta = 0.49\pi, & \text{if } i \mod 3 = 1, \\ r = 1.01, \theta = 0.2\pi, & \text{if } i \mod 3 = 2. \end{cases} \tag{13}$$

The obtained polynomiographs are presented in Fig. 4.

The last example presents the use of combined iteration process (9). The parameters to generate the polynomiographs were the same as in Fig. 1c, the perturbation mapping was given by (6) with $v = \varepsilon(\cos\pi + \mathbf{i}\sin\pi)$, and the values of α were the following: (a) $-300 + 10\mathbf{i}$, (b) -10, (c) 100, (d) 200. The obtained polynomiographs are presented in Fig. 5.

5 Conclusions

In this paper, we presented a modification of the standard rendering algorithm for polynomiographs. The modification was based on the use of a perturbation mapping. The mapping was added in the iteration process of the root finding method in two ways. In the first method we used the perturbation mapping before the root finding method, and in the second method we used combination of the original root finding method and its perturbed version. Moreover, the convergence test of the algorithm was modified. The presented examples show that using the proposed methods we are able to obtain very interesting and diverse patterns, that differ from the original patterns obtained with the standard polynomiography.

In our further work we will try to extend the results of the paper by using the q-system numbers [9] and bicomplex numbers [12] instead of the complex numbers. Moreover, we will try to bring the perturbation mappings into the quaternion Newton method [3] and to develop an algorithm of visualization of the quaternionic root finding process in 3D.

References

1. Ardelean, G.: A comparison between iterative methods by using the basins of attraction. Appl. Math. Comput. **218**(1), 88–95 (2011)
2. Chun, C., Neta, B., Kim, S.: On Jarratt's family of optimal fourth-order iterative methods and their dynamics. Fractals **22**(4), 1450013 (2014)
3. Falcão, M.: Newton method in the context of quaternion analysis. Appl. Math. Comput. **236**, 458–470 (2014)
4. Gdawiec, K.: Mandelbrot- and Julia-like rendering of polynomiographs. In: Chmielewski, L., Kozera, R., Shin, B.S., Wojciechowski, K. (eds.) Computer Vision and Graphics, LNCS, vol. 8671, pp. 25–32. Springer (2014)
5. Gdawiec, K., Kotarski, W., Lisowska, A.: Polynomiography based on the non-standard Newton-like root finding methods. Abstr. Appl. Anal. **2015**, 797594 (2015)
6. Kalantari, B.: Two and three-dimensional art inspired by polynomiography. In: Bridges 2005, pp. 321–328. Banff, Canada (2005)
7. Kalantari, B.: Polynomial root-finding and polynomiography. World Scientific, Singapore (2009)
8. Lagae, A., Lefebvre, S., Cook, R., Rose, T.D., Drettakis, G., Ebert, D., Lewis, J., Perlin, K., Zwicker, M.: State of the art in procedural noise functions. In: Hauser, H., Reinhard, E. (eds.) State of the Art Reports, pp. 1–19. Norrköping, Sweden (2010)
9. Levin, M.: Discontinuous and alternate q-system fractals. Comput. Graph. **18**(6), 873–884 (1994)
10. Liu, X.D., Zhang, J.H., Li, Z.J., Zhang, J.X.: Generalized secant methods and their fractal patterns. Fractals **17**(2), 211–215 (2009)
11. Lu, S., Jaffer, A., Jin, X., Zhao, H., Mao, X.: Mathematical marbling. IEEE Comput. Graph. Appl. **32**(6), 26–35 (2012)
12. Wang, X.Y., Song, W.J.: The generalized M-J sets for bicomplex numbers. Nonlinear Dyn. **72**(1–2), 17–26 (2013)
13. Wannarumon, S., Bohez, E., Annanon, K.: Aesthetic evolutionary algorithm for fractal-based user-centered jewelry design. Artif. Intell. Eng. Des. Anal. Manuf. **22**(1), 19–39 (2008)

PCA Based Hierarchical Clustering with Planar Segments as Prototypes and Maximum Density Linkage

Jacek M. Leski, Marian Kotas and Tomasz Moroń

Abstract Clustering is an indispensable tool for finding natural boundaries among data. One of the most popular methods of clustering is the hierarchical agglomerative one. For data of different properties different versions of hierarchical clustering appear favorable. If the data possess locally linear form, application of hyperplanar prototypes should be advantageous. However, although a clustering method using planar prototypes, based on hierarchical agglomerative clustering with maximum density of planar segment linkage is known, it has a crucial drawback. It uses linear regression to model a cluster. When data for a cluster are parallel to the independent variable axis the use of linear regression can not be effective and the data are not described well. As a result, quality of the obtained group is low. The goal of this work is to overcome this problem by developing a hierarchical agglomerative clustering method that uses the PCA based maximum density of planar segment linkage. In the experimental part, we show that for data that possess locally linear form this method is competitive to the method of the agglomerative hierarchical clustering based on the maximum density of planar segment linkage.

Keywords Hierarchical clustering · PCA · Prototype based clustering · Maximum density linkage · Segments of hyperplanes

1 Introduction

One of the fundamental faculties of human being is clustering of similar cases for their later processing and classification [5]. The Swedish scholar C. Linnaeus even writes in one of his works that "All the real knowledge which we possess, depends on

J.M. Leski (✉) · M. Kotas · T. Moroń
Institute of Electronics, Silesian University of Technology, Gliwice, Poland
e-mail: jacek.leski@polsl.pl

M. Kotas
e-mail: marian.kotas@polsl.pl

T. Moroń
e-mail: tomasz.moron@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_43

methods by which we distinguish the similar from the dissimilar". Thus, clustering plays an important role in many engineering fields such as image segmentation, data mining, pattern recognition, signal processing, Web mining, modeling, communication, and so on [1, 7, 12, 13, 15–17, 19, 20]. The clustering methods divide a set of N vector observations $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N \in \mathbb{R}^t$ into c groups denoted $\Omega_1, \Omega_2, \ldots, \Omega_c$ so that the members of the same group are more similar to one another than to the members of the other groups [4].

Generally, clustering methods can be divided into [6]: hierarchical, graph theoretic, decomposing a density function, minimizing a criterion function. In this paper hierarchical agglomerative clustering will be applied. Clustering finds important applications in, for instance [1, 5, 20]: medicine, psychiatry, chemistry, marketing, biology, sociology, astrophysics, education, and archeology. In biological taxonomy, which originates hierarchical clustering, species are grou-ped together into genera, genera into orders, orders into classes, classes into types, types into varieties, which finally compose kingdoms [4, 6, 10]. Thus the insignificant discrepancies between groups are neglected, and they are gradually merged into larger groups. This approach is usually called as agglomeration. In the antithetic approach, which will not be considered in this work, the groups are split on the basis of some differences among their members. It is called as divisive hierarchical clustering.

The agglomerative approach aims to create c groups on the basis of N data vectors in the following way. We initially create N groups, with each group containing one data vector, only. In the next step, we seek for two groups differing the least (with the greatest similarity) and we join them, reducing the number of groups to $N-1$. In the succeeding steps, we continue combining the least differing groups, obtaining $N-2, N-3, \ldots, c$ groups.

The measures of similarity or dissimilarity between any two groups Ω_ℓ and Ω_r are needed to proceed with the agglomerative approach to clustering. A measure of the groups dissimilarity, which can be understood as a distance (referred in clustering as a linkage) between data groups, can be defined as [3, 5, 10]:

- single linkage $d_{\min}(\Omega_\ell, \Omega_r) = \min_{\mathbf{x}' \in \Omega_\ell,\ \mathbf{x}'' \in \Omega_r} d(\mathbf{x}', \mathbf{x}'')$,
- complete linkage $d_{\max}(\Omega_\ell, \Omega_r) = \max_{\mathbf{x}' \in \Omega_\ell,\ \mathbf{x}'' \in \Omega_r} d(\mathbf{x}', \mathbf{x}'')$,
- average linkage $d_{\text{ave}}(\Omega_\ell, \Omega_r) = \frac{1}{|\Omega_\ell||\Omega_r|} \sum_{\mathbf{x}' \in \Omega_\ell} \sum_{\mathbf{x}' \in \Omega_r} d(\mathbf{x}', \mathbf{x}'')$,
- centroid linkage $d_{\text{mean}}(\Omega_\ell, \Omega_r) = d(\mathbf{m}_\ell, \mathbf{m}_r)$,
- median linkage $d_{\text{median}}(\Omega_\ell, \Omega_r) = d(\mathfrak{m}_\ell, \mathfrak{m}_r)$
- minimum variance linkage $d_{\text{mv}}(\Omega_\ell, \Omega_r) = \sqrt{\frac{|\Omega_\ell||\Omega_r|}{|\Omega_\ell|+|\Omega_r|}} d(\mathbf{m}_\ell, \mathbf{m}_r)$,
- minimax linkage $d_{\text{minimax}}(\Omega_\ell, \Omega_r) = \min_{\mathbf{x}' \in \Omega_\ell \cup \Omega_r} \left[\max_{\mathbf{x}'' \in \Omega_\ell \cup \Omega_r} d(\mathbf{x}', \mathbf{x}'') \right]$,

where $d(\mathbf{x}', \mathbf{x}'')$ is the dissimilarity between data points $\mathbf{x}'$ and $\mathbf{x}''$, $|\Omega_r|$ is the cardinality of the rth cluster, $\cup$ is the set-theoretic union operation, $\mathbf{m}_r$ and $\mathfrak{m}_r$ denote rth cluster mean and median, respectively.

Some of the mentioned above dissimilarities can be calculated iteratively [11, 18]. In other words, the dissimilarities between a group and the union of other two groups may be expressed as a function of the dissimilarities among these groups before their union.

The hierarchical clustering provides a simple way to infer on groups number. The results of the clustering can graphically be presented with a dendrogram. The height of its node expresses the dissimilarity between the union of corresponding groups. Thus by cutting the dendrogram at height ξ, we obtain information on the number of the groups whose dissimilarities to each other are at lest equal to ξ.

The fundamental drawback of the hierarchical methods of clustering is the lack of group prototypes which would be beneficial for the results interpretability. In [3] the minimax linkage was introduced. The $\mathbf{x}$ datum whose distance to the farthest point within the union of groups is smallest becomes a prototype of the resulting group: $\Omega_\ell \cup \Omega_r$. The distance itself is taken as the measure of dissimilarity between the considered pair of groups $d_{\text{minimax}}(\Omega_\ell, \Omega_r)$. This dissimilarity measure can be interpreted as the radius of the minimal hyperball containing the resulting union of groups and centered at the prototype. Unfortunately, contrary to the classical linkages, for this one the dissimilarities cannot be calculated iteratively.

One of the most popular clustering methods based on minimization of a criterion function is the fuzzy c-means one. Its generalization by application of hyperplane shaped prototypes of the clusters is known as the Fuzzy C-Regression Models (FCRM) method [8]. The basic disadvantage of the FCRM is the infinite extent of such prototypes which can cause addition to a cluster of very distant data points, not necessarily similar to the majority within the cluster.

To overcome the above disadvantage the agglomerative hierarchical clustering based on Maximum Density of Planar Segment linkage (MDPS) was introduced in [14]. However, the above mentioned hierarchical clustering has a crucial drawback. It uses linear regression to model a cluster. When data for a cluster are parallel to the independent variable axis, the use of linear regression can not be effective. During calculations we have to invert a matrix that is singular or close to singular. As a result, the calculations are unstable and the description of the obtained group is poor.

The goal of our work is to introduce a method of hierarchical clustering with the prototypes confined to the segments of hyperplanes determined with the use of principal component analysis (PCA). By this confinement, we are going to overcome the aforementioned unfavorable property of the planar prototypes. It will be investigated if the method is competitive with respect to the hierarchical method with the maximum density of planar segment linkage.

2 Hierarchical Clustering with Planar Segments as Prototypes

In this section we recollect the notion of hierarchical clustering based on maximum density of planar segment linkage. This method allows the prototypes to be t-dimensional linear varieties rather than points. As might be expected, this type of clustering is most applicable to data which consist of clusters that are drawn from linear varieties of the same dimension. We deal with such problems when we analyze biomedical signals and medical databases, where some dependent variable y is explained by many independent variables $\mathbf{x}$. For instance a signal amplitude depends on the time variable and the gray level on a tomographic image depends on the spatial coordinates. In such cases the vector of features of the clustered objects must be split into independent variables $\mathbf{x}$ and the dependent one y, i.e. $\mathbf{z} = [\mathbf{x}^\top \ y]^\top$. Let Ω be a set containing data pairs $\Omega = \{(\mathbf{x}_k, y_k) | \, k \in \ell_\Omega\}$, where the independent observation $\mathbf{x}_k \in \mathbb{R}^{(t-1)}$ has a corresponding dependent observation $y_k \equiv x_{k,t} \in \mathbb{R}$, ℓ_Ω is the set of indices for the elements of Ω. $\mathrm{Card}(\Omega) = N_\Omega$ denotes the cardinality of Ω.

The parameters of a regression model describing the elements of Ω, are estimated by minimizing the sum of the squared errors [14]
$J(\mathbf{w}_\Omega) = (\mathbf{X}_\Omega \mathbf{w}_\Omega - \mathbf{y}_\Omega)^\top (\mathbf{X}_\Omega \mathbf{w}_\Omega - \mathbf{y}_\Omega)$, where $\mathbf{X}_\Omega = \left[\left(\mathbf{x}_k^\top \ 1\right)\right]_{k \in \ell_\Omega} \in \mathbb{R}^{N_\Omega \times t}$ and $\mathbf{y}_\Omega = [y_k]_{k \in \ell_\Omega} \in \mathbb{R}^{N_\Omega}$. If we denote $\mathbf{w}_\Omega^\star = \arg\min J(\mathbf{w}_\Omega)$, the optimality condition is as follows $\mathbf{w}_\Omega^\star = \left(\mathbf{X}_\Omega^\top \mathbf{X}_\Omega\right)^{-1} \mathbf{X}_\Omega^\top \mathbf{y}_\Omega$. The measure of the planar segment density is introduced in [14]

$$\mathcal{D}(\Omega) = \frac{\mathcal{R}(\Omega)}{N_\Omega} \sqrt{J(\mathbf{w}_\Omega^\star)}, \tag{1}$$

where

$$\mathcal{R}(\Omega) = \prod_{j=1}^{t-1} \left[\max_{k \in \ell_\Omega} \left(x_{k,j}\right) - \min_{k \in \ell_\Omega} \left(x_{k,j}\right) \right], \tag{2}$$

$x_{k,j}$ denotes the jth component (feature) of the kth datum. Measure (1) is a product of two terms: the density of the data projected on the independent variables subspace, and the regression errors. This measure favors data sets of high cardinality whose elements are located close to the hyperplane. Finally, the maximum density of planar segment (MDPS) linkage between two clusters Ω_ℓ and Ω_r is defined as [14] $d_{\mathrm{MDPS}}(\Omega_\ell, \Omega_r) = \mathcal{D}(\Omega_\ell \cup \Omega_r)$, i.e. the distance between the groups is expressed as the defined density measure (1) corresponding to the resulting merged cluster. However, the above mentioned hierarchical agglomerative clustering with maximum density of planar segment linkage has a crucial drawback. This method is based on a linear regression model of clusters. When data for a cluster are parallel to the axis of the independent variable, the method using linear regression can not describe them well. More precisely, in solution we have the inverse of $\left(\mathbf{X}_\Omega^\top \mathbf{X}_\Omega\right)$ matrix, which is

singular or close to singular. As a result, description of the obtained group is poor (unstable). In the next section of this work this problem is solved by developing a hierarchical agglomerative clustering method that uses the PCA based maximum density of planar segment linkage.

3 PCA Based Maximum Density of Planar Segment Linkage

Principal component analysis can be seen as a sequential constructing of a linear manifold approximating a set of points by its projections. These projections are ordered in variance and uncorrelated. Let Ω be a set containing data $\Omega = \{\mathbf{x}_k | k \in \ell_\Omega\}$, where $\mathbf{x}_k \in \mathbb{R}^t$ and ℓ_Ω is the set of indices for the elements of Ω with the cardinality $\mathrm{Card}(\Omega) = N_\Omega$.

Let's denote the rank $(t-1)$ linear model representing Ω as $\mathbf{f} = \mathbf{a} + \mathbf{V}_{t-1}\mathbf{b}$, where $\mathbf{a}$ denotes a location in $\mathbb{R}^t$, $\mathbf{V}_{t-1}$ is a $t \times (t-1)$ matrix with orthogonal vectors in columns, and $\mathbf{b}$ is a vector of parameters. Fitting the above linear model to data from cluster Ω by minimizing least squares leads to [9]: $\mathbf{a} = \overline{\mathbf{x}}$, $\mathbf{b}_i = \mathbf{V}_{t-1}^\top \mathring{\mathbf{x}}_i$, where $\mathring{\mathbf{x}}_i = \mathbf{x}_i - \overline{\mathbf{x}}$. Matrix $\mathbf{V}_{t-1}$ is obtained from singular value decomposition of centered data $\mathring{\mathbf{X}} = [\mathring{\mathbf{x}}_1, \mathring{\mathbf{x}}_2, \ldots, \mathring{\mathbf{x}}_{N_\Omega}]$ [9]:

$$\mathring{\mathbf{X}}^\top = \mathbf{U}\mathbf{D}\mathbf{V}^\top, \tag{3}$$

$\mathbf{U}, \mathbf{V}$ are $N_\Omega \times t$ and $t \times t$ orthogonal matrices, with the columns of $\mathbf{V}(\mathbf{U})$ spanning the column (row) space of $\mathring{\mathbf{X}}$. $\mathbf{D}$ is $t \times t$ diagonal matrix, with diagonal entries $d_1 \geq d_2 \geq \cdots \geq d_t$ called the singular values of $\mathring{\mathbf{X}}$. The columns of $\mathbf{V}$ are also called the principal components directions of $\mathring{\mathbf{X}}$, and $\mathbf{V}_{t-1}$ is obtained from $\mathbf{V}$ using its the first $t-1$ columns. The variance of ith principal component is equal to d_i^2. We define the following measure of the planar segment density

$$\mathcal{D}_{\mathrm{PCA}}(\Omega) = \frac{\mathcal{R}_{\mathrm{PCA}}(\Omega)}{N_\Omega}\, d_t, \tag{4}$$

where

$$\mathcal{R}_{\mathrm{PCA}}(\Omega) = \prod_{j=1}^{t-1} \left[\max_{k \in \ell_\Omega} \left(\breve{x}_{k,j} \right) - \min_{k \in \ell_\Omega} \left(\breve{x}_{k,j} \right) \right], \tag{5}$$

$\breve{x}_{k,j}$ denotes the jth component of the kth datum after projecting it on the rank $(t-1)$ linear manifold spanned on $\mathbf{V}_{t-1}$. Measure (4) is as previously a product of two terms. The first term is the density of the data projected on rank $(t-1)$ linear manifold (hypervolume of a hyperrectangle (orthotope) containing all points, divided by the number of these points); in contrast to the previous section, the hyperrectangle sides are not parallel to the respective coordinates of $\mathbf{x}$. The second term is the standard deviation of the last principal component (corresponding to the regression errors).

The PCA based maximum density of planar segment (PCAMDPS) linkage between two clusters Ω_ℓ and Ω_r is defined as $d_{\text{PCAMDPS}}(\Omega_\ell, \Omega_r) = \mathcal{D}_{\text{PCA}}(\Omega_\ell \cup \Omega_r)$. This measure can only be applied for sets of the sufficient cardinality. For $N_\Omega < t$ the measure is undefined. For $N_\Omega = t$ this measure equals zero. Thus, it can only be applied to data clusters of greater cardinality: $N_\Omega \geq t$.

Therefore in the initial phase of the algorithm execution, we apply another measure of the data similarity, a modification of the minimax linkage [14]:

$$\mathcal{D}(\Omega) = \begin{cases} \min\limits_{\mathbf{x}\in\Omega} \left[\max\limits_{\mathbf{x}'\in\Omega} d(\mathbf{x}, \mathbf{x}') \right], & N_\Omega \leq t + \xi, \\ \gamma, & N_\Omega > t + \xi, \end{cases} \tag{6}$$

where γ is an extremely large positive number (denoted as *inf* in MATLAB, being the IEEE arithmetic representation for positive infinity), ξ is a small integer value. Application of this measure results in forming the clusters confined to the hyperspheres of minimal radius, but cardinality not exceeding $t + \xi$. In the experiments that will be presented, we used $t + \xi = 3 + 1 = 4$ (assuming the minimal surplus of clusters cardinality that prevents the clusters to have by definition a zero value of the PCAMDPS linkage in the 3-dimensional space). When this condition is satisfied, we can proceed using the PCAMDPS linkage. This algorithm is called the agglomerative hierarchical clustering based on PCAMDPS linkage, and can be denoted as

`Algorithm PCAMDPS:`

1. Fix a number of clusters c $(1 \leq c < N)$.
2. Let $\Omega_\ell = \{x_\ell\}$ for $\ell = 1, 2, \ldots, N$. Set the current number of clusters $p = N$.
3. Repeat until the cardinality of each cluster is greater than $t + \xi$:
   (a) Find a pair of the nearest clusters (denoted as Ω_{ℓ_1} and Ω_{ℓ_2}) using a modified minimax linkage (6).
   (b) Merge Ω_{ℓ_1} and Ω_{ℓ_2}, delete Ω_{ℓ_1}, and decrement p by one.
4. Repeat until $c < p$:
   (a) For each pair of clusters (denoted as Ω_{r_1} and Ω_{r_2}) do:
      i. Calculate the singular value decomposition of centered data $\Omega = \Omega_{r_1} \cup \Omega_{r_2}$ (see (3)).
      ii. Calculate the PCA based maximum density of planar segment linkage between Ω_{r_1} and Ω_{r_2} clusters using (4).
   (b) Find the nearest pair of clusters (denoted as Ω_{ℓ_1} and Ω_{ℓ_2}) using the above linkages.
   (c) Merge Ω_{ℓ_1} and Ω_{ℓ_2}, delete Ω_{ℓ_1}, and decrement p by one.
5. Stop.

Remark. Please note that for $c = 1$ we obtain the full dendrogram.

4 Numerical Experiments

The aim of these experiments is to investigate the proposed method performance on simple three-dimensional data that are similar to those studied on Fig. 6 in [2]. They are the so-called goldfish datasets, presented in Fig. 1. As we can see, they consist of two three-dimensional letters 'X' and 'O', visually well separated. The figure presents the results of the agglomerative hierarchical clustering based on MDPS linkage (6 clusters are formed by cutting the dendrogram at level six). For comparison the results of clustering with the use of the agglomerative hierarchical clustering based on PCAMDPS linkage are presented in Fig. 2 (with the same number of clusters). Analyzing the above figures, we can see that both clustering methods managed to describe the data pretty well. Within letter 'O' in Fig. 1, however, we can discern many points that are not assigned well to the formed clusters. This is caused by relatively wide spread of these points in the vertical direction. They could have been approximated best by a vertical surface but the method based on linear regression (MDPS) was not able to form such a surface.

The PCAMDPS linkage based method managed to solve the problem much better: all points lie closely to the determined linear manifolds. In the next experiment, we rotated the data by $-\pi/12$ to make one branch of letter 'X' be approximately vertical (see Fig. 3). In the figure we can notice that construction of a cluster describing this branch of the letter failed completely. Again, this problem results from the MDPS method inability to describe data that are parallel to the independent variable axis. By contrast, in Fig. 4 we can see that the PCAMDPS linkage based method deals with the description of the rotated letter 'X' without any troubles. The quality of the

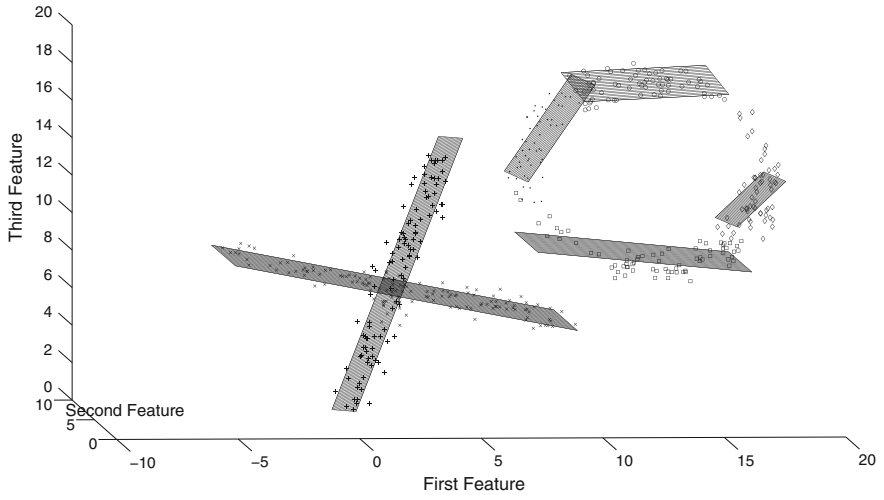


Fig. 1 A scatter plot of the 3-dimensional goldfish dataset used in the experiment, with the regression surfaces estimated by the MDPS linkage method (dendrogram is cut at level six)

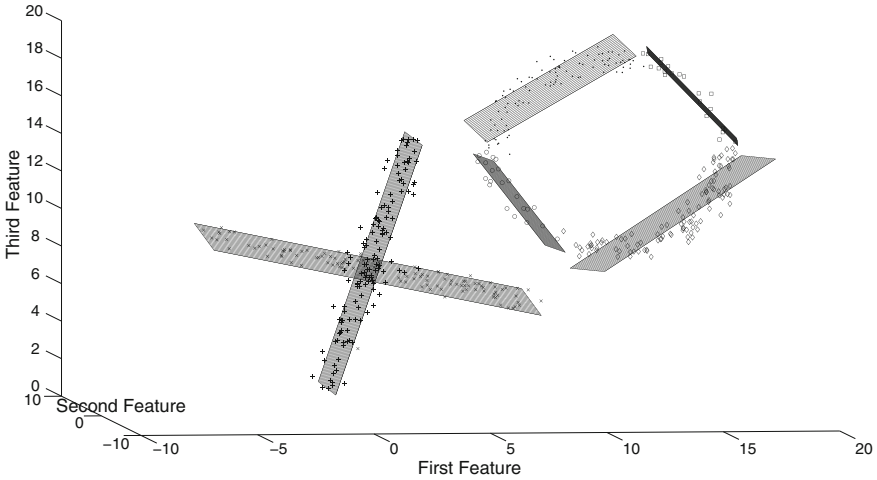


Fig. 2 A scatter plot of the 3-dimensional goldfish dataset used in the experiment, with the regression surfaces estimated by the PCAMDPS linkage method (dendrogram is cut at level six)

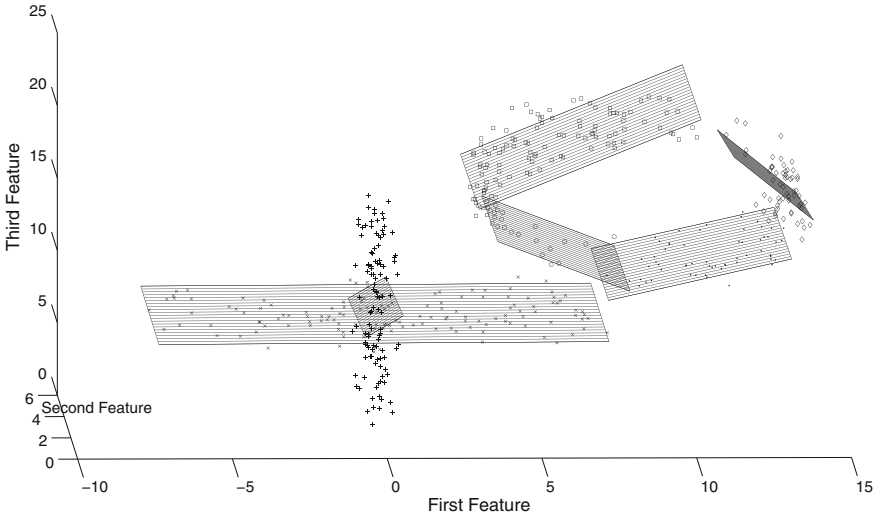


Fig. 3 A scatter plot of the 3-dimensional goldfish dataset from Fig. 1 rotated by $-\pi/12$, with the regression surfaces estimated by the MDPS linkage method (dendrogram is cut at level six)

dataset approximation by segments of hyperplanes is estimated with the root mean squared error (RMS). An error is defined as the difference between a datum and its nearest prototype. For the investigated database, we have presented in Table 1 the relation between PCAMDPS and MDPS methods for different rotation angles.

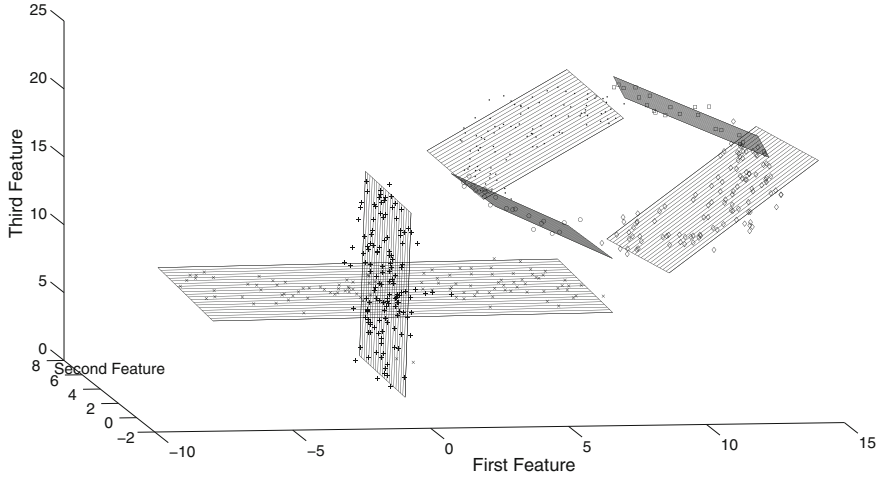


Fig. 4 A scatter plot of the 3-dimensional goldfish dataset from Fig. 1 rotated by $-\pi/12$, with the regression surfaces estimated by the PCAMDPS linkage method (dendrogram is cut at level six)

Table 1 Quantitative results of the experiment, obtained for the data rotated by different angles

Method	$-\frac{5\pi}{12}$	$-\frac{4\pi}{12}$	$-\frac{3\pi}{12}$	$-\frac{2\pi}{12}$	$-\frac{\pi}{12}$	0	$\frac{\pi}{12}$	$\frac{2\pi}{12}$	$\frac{3\pi}{12}$	$\frac{4\pi}{12}$	$\frac{5\pi}{12}$
MDPS	2.039	0.770	0.660	0.725	0.744	0.945	2.089	0.891	0.772	0.988	0.947
PCAMDPS	0.582	0.582	0.582	0.582	0.582	0.582	0.582	0.582	0.582	0.582	0.582

5 Conclusions

We have shown that the method of agglomerative hierarchical clustering based on PCA linkage allows the user to describe easily the distribution of the data groups with the segments of hyperplanes in every stage of clustering. The PCA based maximum density of planar segment linkage can be a useful alternative to the other definitions of the distance between merged clusters as well as to the previously used linear regression based maximum density of planar segment linkage. In contrast to the method of fuzzy c-regression models, the method proposed confines the extent of the planar prototypes.

The numerical example is given to illustrate the validity of the PCA based maximum density of planar segment linkage when applied to 3-dimensional datasets. This numerical example shows the usefulness of this method for data modeling, especially when data for a cluster are parallel to the independent variable axis and the use of linear regression can not be effective, and the data are not described well. The comparison of the PCA based hierarchical clustering with maximum density of planar segment linkage with the linear regression based maximum density of planar segment linkage reveals that the introduced method prevails in clustering the data with locally planar distribution.

Acknowledgments This research was partially supported by statutory funds (BK-2015) of the Institute of Electronics, Silesian University of Technology and GeCONiI project (T. Moroń). The work was performed using the infrastructure supported by POIG.02.03.01-24-099/13 grant: GeCONiI–Upper Silesian Center for Computational Science and Engineering.

References

1. Aggarwal, C., Reddy, C.: Data clustering. Algorithms and applications. CRC Press, Boca Raton (2014)
2. Bezdek, J.: Pattern recognition with fuzzy objective function algorithms. Plenum Press, New York (1982)
3. Bien, J., Tibshirani, R.: Hierarchical clustering with prototypes via minimax linkage. J. Ame. Stat. Assoc. **106**(495), 1075–1084 (2011)
4. Duda, R.O., Hart, P.E.: Pattern classification and scene analysis. Wiley, New York (1973)
5. Everitt, B., Landau, S., Leese, M., Stahl, D.: Cluster analysis, 5th edn. Wiley, London (2011)
6. Fukunaga, K.: Introduction to statistical pattern recognition. Academic Press, San Diego (1990)
7. Gan, G., Ma, C., Wu, J.: Data Clustering: theory, algorithms, and applications. SIAM, Philadelphia (2007)
8. Hathaway, R., Bezdek, J.: Switching regression models and fuzzy clustering. Trans. Fuzzy Syst. **1**(3), 195–204 (1993)
9. Hestie, T., Tibshirani, R.J.F.: The elements of statistical learning. data mining, inference, and prediction, 2nd edn. Springer, New York (2008)
10. Kaufman, L., Rousseeuw, P.: Finding groups in data. An Introduction to cluster analysis. Wiley, Hoboken (1990)
11. Lance, G., Williams, W.: A general theory of classificatory sorting strategies: hierarchical systems. Comput. J. **9**, 373–380 (1967)
12. Leski, J.: An ε-margin nonlinear classifier based on fuzzy if-then rules. IEEE Trans. Syst. Man, Cybern. Part B: Cybern. **34**(1), 68–76 (2004)
13. Leski, J.: Fuzzy (c+p)-means clustering and its application to a fuzzy rule-based classifier: towards good generalization and good interpretability. Trans. Fuzzy Syst. (2014). doi:10.1109/TFUZZ.2327995. (Accepted for publication)
14. Leski, J., Kotas, M.: Hierarchical clustering with planar segments as prototypes. Pattern Recognit. Lett. **54**, 1–10 (2015)
15. Mirkin, B.: Clustering. A data recovery approach. CRC Press, Boca Raton (2013)
16. Ripley, B.: Pattern recognition and neural networks. Cambridge University Press, Cambridge (1996)
17. Tou, J., Gonzalez, R.: Pattern recognition principles. Addison-Wesley, London (1974)
18. Ward, J.: Hierarchical grouping to optimise an objective function. J. Am. Stat. Assoc. **58**, 236–244 (1963)
19. Webb, A.: Statistical pattern recognition. Arnold, London (1999)
20. Xu, R., Wunsch II, D.: Clustering. Wiley, Hoboken (2009)

Feature Thresholding in Generalized Approximation Spaces

Dariusz Małyszko

Abstract Recent advances in information sciences are extending classical set theory frontiers into new domains of uncertainty perception, incompleteness, vagueness of knowledge—giving new mathematical approach to development of intelligent information systems. The paper addresses the problem of construction of rough measures in generalized approximation spaces introducing a new method of rough feature thresholding. The algorithm creates rough feature blocks and assigns them image blocks from the block min, avg, max statistics. The algorithm converts data blocks into rough approximations of feature blocks. The introduced solution contributes to the highly precise internal data structure descriptors on one side and constitutes the algorithmic base for rough data analysis entirely embedded in generalized approximation spaces at the same time. The scope of possible applications includes image descriptors, image thresholding, image classifications.

Keywords Approximation spaces · Generalized approximation spaces · Rough sets · Thresholding

1 Introduction

In recent days, development of intelligent information systems requires construction of more robust and reliable data analysis algorithms. Most important methods employed in data processing involve data granulation, clustering and thresholding. Rough set theory presents methodology for handling unprecise and vague information. Rough Extended Framework presented in [5–7] extensively developed method of data analysis based upon data structure inferred from metric relations in rough, fuzzy and probabilistic approach. The theory of rough sets and fuzzy sets have applied in many image analysis algorithms as described in [8]. Support vector machines in rough sets has been presented in [4]. Kernelized rough sets are introduced in [9]. Object-based image retrieval method has been proposed in [2]. Image retrieval

D. Małyszko (✉)
Faculty of Computer Science, Bialystok University of Technology, Bialystok, Poland
e-mail: d.malyszko@pb.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_44

insensitive to location variations has been presented in [1]. Rough sets theory can also be combined with various computational intelligence techniques. Rough sets extensions into covering spaces are extensively explored as in [10, 11].

The introduced solution gives algorithm that describes data structure by means of measures of feature lower and upper approximations that act as data descriptor or image descriptor. Rough feature descriptor presents universal solution that can be applied to any numerical features, incorporates data granularity dependent upon block sizes and feature partition sizes, presents mapping into rough data descriptors by giving image data rough measures. The algorithm presents novelty interoperability between rough-fuzzy and rough probabilistic approximation spaces.

The main contribution of the paper is in presentation of concise, exact feature thresholding algorithm for generalized approximation spaces with thresholding algorithmic scheme embedded in generalized approximation spaces. Introduced rough extended model describes object properties by means of its metric relation to feature blocks in rough, fuzzy and probabilistic setting. The universality of the algorithm comes from independence from the selected image features, data granularity, adaptability.

The paper is structured in the following way. In Sect. 2 the introductory information on generalised approximation spaces, related rough models and new rough extended model has been presented. In Sect. 3 description of feature thresholding algorithm has been given. In Sect. 4 experimental setup and experimental results are given. Main achievements and future research is presented as the summary of the paper.

2 Generalized Approximation Spaces

Similar indiscernible objects create an elementary set describing our knowledge about the universe. Elementary sets are referred to as precise sets. All other sets are called rough, imprecise, vague. An information system IS $= (U, A)$ consists of objects described by attributes. Information system with objects divided into classes by reflexive, symmetric and transitive equivalence relation R on U is called an approximation space. Lower and upper approximations of any subset X of U are defined as two sets completely contained or partially contained in the equivalence classes. This approach has been generalized by extending equivalence relations to tolerance relations, similarity relations, binary relations leading into formulation of the concept of Generalized approximation spaces.

A generalized approximation space can be defined as a tuple GAS $=$ (U, N, υ) where N is a neighbourhood function defined on U with values in the powerset $P(U)$ of U. The overlap function υ is defined on the Cartesian product $P(U) \times P(U)$ with values in the interval [0, 1] measuring the degree of overlap of sets. The lower GAS_* and upper GAS^* approximation operations can be defined in a GAS by

$$GAS_*(X) = \{x \in U : \upsilon(N(x), X) = 1\} \tag{1}$$

$$GAS^*(X) = \{x \in U : \upsilon(N(x), X) > 0\} \tag{2}$$

Generalized approximation spaces present environments with specialized rough set models applied such as similarity based rough set model with reflexive neighborhood function and variable precision model with different thresholds definition in overlap functions.

Introducing metric function in generalized approximation spaces allows for finding similarity degree of points and sets. Measure is a function that for each set assigns a number that describes this set properties. Rough measure describes rough properties of the set. In image analysis, objects are described by features, that most often are embedded in metric spaces. Rough extended model for generalized approximation spaces, defines for each set, rough measures on the base of its rough, fuzzy and probabilistic properties. The presented rough extended model for generalized approximation spaces concerns mainly on data objects that are put in metric spaces. Metric spaces defined the distance function that make possible to compare objects, their similarity, relations, data structure.

In the presented rough extended model the universe with data, is divided into data blocks N_i, features space is divided into feature blocks F_i. For each data block N_i its inclusion $\upsilon(N_i)$ in feature block is calculated. Each data block is described as statistic values $s_i = \{\text{min, avg, max}\}$ that create its bounding block b_i. Image blocks that have feature average value in the F_i are assigned to its lower approximations, each feature block that is contained in the (min, max) bounding block are assigned to its upper approximation.

$$GAS_*(F_i) = \{N_i \in P(U) : \upsilon(N_i, F_i) = 1\} \tag{3}$$

$$GAS^*(F_i) = \{N_i \in P(U) : \upsilon(N_i, F_i) > 0\}. \tag{4}$$

The degree of data blocks and feature blocks inclusion represented by overlap function is defined as

$$\upsilon(N_i, F_i) = \begin{cases} 1 & \text{if } b(F_i) \cap N_i \neq \emptyset \\ 0.5 & \text{if } s(N_i, avg) \in F_i \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

with $s(N_i, avg)$ representing average value for data block N_i.

Detailed description of feature thresholding based upon creation of above lower and upper approximations data blocks, feature blocks and overlap function has been presented in the next section.

3 Feature Thresholding in Generalized Approximation Spaces

In image analysis, segmentation describes partitioning of a digital image into multiple regions—sets of pixels, according to some homogeneity criterion. Image thresholding presents robust segmentation method with feature thresholds forming boundaries of the segments. In feature thresholding algorithm image is divided into image blocks. Each image block is described by its bounding block determined by calculated block min and max values. Each feature block contained in bounding block has upper approximation increased, lower approximation of the feature block that contains average block value is incremented. Putting generalized approximation spaces concepts into feature thresholding setting, the following definitions and descriptions are presented.

The image is denoted as $I = \{x_1, \ldots, x_n\}$ with arbitrary k features, each image pixel is denoted as $x_i = \{a_1, \ldots, a_k\}$. We define image blocks as $N = \{N_1, \ldots, N_m\}$. Feature space is divided into feature blocks $F = \{F_1, \ldots, F_n\}$. Further, for each image block N_i its statistic in the form $s(N_i) = \{\min, \text{avg}, \max\}$ is calculated. Image block bounding block is created $b(N_i)$ represented by Cartesian power $(min, max)^k$ with k—number of features.

$$s(N_i) = \{min, avg, max\}^k \tag{6}$$

$$b(N_i) = (min, max)^k \tag{7}$$

For each image block N_i, all feature blocks F_p that intersect the bounding block $b(N_i)$ are assigned to their upper approximations F_p^* and their measures are increased by 1.0. The feature block F_p that contains bounding block average value of N_i, is considered to contain entirely this image block so its lower approximation F_{*p} and its measure is increased by 1.0.

In general case image blocks may be overlapping or not, have regular or irregular shape. Image features are not constrained to any selected feature class. The feature thresholding algorithm takes as an input image described by at least two features.

Algorithm 1 Feature thresholding algorithm

Input - I - image, S - block creation strategy, T - feature thresholds
Output - F_*, F^* - lower and upper approximations of feature blocks F
Divide image into image blocks $->$ N
Divide feature space into feature blocks $->$ F
foreach *Image block* N_i *in* N **do**
  Calculate block statistics - min, avg, max of $s(N_i)$
  Create feature bounding block $b(N_i)$=$(min, max)^k$
  Increment F_{*p} feature block which contains average value by 1.0
  foreach *Feature block* F_p *contained in* $b(N_i)$ **do**
  Increment F_p^* by 1.0

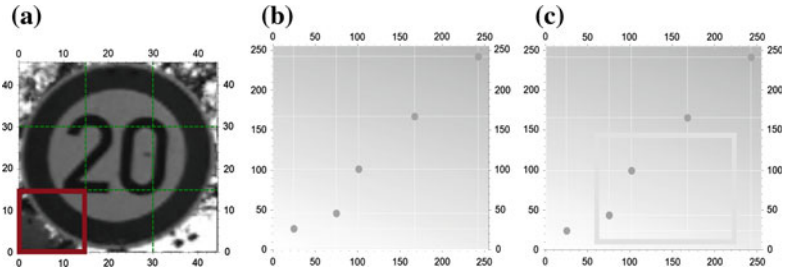


Fig. 1 Road sign of speed limit 20 with image blocks N_i, **b** feature space *red-blue* with feature blocks F_i, **c** feature bounding block—$b(N_i)$

For each feature, the set of feature thresholds should be given. In the preprocessing stage, image is divided into image blocks, feature space is divided into feature blocks. The algorithm performs taking the min, avg, max statistic from each image block and on that base, creating image block bounding block. It is possible to apply other meaningful statistics. Complete feature thresholding algorithm involves the following steps described below.

The feature thresholding algorithm easily scales to 3D or higher dimensional feature sets $\mathrm{A} = \{A_1, A_2, A_3, \ldots, A_k\}$. In case of 3D feature thresholding, additional feature thresholds should be added. Feature thresholds should be carefully selected in order to properly divide image objects into distinctive groups.

In Fig. 1a image of the road sign has been divided into 9 image blocks $N = N_1, \ldots N_9$. In Fig. 1b image feature space with two attributes—red and blue attribute—has been divided into 6×6 feature blocks—$F = F_1, \ldots F_{36}$. Selected image block outlined in red F_7 gives feature block with red outline in Fig. 1c.

4 Experimental Setup and Results

The experimental images consisted of 43 classes of road signs from database [3]. In order to assess the robustness of proposed feature thresholding algorithm, the thresholding has been performed on different sets of features, the algorithm has been tested on solid SVM classification framework.

4.1 Experimental Sets

In the experimental part, road sign data set has been chosen, consisted of 43 German road signs. In this image set, for each image two sets of features have been calculated for testing purposes

1. standard set of features,
2. rough set of features.

Standard set of features consisted from 190 features calculated from RGB images from image dataset. The rough features consisted from 130 features calculated from the same image dataset. Images for all road signs have been resized to 60×60 for calculation of standard features. Totally, 190 standard features are obtained for each image from database with the following standard image features obtained for each image

1. hsv histogram—32 features,
2. autocorrelogram—64 features,
3. color moments—6 features,
4. Gabor filters—gray scale mean amplitude and energy, for 4 scales, 6 orientations, totally 2×24 features,
5. wavelets—40 features, the first 2 moments of wavelet coefficients for gray scale image.

For creation of the rough feature set, the same images as for standard feature have been chosen. The feature thresholding algorithm has been performed for two dimensional data, by selecting only two bands red and blue from the full RGB image bands. Each experimental image with red and blue features selected has been further divided into image blocks as described in Sect. 3. The strategy of image block creation consisted in creation of non overlapping 8×8 image blocks covering all image surface. In case of experimental images sizes 64×64, each image block has 8×8 in size, giving 64 image blocks. For rough features, the following thresholds for red and green band have been selected as presented in Table 1.

The set of two threshold sets for red and blue bands, results in creation of 2D feature blocks, analogous to the feature blocks presented in Fig. 1b. These 2×7 thresholds create 64 feature blocks. Rough feature set 2×64 features—one set for lower approximation and the second for upper approximation blocks has been enhanced by calculating entropy for each block. Totally 130 rough features.

4.2 Experimental Results

Selected images have been divided into training and testing sets,then SVM algorithm has been executed for training phase, next the tested set has been classified by means of SVM learned system in the two categories of standard features and rough features.

Table 1 Feature thresholds for red and blue bands, totally create 64 feature blocks

Feature							
Red band	20	30	50	80	140	180	200
Blue band	20	40	70	95	150	180	210

Table 2 Experimental results for standard features and rough features generated by feature thresholding algorithm and tested on SVM classification framework

Feature	100		
	min	avg	max
Std features	93.44	94.33	95.81
Rough features	93.95	95.25	96.27

The goal of SVM algorithm was to classify correctly images to the proper road sign classes. Given road sign image of speed limit 20, it should be correctly classified as belonging to the class road sign of speed limit 20. The algorithm for each of two feature spaces has been performed 100 times, the best, average and worst classification accuracy from testing phase has been presented in the table. Experimental results have been presented in Table 2

The experimental results show that rough feature set outperforms standard set of features when applied to classification framework of SVM.

5 Conclusions and Further Research

In the paper, new algorithmic approach to feature thresholding scheme has been proposed. The feature thresholding algorithm performs assignment of image blocks to feature blocks acting as data descriptor or image descriptor. This rough feature descriptor can be easily used in classification frameworks such as SVM framework described in experimental section. The algorithm is embedded in the generalized approximation spaces.

Experimental results show that including introduced rough features during image thresholding performs better compared to standard image features. Invented feature thresholding introduces universal solution that can be applied to any numerical features in standard rough, fuzzy and probabilistic setting. At the same time rough feature thresholding incorporates data granularity dependent upon block sizes and feature partition sizes, presents mapping into rough data descriptors by giving image data rough measures. The algorithm presents novelty interoperability between rough-fuzzy and rough probabilistic approximation spaces that will further explored in future research.

References

1. Chan, Y.K., Ho, Y.A., Liu, Y.T., Chen, R.C.: A ROI image retrieval method based on CVAAO. Image Vis. Comput. **26**(11), 1540–1549 (2008)
2. Ghali, N., Abd-Elmonim, W., Hassanien, A.: Object-based image retrieval system using rough set approach. In: Kountchev, R., Nakamatsu, K. (eds.) Advances in reasoning-based image processing intelligent systems, intelligent systems reference library, vol. 29, pp. 315–329. Springer, Berlin (2012)

3. GTSRB: Road signs database. http://benchmark.ini.rub.de/
4. Lingras, P., Butz, C.: Rough set based 1-v-1 and 1-v-r approaches to support vector machine multi-classification. Inf. Sci. **177**(18), 3782–3798 (2007)
5. Malyszko, D., Stepaniuk, J.: Adaptive rough entropy clustering algorithms in image segmentation. Fundam. Inf. **98**(2–3), 199–231 (2010)
6. Malyszko, D., Stepaniuk, J.: Granular multilevel rough entropy thresholding in 2D domain. In: IIS 2008. pp. 151–160. Zakopane, Poland (2008)
7. Malyszko, D., Stepaniuk, J.: Adaptive multilevel rough entropy evolutionary thresholding. Inf. Sci. **180**(7), 1138–1158 (2010)
8. Pal, S.K., Peters, J.: Rough fuzzy image analysis: foundations and methodologies. CRC Press Inc, Boca Raton, FL 33487, USA (2010)
9. Qiang, S., Wangli, C., Qianqing, Q., Guorui, M.: Gaussian kernel-based fuzzy rough set for information fusion of imperfect image. In: ICSP 2014. vol. 284 (2014)
10. Restrepo, M., Cornelis, C.G.J.: Partial order relation for approximation operators in covering based rough sets. Inf. Sci. **284**, 44–59 (2014)
11. Yao, Y., Yao, B.: Covering based rough set approximations. Inf. Sci. **200**, 91–107 (2012)

Progressive Reduction of Meshes with Arbitrary Selected Points

Krzysztof Skabek, Dariusz Pojda and Ryszard Winiarczyk

Abstract The usage of progressive meshes for representation and transmission of triangular meshes of faces at certain level of details was described in the paper. We focused on progressive representation based on Garland and Hoppe approach. The comparison of simple methods using quadric mesh simplification to the view-dependent implementation of progressive methods was performed. The mesh simplification is further improved by introducing the characteristic points fixing the distinguishable points on human face. The tests and discussion of the implemented method were given using face scans from 3dMD face scanner.

Keywords Progressive mesh · Quadric error · Mesh transmission

1 Introduction

There are some ways to make the processing, transmission and presentation of 3D complex mesh objects more efficient. One of the improvements defines several levels of details for the mesh object, e.g. to display the more detailed model when viewer is coming closer. Transmitting a mesh over communication line one may want to see a model with a coarse shape approximation and next increase levels-of-details approximations. Mesh storing is very memory consuming. Such problem may be solved in different ways and one of them is preparing progressive meshes for both mesh simplification or compression.

There are many different ways to represent graphical 3D models, in this article we focused on *progressive meshes* which were introduced by Hoppe [6] and further extended to the view-dependent representation [7].

K. Skabek (✉) · D. Pojda · R. Winiarczyk
Institute of Theoretical and Applied Informatics, PAS, Gliwice, Poland
e-mail: kskabek@iitis.pl

D. Skabek
e-mail: darek.pojda@iitis.pl

R. Winiarczyk
e-mail: ryswin@iitis.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_45

Fig. 1 3dMDFace scanning system—measuring device

2 Data Acquisition

We use 3dMD face scanner [1] to obtain triangular meshes with textures. 3dMDFace scanning system is mainly used in medicine. The device consists of two modules and three cameras are mounted in each module (Fig. 1). The most suitable application for the system is obtaining the surface models of human faces. It is often used in the planning the facial surgery. The resulting model is the 3D triangular surface of the face from ear to ear and the RGB texture. The scanning range is approximately 150 cm and the average objects have about 30 cm diameter.

3 Progressive Representations

Many techniques have been proposed to compress and transmit mesh data. They can be divided into nonprogressive and progressive methods. The first group comprises methods which encode the entire data as a whole. They can either use the interlocking trees (vertex spanning tree and triangle spanning tree) or utilize the breadth-first traversal method to compress meshes. On the other hand there are methods that perform mesh compressing progressively. The solution proposed by Hoppe [6] enables continuous transition from the coarsest to the finest resolution. In such case a hierarchy of level-of-detailed approximation is built. Also the efficient quadric algorithm for mesh decimation was proposed by Hoppe [9] and further extended by Garland [1].

The article [4] shows the recent technique based on zerotree (wavelet) compression as a lossy mesh compression method. Such representation was proposed in part 16: AFX (Animation Framework Extension) of the MPEG-4 international standard for 3D models encoding.

The mesh geometry can be denoted by a tuple (K, V) [9], where K is *a simplicial complex* specifying the connectivity of the mesh simplices (the adjacency of the vertices, edges, and faces), and $V = \{v_1, \ldots, v_m\}$ is the set of vertex positions defining the shape of the mesh in R^3. More precisely, we construct a parametric domain $|K| \subset R^m$ by identifying each vertex of K with a canonical basis vector of R^m, and define the mesh as the image $\phi_v(|K|)$ where $\phi_v : R^m \rightarrow R^3$ is a linear map.

[1] http://www.3dmd.com/category/3dmd-systems/3d-systems/.

Besides the geometric positions and topology of its vertices, the mesh structure has another appearance attributes used to render its surface. These attributes can be associated with faces of the mesh. A common attribute of this type is the material identifier which determines the shader function used in rendering a face of the mesh. Many attributes are often associated with a mesh, including diffuse colour (r, g, b), normal (n_x, n_y, n_z) and texture coordinates (u, v). These attributes specify the local parameters of shader functions defined on the mesh faces. They are associated with vertices of the mesh.

We can further express a mesh as a tuple $M = (K, V, D, S)$, where V specifies its geometry, D is the set of discrete attributes d_f associated with the faces $f = \{j, k, l\} \in K$, and S is the set of scalar attributes $s(v, f)$ associated with the corners (v, f) of K.

As many vertices may be connected in one corner with the same attributes, the intermediate representation called *wedge* was introduced to save the memory [8]. Each vertex of the mesh is partitioned into a set of one or more wedges, and each wedge contains one or more face corners. Finally we can define the mesh structure that contains an array of vertices, an array of wedges, and an array of faces, where faces refer to wedges, and wedges refer to vertices. Face contains indices to vertices, additionally this structure contains array of face neighbours (f_{nei}) in which indices of tree adjacent faces are stored, this information is necessary to build a progressive mesh. There is nothing said in reference papers about order of vertices and indexes of adjacent faces in face structure. In our implementation the counter is stored clockwise and additionally first adjacent face is at first position as first vertex, so if we cross first edge we find the first neighbour, if we cross second we find the second, etc.

In many places of this article we use the word *edge*. The edge is a connected pair of vertices or, in other words, it is a pair of adjacent vertices. There is no additional list of edges, but the first vertex and the face to which this edge belongs are defined instead. Using wedge we can access vertex, even if the adjacent face does not exist we can define edge. Definition of edges is necessary to simplify meshes, to create progressive meshes as well as to determine which edge (vertex) could be collapsed.

3.1 Construction of Progressive Meshes

Progressive mesh (PM) [6] is special case of a mesh or rather an extension of mesh representation, it makes it possible to build mesh for different level-of-details (LOD) [10]. It also allows loading the base mesh M^0, as the mesh of the lowest LOD, and then process the loading of the remaining parts of the mesh structure. As an input source we may use a memory input stream.

In PM form, an arbitrary mesh $\widehat{M}$ is stored as a much coarser mesh M^0 together with a sequence of n detail records that indicate how to incrementally refine M^0 exactly back into the original mesh $\widehat{M} = M^n$. Each of these records stores the information about *a vertex split*, an elementary mesh transformation that adds an additional vertex to the mesh. Thus the PM representation of $\widehat{M}$ defines a continuous sequence

of meshes $M^0, M^1, \ldots M^n$ of increasing accuracy, from which LOD approximations of any desired complexity can be efficiently retrieved. Moreover, smooth visual transitions (*geomorphs*) [6] can be efficiently constructed between any two such meshes. In short, progressive meshes offer an efficient, lossless, continuous-resolution representation. Progressive meshes makes it possible not only to store the geometry of the mesh surface, but, what is more important, preserve its overall appearance, as defined by the discrete and scalar attributes associated with the surface.

There are three operations that make it possible to determine the base mesh of $\widehat{M}$: *edge collapse*, *vertex split* and *edge swap*. Edge collapse operation is sufficient to successfully simplified meshes. Edge collapse operation $ecol(v_s, v_t)$ remove one edge and instead two vertices v_s and v_t insert new one v_s. Additionally two faces (v_t, v_s, v_l) and (v_t, v_r, v_s) are removed. The initial mesh M_0 can be obtained by applying a sequence of n edge collapse operations to $\widehat{M} = M^n$:

$$(\widehat{M} = M^n) \overset{ecol_{n-1}}{\rightarrow} \cdots \overset{ecol_1}{\rightarrow} M^1 \overset{ecol_0}{\rightarrow} M^0$$

Edge collapse operation is invertible. The inverse transformation is called vertex split. Vertex split operation adds in place of vertex v_s two new vertices v_s and v_t and two new faces (v_t, v_s, v_l), (v_t, v_r, v_s) if edge $\{v_s, v_t\}$ is boundary then adds only one face. Because edge collapse transformation is invertible our mesh $\widehat{M}$ can be presented as a simple M^0 and sequence of n vsplits records:

$$M^0 \overset{vsplit_0}{\rightarrow} M^1 \overset{vsplit_1}{\rightarrow} \cdots \overset{vsplit_{n-1}}{\rightarrow} (\widehat{M} = M^n)$$

We call $(M^0, vsplit_0, ..., vsplit_{n-1})$ a progressive mesh (PM) representation of M.

3.2 Quadratic-Based Mesh Reduction

In order to perform the mesh reduction it is necessary to select a sequence of edges to be removed. The problem of the proper choice may be solved in several ways. One of the most efficient methods is based on quadratic error metrics [1].

We understand quadric Q as a symmetric matrix of size 4×4 that holds information about planes of neitghbour faces. The definition is as follows:

$$Q_v = \sum_{p_v} K_p \tag{1}$$

where: p_v is the set of planes containing faces and directly adjacent to the considered vertex v, K_p is the base error quadric stored also in matrix of size 4×4 and used to calculating the square distance from the plane of any point p.

The implementation of the method was described in [13].

3.3 Selection of Characteristic Points in Face Scans

Although, the quadratic-based reduction gives the efficient decimation of the mesh, such reduction of the mesh data often loose the important points on the model surface. Effects of this process are in fact unpredictable especially with regard to potentially crucial points that have no significant importance in quadratic reduction. Such cases occur particularly in the analysis of of scans of human face.

We improve the quality of the mesh simplification using the characteristic points of faces. The scan is treated as a surface model of human face. The markers can be places on such surface using several methods [14]. We considered two of them: anatomical points from traditional anthropometric analysis and points adopted from MPEG4 standard describing its characteristic features [15].

3.4 View-Dependent Progressive Meshes

In the case of progressive mesh, parameters that were sufficient to precisely locate modified area were vertices: v_l, v_r, v_t, v_s. However in the case of view-dependent extension, required vertices are only: v_u, v_t, v_s, but there are additional triangles needed: f_l, f_r, f_{n0}, f_{n1}, f_{n2}, f_{n3}. During refinement operations vertex v_s is replaced by its descendants: v_t, v_u lying between neighboring triangles f_{n0} and f_{n1}, and then face f_l is inserted. Moreover, by analogy between f_{n2} and f_{n3}, face f_r is added.

The detailed description of the implementation of view-dependent progressive meshes is given in [12].

4 Implementation

The implementation of our progressive mesh coding algorithm was fully described by Skabek and Pojda [13]. It is based on the code developed by He Zhao [5]. This method using quadric error metrics provided by Garland [1, 2] and also Hoppe methods for progressive meshes [6, 8].

4.1 Progressive Coding Method

The clue of the approach is selection of vertices for reduction and calculation of the new vertex. For each pair of vertices that are connected with edge, or optionally, for point cloud, that are closer than a given threshold, a quadric error is calculated. The quadric error is a measure of mesh distortion in case of reduction of vertices.

The principles of algorithm are as follow in Skabek and Pojda [13]:

1. for each pair of connected vertices calculate quadric error
2. while number of faces is greater then expected, do:
   (a) find the pair of vertices with smallest quadric error
   (b) calculate coordinates for new vertex
   (c) find all faces containig any vertices belonging to the selected pair, and:
      – remove all faces connected to both vertices
      – for all faces containing only one vertex found for pair, change it to the new vertex
   (d) recalculate quadric error for all pairs of vertices containing the new vertex

The support for meshes with texture was implemented too. The texturing method is based on a simple texture mapping, where the structure of texture, its color and brightness are not analyzed.

A quality of representation of texture on mesh which was coded with our method is worse than described in publications by Hoppe [11] and Garland [3]. This especially applies to strongly reduced meshes.

4.2 Fixed Points Extension

Using the above algorithm, the only criterion for selecting the edges to reduce is the value of the error. The edges are removed in the order to maintain the best quality of the mesh. However, it is a measure of the quality of the entire surface. In some cases, we want you to get better precision for selected surface areas at the expense of other less important parts of it.

We modified the algorithm in such a way that it is possible to define a set of vertices that will not modified during encoding. To this end, we implemented a new version of the procedure for appointing the new vertex and calculating quadric error.

First, check to see if any of the vertices of the edge is in the set of fixed points. If so, then instead of to calculate coordinates for the new vertex, we arbitrarily rewrite them from the fixed point. So even if the selected edge is reduced, it will be replaced by a fixed vertex.

A special case occurs when the two vertices of the edge are in the set of fixed points. Then we recognize that this edge should not be removed. For this purpose, we define the error value for the edge as equal max float.

We have implemented a mechanism for easy selection of points on the mesh. The set of points can be then saved into a file for the future use. We developed software for visualization and processing of three-dimensional surface. It also allows the encoding and decoding of progressive meshes stored in a format developed by us.

5 Comparison

As we described earlier, we have implemented an algorithm for progressive coding of 3d meshes (Skabek and Pojda [13]). In original it is an modification of He Zhao's [5] code with corected performance of coding. We have introduced further modifications into it, so that we can identify a number of points (vertices) on the surface. These vertices are treated during the processing of the grid as a permanent and can not be removed or transformed.

We expected that this modified algorithm will achieve a significant improvement in the quality of the resulting meshes in places of special interest. For the tests, we chose several meshes obtained by scanning a human faces. Figure 2 shows three of them. On each surface we pointed 78 points which are placed around the eyes, nose and mouth as well as to mark the outline of the face. We selected a subset of points describing the face defined in MPEG4. Meshes with selected points is shown on Fig. 3.

Each of the selected meshes we reduced using our algorithm in order to obtain a base mesh size of 1000, 200 and 100 vertices. The reduction is moved around by using the original algorithm, without specifying the fixed points, as well as using the modified algorithm with defined 78 points. Results for a example mesh are shown on Fig. 4.

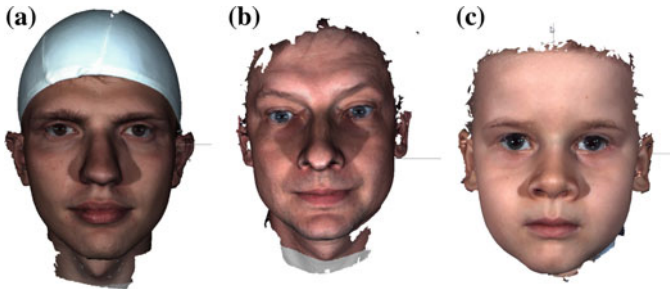


Fig. 2 Meshes used for tests. **a** Face 1, **b** Face 2, **c** Face 3

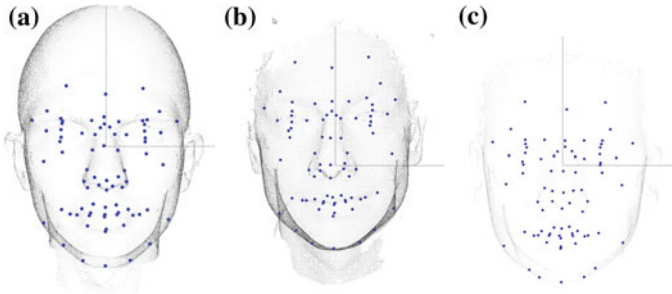


Fig. 3 Meshes with set of fixed points. **a** Face 1, **b** Face 2, **c** Face 3

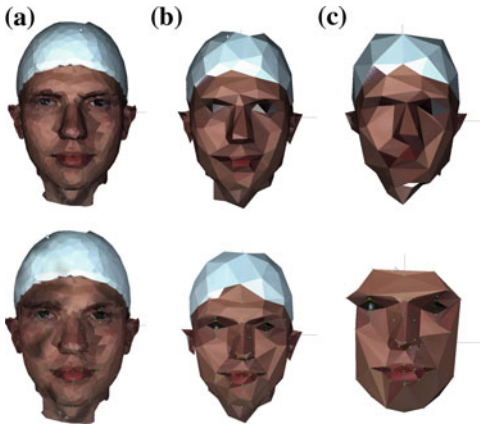


Fig. 4 Comparition of automatic decimation without (*top*) and with selected points (bottom). **a** 1000 vertices, **b** 200 vertices, **c** 100 vertices

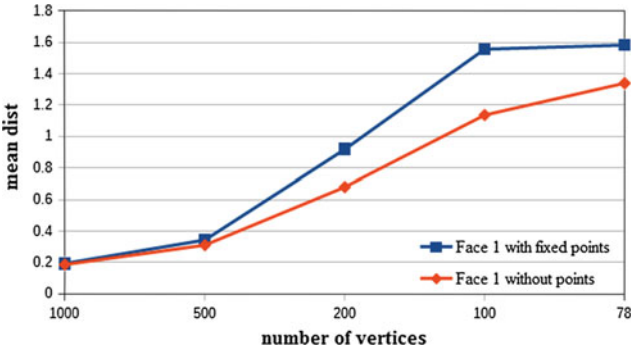


Fig. 5 The mean distance depending on the number of vertices

As can be seen, the more reduced the mesh, the more the selection of fixed points affects the quality of the resulting base mesh. The areas around which fixed points are marked, are less distorted and it is possible to recognize the parts of the face.

It can be seen in Fig. 5 that the mean distance of the reduced mesh to the original is worse when the fixed points was defined than for the unmodified quadric method. This situation results directly from disruption of the quadric algorithm which gives good approximation of the reduced mesh. However, we obtained the better fitting for the selected points and finally we improved the recognizability of the reduced mesh.

6 Summary

Comparing the above described approaches we can point their advantages and drawbacks. The progressive method is a good solution to reduce the amount of mesh data for transmission or rough rendering. It is possible to choose the appropriate

level-of-detail (LOD). We can preserve the characteristic parts of surface model by selecting the certain points to improve the accuracy surface similarity. We treat the progressive method as a kind of mesh compression, however, the compression rate is meaningful considering highly decimated meshes. When we need to make the mesh dense, the amount of the specifying data may exceed the original mesh. In case of view-dependent progressive meshes the situation is similar. We use additional data records to localize reduction and this way it is possible to specify the topologically consistent partial area of the mesh. When we aim to reconstruct the whole dense surface the amount of adjusting data is significantly greater even than in case of classical progressive meshes.

Acknowledgments This work has been partially supported by the National Science for Research and Development project INNOTECH-K2/IN2/50/182645/NCBR/12.

References

1. Garland, M.: Quadric-Based Polygonal Surface Simpling. Ph.D. thesis, School of Computer Science Carnegie Mellon University, Pittsburgh (1999)
2. Garland, M., Heckbert, S., P.: Surface simplification using quadric error metrics. In: SIGGRAPH, pp. 209–216. Los Angeles, USA (1997)
3. Garland, M., Heckbert, P.S.: Simplifying surfaces with color and texture using quadric error metrics. In: VIS 1998, pp. 263–269 (1998)
4. Giola, P., Aubault, O., Bouville, C.: Real-time reconstruction of wavelet-encoded meshes for view-dependent transmission and visualization. IEEE Trans. Circ. Syst. Video Technol. **14**(7), 1009–1020 (2004)
5. He, Z.: A surface simplification software, http://hezhao.net/projects/progressive-meshes
6. Hoppe, H.: Progressive meshes. Comput. Graphics **30**, 99–108 (1996)
7. Hoppe, H.: View-dependent refinement of progressive meshes. In: SIGGRAPH, pp. 189–198. Los Angeles, USA (1997)
8. Hoppe, H.: Efficient implementation of progressive meshes, MSR-TR-98-02. Techincal report, Microsoft Research (1998)
9. Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., Stuetzle, W.: Mesh optimization. In: SIGGRAPH, pp. 19–26. Anaheim, USA (1993)
10. Luebke, D., Reddy, M., Cohen, J., Varshney, A., Watson, B., Huebner, R.: Level of Detail for 3D Graphics. Morgan Kaufmann Publishers, San Francisco (2003)
11. Sander, P., Snyder, J., Gortler, S., Hoppe, H.: Texture mapping progressive meshes. In: SIGGRAPH, pp. 409–416. Los Angeles, USA (2001)
12. Skabek, K., Francki, M., Winiarczyk, R.: Implementation of the view-dependent progressive meshes for virtual museum. In: HSI, pp. 331–336. Rzeszow, Poland (2010)
13. Skabek, K., Pojda, D.: Optimization of mesh representation for progressive transmission. Theor. Appl. Inf. **23**(3–4), 263–275 (2011)
14. Tomaka, A.A.: Analiza obrazow wielomodalnych dla potrzeb nieinwazyjnej diagnostyki ortodontycznej. IITIS PAN, Gliwice (2013)
15. Watkinson, J.: The MPEG Handbook. Focal Press/Elsevier, Burlington (2004)

The Class Imbalance Problem in Construction of Training Datasets for Authorship Attribution

Urszula Stańczyk

Abstract The paper presents research on class imbalance in the context of construction of training sets for authorship recognition. In experiments the sets are artificially imbalanced, then balanced by under-sampling and over-sampling. The prepared sets are used in learning of two predictors: connectionist and rule-based, and their performance observed. The tests show that for artificial neural networks in several cases the predictive accuracy is not degraded but in fact improved, while one rule classifier is highly sensitive to class balance as it never performs better than for the original balanced set and in many cases worse.

Keywords Class imbalance · Sampling strategy · Authorship attribution

1 Introduction

The class imbalance occurs when the numbers of samples representing considered classes are sufficiently different. The class with significantly fewer objects is called *minority*, and the class with many more instances *majority*. Classifiers tend to show bias for majority classes as more information about their objects is available, while for minority classes less knowledge often means worse recognition [5]. The imbalance can be caused by uneven availability of instances, cases of multi-class recognition, or a combination of those and other reasons.

When data is imbalanced, we can either try to use it anyway but with some modified learning approaches [4], which can assign higher significance to objects from minority classes, or we can employ some strategy of re-sampling that leads to obtaining artificially balanced sets [8]. Re-sampling is performed in two directions. Under-sampling causes reduction of instances from majority classes to reach balance, while by over-sampling the objects from minority classes are multiplied by either simple repetition or with introducing some small variations [3]. Typically under-sampling works better than over-sampling, however, the results depend also

U. Stańczyk (✉)
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: urszula.stanczyk@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_46

on the sensitivity to the class imbalance of a particular predictor employed, and the underlying meaning in the application domain.

In the research described two distinctively different classifiers were used, an artificial neural network with Multi-layer Perceptron (MLP) topology, which is characterised by good generalisation and adaptivity properties [1], and one rule (OneR) classifier that in its rule induction algorithm exploits the fact that short decision rules often possess better descriptive capabilities than provided by detailed definitions of patterns given through conditions in longer rules [6].

As the application domain the stylometric analysis of texts was considered, with its task of authorship attribution [7]. To recognise authorship a definition of a writing style is formulated either from the linguistic point of view, but most often from the data mining perspective [10]. To find such definition, some sufficient number of representative text samples is required. When the samples are not representative due to limited lengths or numbers, the recognition can be unreliable even for balanced datasets [9], and it is worse for imbalanced classes.

The paper is organised as follows. Section 2 presents machine learning approaches used, and comments on the class imbalance problem and its meaning in stylometry. In Sect. 3 there is explained the structure of an original balanced training set, construction of imbalanced sets, balancing these sets by under- and over-sampling, and there are given test results. Section 4 concludes the paper.

2 Background

The research described in this paper was focused on the class imbalance considered in the context of construction of training sets for the authorship attribution task, for two types of classification systems employed in data mining.

2.1 Classification Systems

Classification tasks require predictors providing enhanced understanding of learnt knowledge, and structures underlying the input data on one hand, yet capable of adaptation and generalisation on the other. The former can be given by rule classifiers, while the latter by connectionist approach of artificial neural networks.

A rule induction algorithm can return the complete set of rules to be found for a training set, but it can also limit considerations by focusing on rule parameters [12]. *OneR* classifier [6] bases on the fact that short rules often result in good recognition. The rule induction process consists of two phases. Firstly, for each of the considered attributes the candidate rules are constructed for all attribute values and the classes to which they most frequently classify. The quality of inferred rules is evaluated by the resulting classification accuracies (for the training set). From this set of rules the best one is chosen. In the second phase of the procedure from all these selected best

rules again a single one is taken as 1-rule, with the lowest error. As OneR classifier in its algorithm calculates frequencies for classes, it is highly sensitive to the class imbalance problem.

Multi-layer Perceptron (MLP) is a feed-forward artificial neural network widely employed because of its good adaptive and generalisation properties [11]. Back-propagation learning rule enables to adjust weights associated with interconnections between neurons organised into layers, which leads to minimisation of the error on the network output, calculated as the sum of differences between the desired and obtained values for all training facts. The class imbalance present in the training set can cause favouring of the majority class.

The two approaches to data mining can be treated as opposites: OneR classifier providing a decision algorithm consisting of short rules explicitly listing conditions on features leading to specific classes, MLP with learnt knowledge hidden and inseparable from the internal structure. This contrasting behaviour was the reason for using the two classifiers in the research on class imbalance.

2.2 The Class Imbalance Problem

To learn characteristics of data a classification system needs access to representative samples, representative in information they bring just by themselves, by describing patterns for the class they belong to, but valid in presence of objects from other classes, by the knowledge load differentiating among those classes [5].

In case of even distribution of samples for all considered classes, the quality of results is not influenced by over- or under-representation of some class. When the differences in numbers of objects in classes are distinctive, the trained classifier can show bias and better recognise some class, simply because it has learnt more about this class [4]. To prevent such situations it is best to keep representation for all classes at the same level, and obtain more samples for under-represented classes, or discard some from over-represented classes. When more instances are not possible to access, and rejecting some objects would cause the training sets to be too small, we can construct balanced sets basing on those imbalanced by two opposite re-sampling approaches, under-sampling and over-sampling.

In under-sampling the cardinalities of sets of objects belonging to classes are made even by brining them down to the one with fewest instances. The advantage of this approach is that we can reasonably expect that for remaining samples the distribution of patterns should be the same as in the set with more instances. The disadvantage is we disregard information carried with rejected samples, which can negatively affect the performance. Also, with less input data the learning stage can run into problems and result in unreliable observations.

Over-sampling increases the numbers of objects in under-represented classes. It can be done by simple repetition, and then some instances appear in the set more than once, which reinforces information brought by them. Yet, such repeated objects can be disregarded by the learning algorithm as nothing new can be learnt from

them. In the SMOTE approach [3] small numbers are added to values in the repeated instances. The drawback of this solution is introducing synthetic artificial objects, not necessarily appearing in real life data.

2.3 The Class Imbalance in Stylometry

Stylometry is a a study of writing styles and for its task of authorship attribution linguistic styles are defined [9]. By application of feature selection or ranking approaches [13] there are found features that capture characteristics unique for writers [10]. In analysis there are employed statistics-oriented computations [2] or machine learning approaches [12], referring to frequencies of usage for lexical and syntactic markers [7]. For reliable recognition features need to be calculated over text samples of sufficient length, and providing information on style variations.

In stylometry the class imbalance can be a case of recognition of an author against many. For binary authorship recognition one writer can simply produce more manuscripts over longer periods of time, thus their styles show more variations in samples, while for another author there are few works with more consistent style. Also pieces of writing can significantly vary in length (for example short stories vs. novels), giving base to different numbers of produced samples.

When under-sampling is employed to handle the class imbalance problem within stylometric analysis [8], a balanced set can be constructed not only by rejecting some samples from the majority class. An alternative way is to reconstruct the text samples from which features are extracted, that is to build fewer bigger text parts. However, then their lengths cease to be comparable with those in the minority class, which puts calculated characteristics to question, and can result in disregarding slight variations in style visible in smaller text samples.

In over-sampling minority instances are repeated as they are or with modifications, or text samples are divided to create more smaller parts, and features are re-calculated. When examples are repeated without any change it corresponds to a theoretically possible yet unlikely situation of finding new samples with exactly the same characteristics, which reinforces their meaning. Adding small values to samples means constructing artificial samples that are not necessarily close to those extracted from real text when considered together as a pattern. On the other hand, division of text samples into smaller units can result in producing such short texts that their characteristics are no longer representative.

3 Experiments

In the described research firstly balanced training sets were prepared for binary authorship recognition. Basing on them several artificially imbalanced sets were constructed, and then they were balanced again by under- and over-sampling. For all sets the performance for two classifiers was observed, MLP and OneR.

3.1 Construction of Input Datasets

The input datasets with balanced classes were constructed for recognition between two authors, E. Wharton and J. Austen. The selected novels were divided into parts, and for them 25 lexical and syntactic features extracted, giving occurrence frequencies of selected function words and punctuation marks. As it is likely that characteristics calculated for parts of one text are similar, random distribution of samples to learning and testing sets could give falsely high predictions. To avoid it, groups of samples corresponded to separate works, for training 4 titles with 25 samples, and 3 titles with 15 samples per author for testing.

For this original balanced training dataset for both predictors used the classification accuracies were close, 90.00 and 88.89 % respectively for MLP and OneR, which was used as a point of reference in comparisons.

Basing on the original training set the artificially imbalanced sets were prepared using three ways of sampling: random, proportional from all documents, and proportional but from limited documents, for one author treated as the minority class and the other as the majority class, and then reversing the case.

With random approach to sampling the origin of the sample was completely disregarded, and for some of the works the numbers of samples were higher while for others lower. In the second approach the samples were selected in such way as to provide the same lower representation for each document, while still considering all documents. And in the third strategy both the numbers of documents and samples were limited, to one novel, two novels, or three novels. In this case the sets corresponding to all possible combinations of considered documents were prepared. The structures of sets are shown in Fig. 1 on the left.

Each of the produced imbalanced training sets was next transformed into balanced by two approaches, under-sampling and over-sampling. In under-sampling approach the majority class was treated in the same way as minority, thus some group of samples was removed in the same manner as was used for the minority class. In over-sampling simple repetition of minority samples occurred, reflecting their structure, as shown in Fig. 1 on the right. For cases where limited numbers of documents were used, all possible combinations of them were constructed.

3.2 Performance for Imbalanced Training Sets

The results from tests for the training sets imbalanced by random selection of samples, regardless of the documents they were based on, and for sets constructed by proportional selection of samples from all base documents are displayed in Fig. 2, and for sets prepared by selection of samples from reduced number of base documents in Table 1. In all approaches numbers of samples were varied (as illustrated in Fig. 1) and both classes were tested as minority.

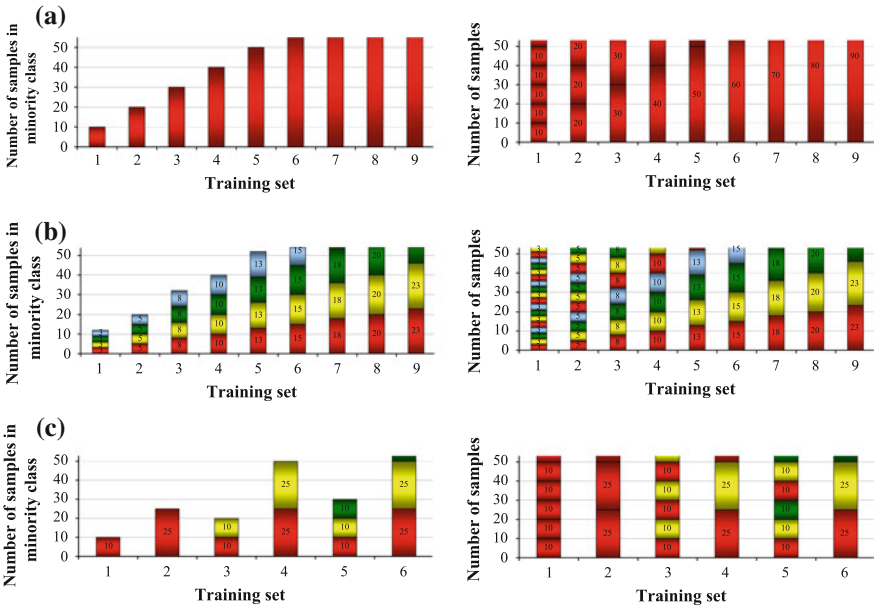


Fig. 1 Construction of training sets from the original balanced set. On the *left* imbalanced training sets produced by three strategies: **a** random selection of samples, **b** proportional selection from all documents, **c** proportional selection from fewer documents; on the *right* structures for balanced sets produced by over-sampling

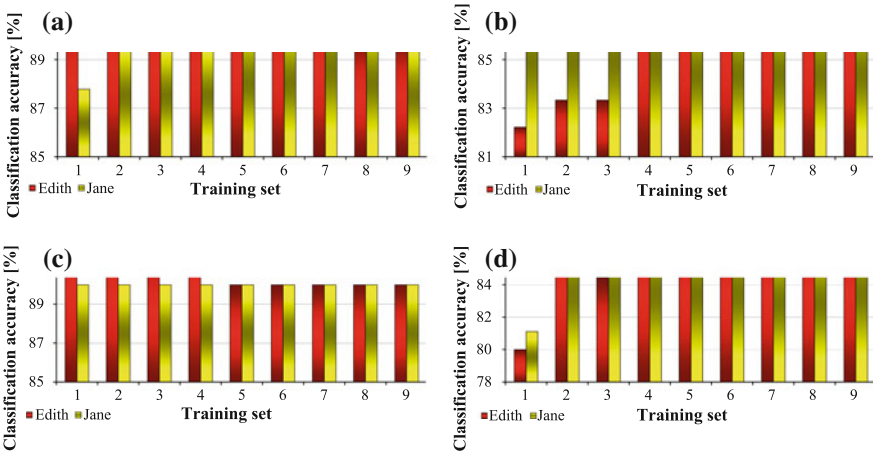


Fig. 2 Performance for training sets imbalanced by: **a** and **b** random selection of samples, **c** and **d** proportional selection of samples from all documents; **a** and **c** for MLP classifier, **b** and **d** for OneR classifier. Series indicate the minority class

Table 1 Performance for training sets that were imbalanced by proportional selection of samples from reduced number of documents

	MLP classifier						OneR classifier					
	Training set						Training set					
	1	2	3	4	5	6	1	2	3	4	5	6
	Edith as minority class											
Avg	89.73	90.28	94.26	93.33	93.89	92.22	74.45	83.33	76.67	86.30	87.50	88.89
Min	71.11	75.56	92.22	90.00	92.22	90.00	68.89	78.89	68.89	78.89	84.44	88.89
Max	96.67	96.67	96.67	96.67	95.56	95.56	80.00	88.89	88.89	88.89	88.89	88.89
	Jane as minority class											
Avg	87.50	89.45	89.82	90.56	90.00	90.00	76.11	82.78	88.34	88.34	88.61	88.61
Min	78.89	86.67	88.89	90.00	90.00	90.00	61.11	70.00	87.78	87.78	87.78	87.78
Max	91.11	92.22	90.00	91.11	90.00	90.00	87.78	87.78	88.89	88.89	88.89	88.89

The first and most obvious observation is that, as expected, OneR classifier betrays higher sensitivity to all types of class imbalance than ANNs, as it never performs better than for the original balanced set and in several cases significantly worse, while MLP shows some decreased recognition but improved as well. Also, MLP favours Edith as minority class over Jane, with better predictive accuracies for all imbalancing strategies. For rule classifier the bias to classes is visible when the disproportion between minority and majority is 1:10, 2:5, and 3:10 for random selection of samples, as then for Edith class as minority the recognition is significantly lower than for Jane. In other cases there are no overall big differences in results when the minority class is changed into majority.

When the training sets were based on fewer text documents all combinations of them were studied, thus results given in Table 1 include the minimal, maximal and average classification accuracies. The differences between them clearly indicate that samples from some documents provide better knowledge base than others, especially for Edith as minority class, as there are cases of noticeable improvement for ANNs. It also means that samples from Jane class are more difficult to recognise. For OneR classifier for both classes the performance is close to that for the original balanced training set, except for cases when the ratio of disproportion between numbers of samples from minority and majority classes is of the rank 1:10, 1:4, or 1:5, where the predictive accuracy is decreased.

3.3 Under-Sampling

Under-sampling was achieved by discarding samples from the majority class with the same strategy as used to construct minority, and the classification results for random and proportional selection of samples from all works are given in Fig. 3.

ANNs obtain at least the same, but often improved recognition when compared with the original balanced training set, while for rule classifier the accuracy is degraded for all but one set with random selection of samples, and close to a half of sets with proportional selection. While comparing imbalancing re-sampling strategies, for both MLP and OneR random selection of samples leads to slightly better recognition results than proportional selection.

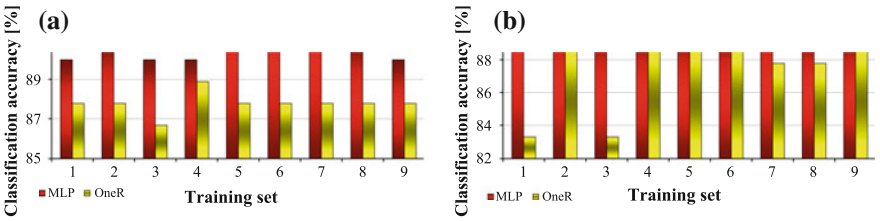


Fig. 3 Performance for training sets balanced by under-sampling with: **a** random selection of samples, **b** proportional selection from all documents

For the third imbalancing strategy, with considering fewer than available documents, again it is possible to prepare several combinations, hence the average, worst, and best performance listed in Table 2. The overall observation for the rule classifier is that the average classification accuracy is lower than for imbalanced training sets. For neural networks for the second training set, containing 25 samples based on a single document per each author, the curious case of at the same time the worst and the best performance is detected, depending on the particular combination of chosen documents. Yet the average in this case is below the reference point of recognition for the original balanced training set.

3.4 Over-Sampling

Over-sampling for training sets was obtained by simple repetition of samples for minority classes, without any modifications of these samples. The classification results for re-balanced sets based on random selection of samples, and for proportional selection from all considered documents are given in Fig. 4.

Both classifiers are worse at recognition of objects from Jane class. For ANNs in fact for Edith as the minority class the performance is improved for almost all sets for random, and for 4 out of 9 sets for proportional selection of samples. ANNs predict at the similar level when learning from sets based on random and proportional selection of samples, and rule classifiers work better for proportional selection. This performance is very close to the original balanced training set, except for sets basing on minority cases when the disproportion was 1:10 (thus in over-sampling 10 samples were multiplied 10 times to even the numbers of objects), where the predictive accuracy is distinctively lower.

The test results for over-sampled training sets based on proportional selection of samples from fewer documents listed in Table 3 show that the performance for both classifiers is similar to the one observed previously for sets imbalanced with this strategy. ANNs show favour for Edith class while OneR for Jane.

When the averaged test results for all training sets for the two classifiers are compared, the conclusion is that for both under-sampling works only slightly better than over-sampling. In fact both re-sampling strategies give similar predictive accuracies to those for imbalanced training sets. For both classifiers using fewer base texts caused lower averaged accuracy than other approaches, but for MLP also cases of significant improvement. MLP always perform better when Edith is the minority class, and for OneR no such strong tendency was detected.

Table 2 Performance for training sets imbalanced by proportional selection of samples from fewer documents with under-sampling as strategy to make them balanced

	MLP classifier						OneR classifier					
	Training set						Training set					
	1	2	3	4	5	6	1	2	3	4	5	6
Avg	87.01	87.71	92.28	91.67	92.01	91.25	75.35	79.93	78.80	83.24	83.13	87.50
Min	73.33	65.56	90.00	88.89	90.00	88.89	30.00	65.56	60.00	50.00	60.00	71.11
Max	94.44	97.78	95.56	96.67	94.44	94.44	87.78	88.89	88.89	88.89	88.89	88.89

Table 3 Performance for training sets that were imbalanced by proportional selection of samples from reduced numbers of documents made as balanced by over-sampling

	MLP classifier						OneR classifier					
	Training set						Training set					
	1	2	3	4	5	6	1	2	3	4	5	6
	Edith as minority class											
Avg	88.89	90.56	93.89	92.59	93.89	92.22	65.83	82.22	81.85	86.30	88.89	88.89
Min	71.11	77.78	92.22	90.00	92.22	90.00	48.89	78.89	70.00	78.89	88.89	88.89
Max	95.56	96.67	95.56	94.44	95.56	95.56	83.33	88.89	88.89	88.89	88.89	88.89
	Jane as minority class											
Avg	86.95	88.89	89.82	90.37	90.00	90.00	78.34	86.39	85.19	87.78	86.95	88.61
Min	77.78	84.44	88.89	88.89	90.00	90.00	67.78	84.44	76.67	85.56	85.56	87.78
Max	91.11	92.22	90.00	91.11	90.00	90.00	88.89	87.78	88.89	88.89	88.89	88.89

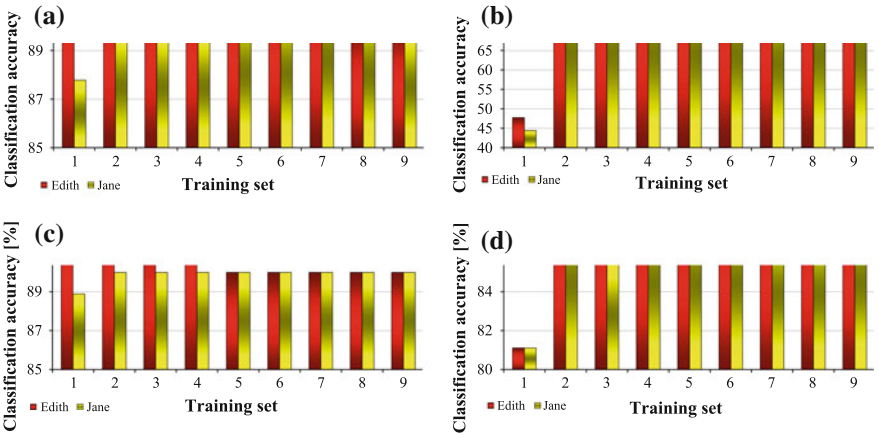


Fig. 4 Performance for training sets balanced by over-sampling. Series indicate the class which was made as minority by: **a** and **b** random selection of samples, **c** and **d** proportional selection of samples from all considered documents; **a** and **c** for MLP classifier, **b** and **d** for OneR classifier

4 Conclusions

The paper presents research on construction of training datasets for the task of binary authorship attribution in cases of imbalanced classes. The imbalance is caused artificially in three ways: by random selection of samples, by proportional selection of samples from all documents on which samples are based, and by using fewer base documents. For all constructed training sets the performance is studied for two types of predictors, artificial neural networks and one rule classifier. The performance is observed for imbalanced sets and then for sets with re-sampled instances, with application of under- and over-sampling.

The experiments show that due to good generalisation properties ANNs achieve even increased accuracy for both imbalanced and re-sampled training sets, while one rule classifier is highly sensitive to class balance and at most keeps the prediction ratio of the original balanced set, in many cases, however, the performance is severely degraded.

Acknowledgments The research described was performed within the project BK/RAu2/2015 at the Institute of Informatics, Silesian University of Technology, Gliwice, Poland.

References

1. Alejo, R., Sotoca, J., Valdovinos, R., Casañ, G.: The Multi-Class Imbalance Problem: Cost Functions with Modular and Non-Modular Neural Networks. In: Wang, H., Shen, Y., Huang, T., Zeng, Z. (eds.) The 6th international symposium on neural networks. AISC, vol. 56, pp. 421–431. Springer, Berlin (2009)

2. Baron, G.: Influence of data discretization on efficiency of Bayesian classifier for authorship attribution. Procedia Comput. Sci. **35**, 1112–1121 (2014)
3. Chawla, N., Bowyer, K., Hall, L., Kegelmeyer, W.: SMOTE: synthetic minority over-sampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
4. Grzymała-Busse, J., Stefanowski, J., Wilk, S.: A Comparison of Two Approaches to Data Mining from Imbalanced Data. In: Negoita, M., Howlett, R., Jain, L. (eds.) Knowledge-based intelligent information and engineering systems. LNCS, vol. 3213, pp. 757–763. Springer, Berlin (2004)
5. He, H., Garcia, E.: Learning from imbalanced data. IEEE Trans. Knowl. Data Eng. **21**(9), 1263–1284 (2009)
6. Holte, R.: Very simple classification rules perform well on most commonly used datasets. Mach. Learn. **11**, 63–91 (1993)
7. Jockers, M., Witten, D.: A comparative study of machine learning methods for authorship attribution. Literary Linguist. Comput. **25**(2), 215–223 (2010)
8. Stamatatos, E.: Author identification: Using text sampling to handle the class imbalance problem. Inf. Process. Manage. **44**, 790–799 (2008)
9. Stamatatos, E.: A survey of modern authorship attribution methods. J. Am. Soc. Inf. Sci. Technol. **60**(3), 538–556 (2009)
10. Stańczyk, U.: Dominance-Based Rough Set Approach Employed in Search of Authorial Invariants. In: Kurzyński, M., Woźniak, M. (eds.) Computer recognition systems 3. AISC, vol. 57, pp. 315–323. Springer, Berlin (2009)
11. Stańczyk, U.: Application of DRSA-ANN Classifier in Computational Stylistics. In: Kryszkiewicz, M., Rybiński, H., Skowron, A., Raś, Z. (eds.) Foundations of intelligent systems. LNAI, vol. 6804, pp. 695–704. Springer, Berlin (2011)
12. Stańczyk, U.: Rule-based approach to computational stylistics. In: Bouvry, P., Kłopotek, M., Marciniak, M., Mykowiecka, A., Rybiński, H. (eds.) Security and intelligent information systems. LNCS, vol. 7053, pp. 168–179. Springer, Berlin (2012)
13. Stańczyk, U.: Ranking of characteristic features in combined wrapper approaches to selection. Neural Comput. Appl. **26**(2), 329–344 (2015)

Part IX
Fuzzy Systems

Approximate Reasoning and Fuzzy Evaluation in Code Compliance Checking

Ewa Grabska, Andrzej Łachwa and Grażyna Ślusarczyk

Abstract This paper deals with the problem of developing intelligent tools which would support the design system by automated checking various design criteria. These criteria contain both building codes and customer requirements. The paper extends the logic-based reasoning methods used previously in computer-aided visual design to a fuzzy classification. The fuzzy classification of the drawings representing early design solutions is based on approximate reasoning, where different types of criteria are considered. Due to this classification the system is able to evaluate the correspondence of potential solutions to the project specification and check their code compliance. The approach is illustrated by example of designing layouts of a small office.

Keywords Visual design system · Building code · Fuzzy classification · Approximate reasoning

1 Introduction

Architectural design solutions of buildings are contemporary created with the use of CAD tools. Majority of them are presented both in the form of drawings understandable to users and internal representations for automated processing. Nowadays CAD systems offer substantial support for analyzing and evaluating drawings during the spiral design process, in which requirements are articulated by the visualization of early design solutions.

E. Grabska · A. Łachwa · G. Ślusarczyk (✉)
The Faculty of Physics, Astronomy and Applied Computer Science,
Jagiellonian University, Kraków, Poland
e-mail: gslusarc@uj.edu.pl

E. Grabska
e-mail: ewa.grabska@uj.edu.pl

A. Łachwa
e-mail: andrzej.lachwa@uj.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_47

Popular vector formats of drawings generally are not suitable for automated analysis of design functionality and automated evaluation. An example of the suitable representation was considered in [6], where internal representations of drawings in the form of hierarchical hypergraphs consist of atoms corresponding to functional components of buildings, such as for instance: a sanitary, reception, hall and staircase. This representation makes it possible to visualize early design solutions in the form of drawings generated by the designer on the monitor screen and to gather design knowledge on which reasoning about designs is based.

Early design solutions are generated on the basis of the designer's knowledge and design specification determined in collaboration with the customer. Usually, the customer has his own vision of the designed building but his requirements and constraints are soft and often conflicting. The criteria can describe simple components, like *the bathroom should be big*, and groups of components, like *the area of the layout should be less than* 100 m^2. Other criteria are of a relative type, like *the sleeping part of the apartment should have the similar area as the rest of the apartment*. The important aspect of the design process is that the criteria evolve during this process. Moreover, most design problems involve several conflicting criteria that the designer tries to satisfy simultaneously. It is often impossible to fulfill all requirements and constraints at the same time. The number of criteria can be decreased by combining simple criteria into more complex ones and creating in this way a hierarchical structure of criteria [1].

Apart from several criteria imposed by the designer, the compliance checking of early design solutions against applicable building codes has to be done. The research in the area of compliance checking has mostly focused on representations of building codes in computational formats [2, 9]. Methods of converting feasible regulation clauses into machine understandable rules are presented in [7]. Many code rules are however only semi-formalizable as they contain fuzzy concepts, like *a space is to be accessible easily*. During the design process it is difficult to control all essential features of the design problem at hand. Therefore tools which support the automated checking of important criteria are still needed.

This paper is an attempt of extending the logic-based reasoning methods used in the computer-aided visual design system discussed in [4, 6]. In this system the early design solutions in the form of drawings generated by the designer are internally represented as hierarchical hypergraphs. However the first-order logic-based reasoning used so far to check the validity of solutions in respect to the given criteria is too strict to evaluate early solutions, where not all requirements are exactly specified. To improve the evaluation method the fuzzy interpretation of instance hypergraphs was introduced in [8]. Due to this interpretation, where crisp values of instance hypergraph attributes are compared with their corresponding crisp or fuzzy values in the specification, the system is able to evaluate the solution.

Considering the fuzzy character of several building code rules and various design requirements, the approximate reasoning about drawings representing early design solutions, which leads to their fuzzy evaluation, is performed. In approximate reasoning the linguistic variables and fuzzy processing are used to compute in which degree the design elements satisfy the required criteria. The obtained results enable

the system to evaluate the correspondence of potential solutions to the project specification and check their code compliance. The final evaluation of generated solutions allows to graduate them according to the degree of particular criteria fulfillment.

2 Visual Layout Language

The problem of using visual languages to support the conceptual stage of the design process was considered in [4] on examples of designing multi-storey buildings with the use of the computer system [5, 10]. This system supports both creating and visualizing solutions, and evaluating their validity in respect to the given criteria. In the system two visual languages for creating design solutions by the designer were used. The first one allows the designer to design 2D floor layouts, while the second one is dedicated to creating 3D building structures.

In this paper in order to explain the proposed method of approximate reasoning about designs and their fuzzy evaluation in a clear way, we concentrate only on generating 2D floor layouts. In the proposed interactive design system the user visualizes his/her ideas related to floor layouts by means of a graphical editor. The editor of floor layouts uses rules of the problem-oriented visual language which are based on a notion of a conceptualization specifying concepts that are assumed to exist in a given design domain and relationships that hold among them [3]. Drawings representing layouts are automatically transformed by the system into their internal computer representations in the form of hierarchical hypergraphs.

The visual language, which enables the designer to create and edit 2D floor layouts is called a layout language [4] and is denoted by L_{layout}. A vocabulary of L_{layout} is composed of geometric primitives corresponding to components like areas, rooms, walls, stairs, doors and windows, while the rules specifying possible arrangements of these components are determined by the syntactic knowledge. Elements of L_{layout} are drawings representing simplified architectural projects. Drawing an initial solution the designer places polygons representing components like functional areas or rooms of a floor layout in an orthogonal grid. The accessibility relation among rooms is represented by primitives corresponding to doors or incomplete walls between rooms. The adjacency relation between rooms is represented by primitives corresponding to walls.

The first design solution is generated by the designer on the basis of the design specification determined in collaboration with the customer. In this specification the required areas, rooms, their functions and purpose, and the rules of their arrangement are fixed. Shapes of rooms, their sizes together with membership functions for fuzzy criteria are also specified. Many requirements and some constraints determined in the specification are soft, i.e., it is assumed that they will be met only to a certain degree. Moreover, many other criteria are of a relative type. Apart from soft requirements and constraints, which are to be fulfilled in some degree, and hard ones, which are to be fully satisfied, there are also sharp requirements and constraints, the fulfillment of which would be desirable but is not absolutely necessary. As most design prob-

lems involve several conflicting criteria, the weights determining importance of the specified requirements and constraints should be determined. Thus, even incoherent and inconsistent specification can be compatible with the specific project.

The design requirements are stored in checklists filled up by the customer. They are composed of variables representing attributes characterizing the project. Sets of values for particular variables are determined by ranges of attribute functions assigned to them. When simple criteria are combined into more complex ones and their hierarchical structure is specified, checklists contain subchecklists for simpler criteria. The design solution must not only satisfy requirements imposed by the customer, but has also to comply with the building code. Most code rules can be expressed in the form of either crisp or fuzzy constraints. These constraints can be added to the initial design specification.

Let us consider the example of designing a floor layout of a small office accessible for people on wheelchairs. It is assumed that the floor area of the whole office should be about 75 m^2. It should consist of a small secretariat room, medium-sized study, hall with the size about 12 m^2. The shape of the study should be *square-like*. All above mentioned requirements are fuzzy and soft. From the building code and the norm DIN 18040-1, which determines the parameters of the toilet for disabled people, it can be inferred that the toilet should have about 5 m^2 and should be *square-like*. These two requirements are treated as fuzzy and hard. Moreover, the study should be accessible from the secretariat, all rooms should be accessible from the hall, and the hall should be accessible from the outside. These three requirements are crisp and hard. The study should have an eastern exposure, while the secretariat a northern one. These two requirements are crisp and soft. The values *about* 12 m^2, *about* 5 m^2, and *about* 75 m^2 can be interpreted as fuzzy numbers (fuzzy sets). The values such as *small*, *medium size*, *square-like* can be explained by means of linguistic variables.

Thus, the concepts of a fuzzy set and a linguistic variable are described. A fuzzy set B in a space X is a set of ordered pairs $B = \{(x, \mu_B(x))|x \in X\}$, where $\mu_B(x) : X \rightarrow [0, 1]$ is a membership function which associates with each $x \in X$ a real number of $[0, 1]$ [11]. This number represents the grade of membership of x in B. A linguistic variable is informal in nature. The intuitive definition determines a linguistic variable as a quadruple (Y, T, U, m), where Y is the name of the variable, T is the set of linguistic values, U is the universe of discourse and m is the interpretation, which combines elements of T with fuzzy sets on U (it is defined in a similar way in [12]).

For the customer, the value *small* characterizing the secretariat room denotes the area from 12 to 16 m^2, the value *medium size* denotes the area from 20 to 30 m^2, see Fig. 1. These are the linguistic values of the linguistic variable *area*.

As far as the squareness of the given shape is considered, the degree of squareness of a shape A can be computed as $max\{0, 1 - \frac{2a}{s(A)}\}$, where $s(A)$ is the area of the smallest square enclosing A and a is the area of the complement of A to $s(A)$. The factor 2 or larger is necessary to ensure the compliance with the natural meaning of the word *square*. The degree of squareness of the shape A presented in Fig. 2a equals $1 - \frac{2}{9} = \frac{7}{9} \approx 0.78$.

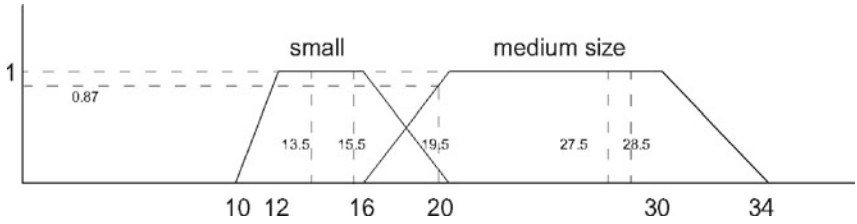


Fig. 1 Membership functions of the two terms describing room sizes

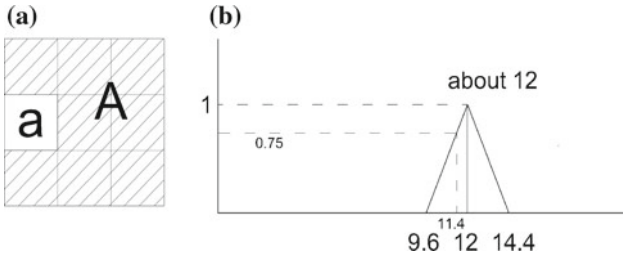


Fig. 2 **a** A *square* divided on a shape *A* and its complement *a*, **b** A triangular membership function of the fuzzy number *about* 12

The fuzzy values determining sizes of the rooms (about 12.5 and 75 m^2) are symmetric triangle numbers with support lengths equal to 40 % of the sharp numbers. For the sharp number 12 presented in Fig. 2b the support length of the triangle representing the fuzzy number 12 is equal to 4.8 (40 % of 12). Therefore the end points of the segment being the triangle base are set as 9.6 and 14.4, respectively. The numbers from the interval [9.6, 14.4] belong to the fuzzy set *about* 12 in grades appointed by the shape of triangle, for example the area equal to 11.4 m^2 has about 12 m^2 in grade 0.75, while the area smaller or equal to 9.6 m^2 has about 12 m^2 in grade 0.

The initial design drawing of an office layout created on the basis of the above specification is presented in Fig. 3a. It is composed of polygons which are placed in an orthogonal grid and represent rooms of the layout. The adjacency relation between rooms is expressed by line segments shared by polygons, while the accessibility relation is represented by line segments with small rectangles located on them. The area of the study (labelled *Study*) equals 27.5 m^2, the area of the secretariat room (labelled *Secretariat*) equals 15.75 m^2, the areas of the hall (labelled *Hall*) and the toilet (labelled *Wc*) are equal to 12 and 3 m^2, respectively. The area of the whole layout is equal to 58.25 m^2.

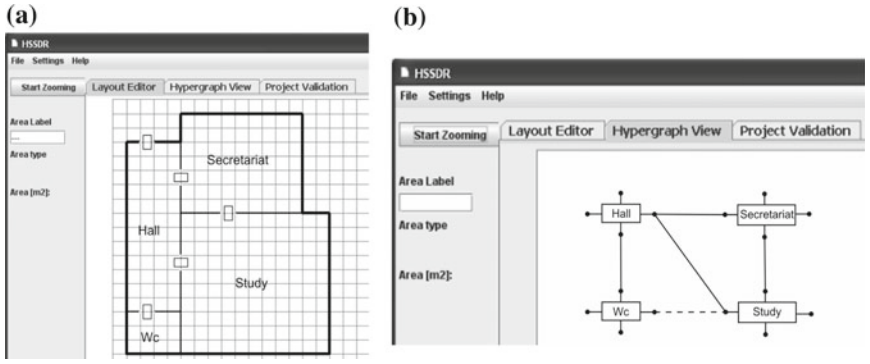


Fig. 3 Design diagrams representing a layout of **a** a small office and **b** a hypergraph representing this layout

3 Hierarchical Hypergraphs

In this paper the internal representation of design drawings in the form of attributed hierarchical hypergraphs described in [4] has been adopted. The considered hypergraphs are composed of hyperedges corresponding to object (drawing) components and nodes corresponding to fragments of these components. Hypergraph arcs connecting nodes and drawn as line segments represent relations among component fragments. Lines in different styles correspond to different types of relations. Hyperedges and nodes are labelled by names of components and their fragments, respectively. Hyperedges represent objects on different levels of detail. Each hyperedge representing an object component can contain a hierarchical hypergraph representing the layout of subcomponents of this component. The characteristic features of the drawing components are represented by attributes assigned to the corresponding elements of the hypergraph.

Let Σ be a fixed alphabet of labels and let Ω be a set of attributes.

Definition 1 An **attributed hierarchical hypergraph** over Σ and Ω is a system $G = (E, V, t, A, lb, att, ch)$, where:

1. E is a nonempty finite set of hyperedges representing object components,
2. V is a nonempty finite set of nodes representing fragments of object components,
3. $t : E \to V^*$ is a mapping assigning sequences of different nodes to hyperedges,
4. $A \subseteq V \times V$ and $\forall a = (v_1, v_2) \in A \; \exists e_1, e_2 \in E : e_1 \neq e_2, v_1 \in t(e_1), v_2 \in t(e_2)$, is a finite set of arcs representing relations between fragments of components,
5. $lb : E \cup V \cup A \to \Sigma$ is a labelling function of hypergraph elements,
6. $att : E \cup V \to 2^{\Omega}$ is an attributing function, where 2^{Ω} is a set of all subsets of Ω,
7. $ch : E \to 2^{E \cup V \cup A}$ is a child nesting function, such that none hypergraph element can be nested in two different hyperedges, a hyperedge cannot be its own child, and nodes of a nested hyperedge e of E are nested in the same hyperedge as e.

The initial design drawing is automatically transformed by the system to its internal representation. The hypergraph representing the drawing presented in Fig. 3a is shown in Fig. 3b. Four hyperedges representing rooms (the study—*Study*, secretariat —*Secretariat*, hall—*Hall*, toilet—*Wc*) are drawn as rectangles. Hypergraph nodes assigned to hyperedges and representing walls of rooms are drawn as black dots. Adjacency relations between rooms represented by arcs labelled *adj* are shown as dashed line segments connecting hypergraph nodes, while accessibility relations represented by arcs labelled *acc* are presented as continuous line segments. To each hyperedge attributes which characterize the corresponding room are assigned. All hyperedges have the attribute *area*, the hyperedges labelled *Secretariat* and *Study* have the attribute *exposure*. The hyperedges representing the study and the toilet have also the attribute *shape*.

Attributed hierarchical hypergraphs representing design drawings are valuated, i.e., the attributes assigned to their hyperedges and nodes have specified values. Hypergraph attributes are divided into two types. Attributes of the first type, called *crisp* attributes, have always crisp values, while attributes of the second type, called *fuzzy* attributes, can take crisp values or can be treated as linguistic variables with linguistic values. In the design specification fuzzy attributes have linguistic values, which are represented by the membership functions specifying the degree of belonging to fuzzy sets. However in hypergraphs representing instance drawings they take crisp values. Then, in order to check the compatibility of the drawings with the specification, these crisp values are compared with linguistic values of the specification.

The valuated attributed hierarchical hypergraph is defined as follows. Let Ω be a set of attributes and D be a set of their possible values.

Definition 2 A **valuated** attributed hierarchical hypergraph is a pair $G_{val} = (G, Val)$, where:

1. $G = (E, V, t, A, lb, att, ch)$ is an attributed hierarchical hypergraph,
2. Val is a family of partial functions assigning values to attributes of the hyperedges and nodes in such a way that $\forall \omega \in \Omega \; val_\omega : (E \cup V, \omega) \rightarrow D$.

One of the designs drawn on the basis of the initial specification is presented in Fig. 4a. The attribute *exposure* is the one which takes only crisp values. For the hyperedge labelled *Secretariat* it is set to *north*, while for the hyperedge labelled *Study* it has the value *east*. The attributes *area* for hyperedges representing rooms *Secretariat*, *Study*, *Hall* and *Wc* are fuzzy ones and are set in the specification to linguistic values *small*, *medium size*, *about* 12 m^2, *about* 5 m^2, which are represented by the fuzzy set membership functions shown in Figs. 1 and 2b, respectively. The attribute *shape* assigned to the hyperedges labelled *Study* and *Wc* has in the specification the linguistic value *square-like* represented by the measure of squareness. The values of the these fuzzy attributes are set in the presented drawing in such a way that the area of the study (*Study*) equals 28.5 m^2, the area of the secretariat (*Secretariat*) equals 12.25 m^2, the areas of the hall (*Hall*) and the toilet (*Wc*) are equal to 15.75 and 4.5 m^2, respectively. The area of the whole layout is equal to 61 m^2. The other possible design drawing is shown in Fig. 4b. In this design the area of the study equals

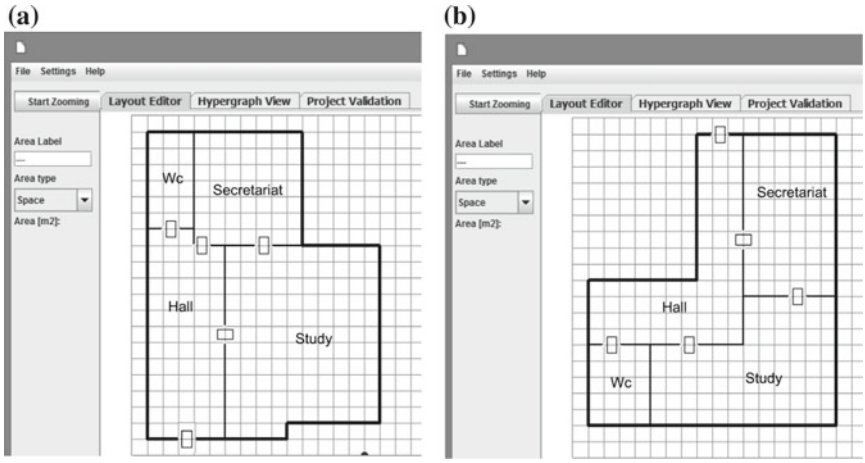


Fig. 4 Design diagrams representing two possible layouts of a small office: **a** the initial design and **b** the design after evaluation

19.5 m^2, the area of the secretariat equals 13.5 m^2, the areas of the hall and the toilet are equal to 13.75 and 5.0 m^2, respectively. The area of the whole layout is equal to 51.75 m^2.

4 Fuzzy Evaluation by Approximate Reasoning

This section deals with approximate reasoning as a method of assessment potential layout designs. In this reasoning process fuzzy rules describing compatibility of designs with the specification are used. In imprecise premises occur membership functions representing fuzzy values [12]. Imprecise conclusions computed on the basis of crisp values of the instance designs lead to partial evaluations making up the overall fuzzy evaluation of designs.

The design drawing instances generated by the designer are evaluated by the system. Then the solutions with the evaluation value over the specified threshold are presented to the designer as they satisfy the design constraints and requirements in the best possible way. In the next step the designer evaluates the functionality and aesthetics of the obtained drawing instances. Analysis of the best solutions presented by the system enables the customer and designer to correct some requirements or constraints.

As it has been considered the visual language L_{layout} contains drawings generated by the designer on the monitor screen and representing simplified 2D-floor layouts. L_{layout} can be treated as an infinitely large design space, i.e., designer's L_{layout} -universum.

Definition 3 By the **space of potential floor-layout design solutions** we understand a set S generated by the designer during the design process and such that $S \subset L_{layout}$.

Drawing instances have their internal representations in the form of valuated attributed hypergraphs. As for each obtained drawing instance all its components have exact values of their features (like room sizes, shapes), the hypergraph attributes always have crisp values. The correspondence degree of the instance drawing to the task specification can be approximately reasoned by comparing these crisp values with fuzzy values assigned to fuzzy attributes by membership functions of the specification and by comparing other crisp values with sharp values and hard values of the specification. As the result of this reasoning the membership degree for each design feature specified in a fuzzy way is computed.

The final evaluation of a design drawing is a sum of degrees of particular criteria fulfillment. The evaluation of criteria with sharp values and hard values is added to degrees of memberships obtained as a result of fuzzification for attributes corresponding to fuzzy criteria. Each criterion with a crisp value is satisfied in either 0 or 1 degree. The degrees of other criteria satisfaction are in the interval [0,1].

The design shown in Fig. 3a cannot be accepted as the area of the toilet is too small to be comfortable for disabled people. Thus, let us consider the correspondence degree of the drawing shown in Fig. 4a to the design task specification. As the result of approximate reasoning the system infers that the secretariat is *small* in degree 1, the study is *medium-sized* in degree 1, the hall has *about* 12 m^2 in degree 0, while the toilet has *about* 5 m^2 in degree 0.5. The area of the whole layout has about 75 m^2 in degree 0.07. The study has the *square-like* shape in degree 0.58, while the toilet has the *square-like* shape in degree 0. Therefore this design also does not comply with the considered building code. The corrected design is shown in Fig. 4b. In this drawing the toilet has *about* 5 m^2 in degree 1, and has the *square-like* shape in degree 0.6.

Drawings of the space of potential design solutions match the design specification in various degrees. For drawings with crisp and hard requirements satisfied in 1 degree and with fuzzy and hard requirements satisfied in degree greater that 0, their *correspondence degree* to the specification is determined. It is computed as the average of membership degrees of respective exact values of drawing component attributes to fuzzy values of attributes in the specification.

If we assign a correspondence degree to each element of the space of potential design solutions and treat it as a membership function value, then we obtain a fuzzy space of solutions.

Definition 4 By the **fuzzy space of solutions** we understand a set of fuzzy subsets defined on S by the function $fzz : S \rightarrow Fuzzy(L_{layout})$, where $Fuzzy(L_{layout})$ is a family of fuzzy subsets on L_{layout}-universum.

The final evaluation of generated solutions enables the system to graduate them according to the degree of particular criteria fulfillment.

In the design shown in Fig. 4b the secretariat is *small* in degree 1 (13.5 m^2), the study is *medium-sized* in degree 0.87 (19.5 m^2), the hall has *about* 12 m^2 in degree

0.25 (13.75 m^2). The study has the *square-like* shape in degree 0.5. The area of the whole layout has about 75 m^2 in degree 0 (51.75 m^2). The sum of the correspondence degrees of all seven attributes with linguistic values computed for the drawing shown in Fig. 4b equals to 4.22. As five other criteria (exposure of the study and secretariat, and three conditions related to the accessibility of rooms) are satisfied, the evaluation sum of the design is equal to 9.22. As the maximal possible sum is 12, the overall evaluation of the solution is 0.768.

In the presented approach the solutions with the evaluation value over 0.6 are presented to the designer. Each such a solution is called admissible. The designer's analysis and functional-aesthetic evaluation of admissible solutions enables him/her to choose the ones which are to be developed further.

5 Conclusions

This paper extends the concept of design reasoning with the use of logic languages and design knowledge stored in hypergraph internal representations of design drawings by introducing a fuzzy classification based on approximate reasoning. Due to the fuzzy classification the system is able to evaluate the correspondence of potential solutions to the project specification.

In the future the mechanism of automatic generation of potential solutions on the base of a set of initial sketches and a set of pattern graphs coding their characteristic features together with design requirements will be developed. Moreover, the RASE methodology in code compliance checking will be used.

References

1. Bittermann, M.: A computational design system with cognitive features based on multi-objective evolutionary search with fuzzy information processing. In: Gero, J. (ed.) DCC 2010, pp. 505–524. Stuttgart, Germany (2010)
2. Clayton, M., Fudge, P., Thompson, J.: Automated plan review for building code compliance using BIM. In: EG-ICE 2013 Workshop, Vienna, Austria (2013)
3. Genesereth, M., Nillson, N.: Logical Foundations of Artificial Intelligence. Morgan Kaufmann, Los Altos (1987)
4. Grabska, E., Łachwa, A., Ślusarczyk, G.: New visual languages supporting design of multi-storey buildings. Adv. Eng. Inform. **26**, 681–690 (2012)
5. Grabska, E., Ślusarczyk, G.: Knowledge and reasoning in design systems. Autom. Constr. **20**, 927–934 (2011)
6. Grabska, E., Ślusarczyk, G., Gajek, S.: Knowledge representation for human-computer interaction in a system supporting conceptual design. Fundamenta Informaticae **124**, 91–110 (2013)
7. Kasim, T., Li, H., Rezgui, Y., Beach, T.: Generic approach for automating standards and best practices. compliance for sustainable building design. In: EG-ICE 2013 Workshop, Vienna, Austria (2013)
8. Łachwa, A., Grabska, E., Ślusarczyk, G.: Fuzzy interpretation of layout hypergraphs. Schedae Informaticae **15**, 165–173 (2006)

9. Macit, S., Ilal, M., Gunaydin, H., Suter, G.: Izmir municipality housing and zoning code analysis and representation for compliance checking. In: EG-ICE 2013 Workshop, Vienna, Austria (2013)
10. Palacz, W., Grabska, E., Gajek, S.: Conceptual designing supported by automated checking of design requirements and constraints. In: Frey, D., Fukuda, S., Rock, G. (eds.) Improving complex systems today. Advanced Concurrent Engineering, pp. 257–266. Springer, London (2011)
11. Zadeh, L.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
12. Zadeh, L.: The concept of a linguistic variable and its application to approximate reasoning: part 1. Inf. Sci. **8**, 199–249 (1975)

Classification Based on Incremental Fuzzy $(1 + p)$-Means Clustering

Michal Jezewski, Jacek M. Leski and Robert Czabanski

Abstract Fuzzy clustering is often applied to determine the rules of the fuzzy rule-based classifiers (usually the antecedents only). In this work a new fuzzy clustering approach is proposed for such a purpose. The idea consists in alternating clustering of the objects from two classes with the prototypes obtained after the previous clustering not allowed to move during the current clustering. As a result each clustering provides new location of a single prototype. The classification quality obtained by the fuzzy rule-based classifier using the proposed clustering was compared with the Lagrangian SVM method on several benchmark databases.

Keywords Clustering · Fuzzy rule-based classification

1 Introduction

Classification consists in assigning objects to predefined classes. There are a lot of classification methods, the Support Vector Machine (SVM) [12] is regarded as one of the most successful ones. The family of fuzzy rule-based classifiers may be divided into: Mamdani-Assilan classifiers (the linguistic terms are used in both antecedents and consequents of the if-then rules) and Takagi-Sugeno-Kang classifiers (linguistic terms are used in antecedents, but consequents are functions—usually linear—of inputs). The goal of clustering is to find groups (clusters) of similar objects and their centers (prototypes). In fuzzy clustering an object may partially belong to several clusters with a membership degree within the range [0, 1]. The most popular fuzzy clustering method is the fuzzy c-means (FCM) method. The are a lot of research describing fuzzy clustering, e.g. FCM for ordinal valued attributes [1], quadratic

M. Jezewski (✉) · J.M. Leski · R. Czabanski
Institute of Electronics, Silesian University of Technology, Gliwice, Poland
e-mail: michal.jezewski@polsl.pl

J.M. Leski
e-mail: jacek.leski@polsl.pl

R. Czabanski
e-mail: robert.czabanski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_48

entropy—[8] or kernel-based [14] FCM, accelerating FCM [15], FCM algorithms for very large data [5], fuzzy clustering observer-biased [3] or with a decreasing number of clusters [4]. Fuzzy rule-based clustering is proposed in [2, 13] describes robust clustering.

In the case of fuzzy rule-based classifiers, fuzzy clustering is often applied to determine the rules (usually the antecedents only). In that case, clustering may be performed for the whole training set or separately for each class in the training set. In the second case it is better when the objects from the class(es) not being clustered at the moment influence the clustering of a given class. The method representing such approach—fuzzy $(c + p)$-means clustering—was proposed in [11], where c concerns prototypes in the class being clustered, and p concerns fixed ("a priori known") prototypes in the other class(es). For two classes, the algorithm consists in alternating clustering of the objects from both classes, taking as the known prototypes the ones from the opposite class. The goal is to perform the clustering of a given class in such a way that the prototypes are attracted to the regions where the objects are dense and simultaneously repulsed from the objects belonging to the other class(es) [11]. The proposed clustering method was applied to determine the rules of the fuzzy rule-based classifier [11].

The aim of this work is to propose the incremental fuzzy $(1 + p)$-means clustering—a clustering method for determining the rules of the fuzzy rule-based classifier, and to verify its usefulness by the obtained classification quality. The goal of the incremental fuzzy $(1 + p)$-means clustering is to perform the clustering of a given class with the prototypes being repulsed from the prototypes in the second class and in the class being clustered. In [11] the clusterings into different numbers of clusters are independent from each other. In the incremental fuzzy $(1 + p)$-means clustering, p prototypes (in both classes) in the current clustering, are the result of the previously performed clustering into p clusters, and are not allowed to move during the current clustering. As a result, each successive clustering provides new location of a single prototype. In other words, having the result (prototypes) of the clustering into a given number of clusters we have the results of the clustering into smaller numbers of clusters. The classification quality obtained by applying the proposed clustering to determine rules of the fuzzy rule-based classifier was compared with the Lagrangian SVM classifier [12]. The research concerning the fuzzy clustering methods dedicated to fuzzy rule-based classification were also presented by us in [6, 7]. In these methods, prototypes from two different classes were attracted to each other.

2 Incremental Fuzzy $(1 + p)$-Means Clustering

The proposed fuzzy clustering is performed using FCM method, however a new algorithm consisting in successive FCM clusterings is proposed. The FCM criterion function has the form [15]

$$J(\mathbf{U}, \mathbf{V}) = \sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik})^m \, d_{ik}^2, \tag{1}$$

with the constraints

$$\underset{1 \le k \le N}{\forall} \quad \sum_{i=1}^{c} u_{ik} = 1, \tag{2}$$

where c (N) denotes number of clusters (objects), $\mathbf{U}$ ($\mathbf{V}$) denotes partition (prototypes) matrix, d_{ik} stands for the Euclidean distance between the ith prototype and the kth object $\mathbf{x}_k$. Parameter $m > 1$ influences the repulsive force between prototypes; the lower value of m, the higher repulsive force. The necessary conditions for minimization of the criterion (1) may be found in [15].

At the beginning of the proposed algorithm, the objects from ω_1 class in the training set are clustered using FCM into 2 clusters starting with 2 initial prototypes in that class. The obtained prototypes and the two initial prototypes in ω_2 class are used to start the $(2 + 2)$-means clustering of the objects from ω_2 class, the prototypes from the previous clustering are not allowed to move. As a result, new locations of 2 prototypes are obtained. In the further steps the algorithm consists in alternating $(1 + p)$-means clustering of the objects from ω_1 and ω_2 classes, where p increases with the step of one, starting from $p = 4$. Each clustering starts with p prototypes obtained after the previous clustering and with one initial prototype in the class being clustered. The prototypes from the previous clustering are not allowed to move during the current clustering. Therefore, each clustering provides new location of a single prototype. In other words, result (prototypes) of the clustering into a given number of clusters includes results of the clustering into smaller numbers of clusters. Initial prototypes are selected from the boundary of the convex hull [11] of the class being clustered.

The incremental fuzzy $(1 + p)$-means clustering algorithm may be written as:

1. Fix $k = (1 + p)$ $(k \ge 4)$ and m $(m > 1)$.
2. Perform the FCM clustering of the data from ω_1 class into 2 clusters. Start the clustering with the first 2 initial prototypes in ω_1 class. This way, the prototypes $\mathbf{v}_1$ and $\mathbf{v}_2$ are obtained.
3. Perform the $(2 + 2)$—means clustering of the data from ω_2 class. Start the clustering with $\mathbf{v}_1$ and $\mathbf{v}_2$ as the prototypes 1st and 2nd (p prototypes) and with the first 2 initial prototypes in ω_2 class as the prototypes 3rd and 4th. During the clustering, update only the prototypes 3rd and 4th. This way, new locations of 2 prototypes $\mathbf{v}_3$ and $\mathbf{v}_4$ are obtained.
4. If $k = 4$ then stop.
5. Set the number of clusters $l = 5$ and the index of the initial prototypes $r = 3$.
6. Perform the $(1 + l - 1)$—means clustering of the data from ω_1 class. Start the clustering with $\mathbf{v}_i$ $(i = 1, 2, \ldots, l - 1)$ as the first $l - 1$ prototypes (p prototypes) and with the rth initial prototype in ω_1 class for the lth prototype. During the

clustering, update only the lth prototype. This way, new location of a single prototype $\mathbf{v}_l$ is obtained.
7. If $l = k$ then stop, else set $l = l + 1$.
8. Perform the $(1 + l - 1)$-means clustering of the data from ω_2 class. Start the clustering with $\mathbf{v}_i$ $(i = 1, 2, \ldots, l-1)$ as the first $l-1$ prototypes (p prototypes) and with the rth initial prototype in ω_2 class for the lth prototype. During the clustering, update only the lth prototype. This way, new location of a single prototype $\mathbf{v}_l$ is obtained.
9. If $l = k$ then stop, else set $l = l + 1$, $r = r + 1$ and go to 6).

In the above algorithm, clustering starts from ω_1 class. However, it is possible to start it from ω_2 class obtaining different results. In each clustering, iterations (whose maximum number equaled to 50) were stopped as soon as the Frobenius norm of the successive $\mathbf{U}$ matrices difference was less than 10^{-3}.

Using the results of the proposed clustering into $k = (1 + p)$ clusters, k rules of the fuzzy rule-based classifier are determined, according to the description in the next section.

3 Nonlinear Fuzzy Rule-Based Classifier

The linear and nonlinear (kernel version) Iteratively Reweighted Least Squares (IRLS) classifiers were proposed in [10]. For the linear IRLS classifier the criterion function for the kth iteration has the form [6, 7, 10]

$$J^{(k)}\left(\mathbf{w}^{(k)}\right) \triangleq \frac{1}{2}\left(\mathbf{X}\mathbf{w}^{(k)} - \mathbf{1}\right)^{\top} \mathbf{H}^{(k)}\left(\mathbf{X}\mathbf{w}^{(k)} - \mathbf{1}\right) + \frac{\tau}{2}\left(\widetilde{\mathbf{w}}^{(k)}\right)^{\top} \widetilde{\mathbf{w}}^{(k)}, \qquad (3)$$

where $\mathbf{w} = \left[\widetilde{\mathbf{w}}^{\top}, w_0\right]^{\top} \in \mathbb{R}^{t+1}$ is the weight vector of the linear discriminant function $d(\mathbf{x}_i) \triangleq \mathbf{w}^{\top}\mathbf{x}_i' = \widetilde{\mathbf{w}}^{\top}\mathbf{x}_i + w_0$, $\mathbf{x}_i \in \mathbb{R}^t$ denotes the ith object from the N element training set, $\mathbf{x}_i' = \left[\mathbf{x}_i^{\top}, 1\right]^{\top}$, $\mathbf{1}$ is the vector with all entries equal to 1. The matrix $\mathbf{X}$ is defined as follows $\mathbf{X}^{\top} \triangleq \left[\varphi_1\mathbf{x}_1', \varphi_2\mathbf{x}_2', \ldots, \varphi_N\mathbf{x}_N'\right]$, where φ_i (equal to $+1$ or -1) is the class label indicating an assignment of the ith object to one of two classes ω_1 or ω_2. Various definitions of the matrix $\mathbf{H}^{(k)} = \operatorname{diag}\left(h_1^{(k)}, h_2^{(k)}, \ldots, h_N^{(k)}\right)$ provide various loss functions (approximations of the misclassification error). For example, if we define $h_i^{(k)}$ as equal to 0 for $e_i^{(k-1)} \geq 0$ and equal to 1 for $e_i^{(k-1)} < 0$, where $\mathbf{e}^{(k-1)} = \mathbf{X}\mathbf{w}^{(k-1)} - \mathbf{1}$, then the Asymmetric SQuaRe (ASQR) loss function is obtained. Other loss functions (ALIN, SIG, ASIGL, AHUB, ALOG, ALOGL) were presented in [10]. Additionally, matrix $\mathbf{H}$ ensures asymmetrization (relaxation) of the loss function. The second term of the criterion (3) is responsible for the maximization of the margin of separation, the parameter τ controls the trade-off between both components.

In the presented work, the fuzzy rule-based classifier using Takagi-Sugeno-Kang rules with linear functions in consequents, proposed in [9], was used. The antecedents were determined using the results of the proposed clustering. The criterion (3) of the linear IRLS classifier, with some substitutions, was applied to find the consequents. The details are described in [6, 7]; however, there are two differences. First, instead of the "final prototypes" used in [6, 7], prototypes $\mathbf{v}_i$ $(i = 1, 2, \ldots, k = (1 + p))$ were used. Second, the dispersions of the Gaussian membership functions were calculated differently than in [6, 7], according to the formula (4). Additionally, different numbers of iterations and stopping conditions in the conjugate gradient algorithm, minimizing criterion (3), were used.

$$(s_{in})^2 = \frac{\sum_{k=1}^{N} u_{ik} \, (x_{kn} - v_{in})^2}{\sum_{k=1}^{N} u_{ik}}. \tag{4}$$

4 Numerical Experiments

To verify the classification quality obtained by the fuzzy rule-based classifier using the proposed clustering, seven benchmark databases were applied: 'Banana' (Ban), 'Breast Cancer' (BreC), 'Diabetis' (Diab), 'Heart' (Hea), Ripley synthetic (Syn), 'Thyroid' (Thy) and 'Titanic' (Tita). Each database was represented by 100 training and testing sets. The classification quality was compared with the Lagrangian SVM (LSVM) method [12] with the Gaussian kernel function. The details concerning the origin of databases and the values of the parameters of LSVM are described in [6]; however, LSVM parameters were determined using the first 5 pairs of the training and the testing sets (instead of 10 as in [6]). Class ω_1 (ω_2) contained objects with class labels equal to $+1$ (-1). During the classification, sometimes the discriminant function for a given object was equal to 0, then ω_2 class was assumed. All the results were expressed in percents. The values of the fuzzy rule-based classifier parameters—number of clusters (rules) and τ in the IRLS criterion (3) were found using the first 5 pairs of the training and the testing sets according to the procedure, which is outlined below.

The first 5 training sets were merged into a single dataset which was clustered using the proposed clustering method into a given number of k-clusters ($k = (1+p)$, k varied from 4 to 20). This way the antecedents of k-rules were found and they were common for all 100 training and testing sets. The k-consequents were determined, using the IRLS criterion with a given value of τ (searched within the range of 0.1 to 10.0 with a step of 0.3), separately for each of the first 5 training sets. The number of rules (k) with the corresponding value of τ ensuring the highest classification quality for the first 5 testing sets were chosen to calculate the final result. To calculate the final result, consequents were determined separately for each of all 100 training

Table 1 Classification quality obtained for various variants of the proposed method

m	Start	Ban	BreC	Diab	Hea	Syn	Thy	Tita
1.1	ω_1	11.142	25.688	23.520	16.770	9.323	2.840	22.286
		(0.529)	(4.343)	(1.859)	(3.405)	(0.519)	(1.911)	(1.134)
		20	7	18	18	9	15	6
	ω_2	10.801	**24.065**	**23.020**	**13.570**	**9.107**	**2.760**	22.335
		(0.415)	(4.605)	(1.769)	(3.349)	(0.460)	(2.179)	(1.188)
		13	18	12	15	4	11	11
1.3	ω_1	11.027	25.286	23.813	17.350	9.600	4.053	22.287
		(0.554)	(4.409)	(1.984)	(3.092)	(0.582)	(2.313)	(1.098)
		20	20	18	12	12	5	5
	ω_2	10.860	26.052	24.723	16.280	9.122	2.867	22.339
		(0.537)	(4.367)	(2.095)	(3.490)	(0.454)	(1.601)	(1.077)
		11	18	19	18	4	5	10
1.5	ω_1	10.893	25.636	24.067	17.190	9.483	3.613	**22.269**
		(0.557)	(4.509)	(1.823)	(3.348)	(0.503)	(2.061)	(1.087)
		20	17	14	13	7	4	5
	ω_2	10.620	24.143	24.407	16.520	9.279	4.947	22.594
		(0.450)	(4.484)	(1.987)	(3.463)	(0.586)	(3.046)	(1.282)
		11	20	14	18	6	10	12
1.7	ω_1	10.797	24.948	23.873	15.980	10.273	6.987	22.294
		(0.469)	(4.508)	(1.731)	(3.219)	(1.338)	(6.576)	(1.089)
		20	15	10	13	18	14	5
	ω_2	10.639	25.247	23.627	15.600	9.344	4.947	22.372
		(0.535)	(3.842)	(1.795)	(3.194)	(0.559)	(2.600)	(1.150)
		20	20	8	17	7	8	11
2.0	ω_1	10.510	25.416	23.937	15.890	10.107	5.053	22.298
		(0.404)	(4.707)	(1.600)	(3.272)	(0.962)	(2.838)	(1.242)
		14	20	13	17	17	13	13
	ω_2	**10.478**	25.623	23.770	15.100	10.260	3.173	22.324
		(0.502)	(3.880)	(1.785)	(3.280)	(1.104)	(1.958)	(1.201)
		15	4	19	16	18	4	19

sets and the mean value and the standard deviation of the classification error for the corresponding 100 testing sets provided the final result. The above procedure was performed for various values of m (from the set {1.1, 1.3, 1.5, 1.7, 2.0}) when starting the clustering from ω_1 or ω_2 class. The ASQR loss function was used in the IRLS criterion (3).

Table 1 presents the quality of the **c**lassification based on the **i**ncremental fuzzy $(\mathbf{1}+\mathbf{p})$-means clustering (CI1P). Clustering was performed using various values of m and starting from ω_1 or ω_2 class. Each cell of the table contains: mean classification error (top), standard deviation (middle) and the number of rules providing them (bottom). For each database the best result is in a boldface. The class used to start the

clustering influences the clustering results, and therefore the classification quality. For all considered values of m, in case of Ban and Hea it is better to start the clustering from ω_2 class; in case of Tita, clustering should be started from ω_1 class. For the rest of the databases the preferable class depends on the value of m. However, taking into consideration only the 7 best results, for all databases except Tita clustering should be started from ω_2 class. Analyzing the value of m and the classification quality starting the clustering from the same class, monotonous relation is not observed for almost any database, the differences of the classification error are at the level from 0.03 % (Tita, ω_1) to 4.15 % (Thy, ω_1). However, 5 of the 7 best results were obtained for $m = 1.1$. Taking the above into consideration, the results obtained by clustering with $m = 1.1$ and starting from ω_2 class are summarized in Table 2.

Two first rows of Table 2 compare the quality of CI1P ($m = 1.1$, start from ω_2) with the LSVM method. For all databases except Ban, CI1P provided lower classification error than LSVM. The statistical significance of the differences between LSVM and CI1P was checked using a paired t-test and is presented in the last row of the table—the differences are statistically significant for all databases except Diab and Tita. The results in Table 1 and in the second row of Table 2 were obtained using ASQR loss function in the IRLS criterion while determining the consequents of rules. For the number of rules chosen using ASQR, other loss functions described in [10] were applied (with alpha = 2.0 for SIG and ASIGL), the value of τ was searched separately for each loss function. The best results with the corresponding loss functions are presented in the third row of Table 2. It may be noticed that changing the loss function slightly improves classification quality for all databases. In case of Tita, extremely small (or equal to 0) distances between prototypes and objects were noticed. The distances lower than the machine precision were treated as equal to 0 and in such cases a special update of the partition matrix was applied.

Figures 1 and 2 present discrimination curves obtained for the first training set of the two-dimensional databases Ban and Syn. Prototypes are marked by circles, ellipses visualize dispersion in the antecedents of the rules.

Table 2 Comparison of the classification quality ($m = 1.1$, start from the ω_2 class)

		Ban	BreC	Diab	Hea	Syn	Thy	Tita
LSVM		10.349	25.987	23.217	16.270	9.312	4.107	22.446
		(0.414)	(4.154)	(1.595)	(3.250)	(0.530)	(2.280)	(1.121)
CI1P	ASQR	10.801	24.065	23.020	13.570	9.107	2.760	22.335
		(0.415)	(4.605)	(1.769)	(3.349)	(0.460)	(2.179)	(1.188)
		13	18	12	15	4	11	11
	Other	10.726	23.688	22.437	13.380	8.704	2.120	22.196
		(0.443)	(4.274)	(1.639)	(3.308)	(0.395)	(1.394)	(1.058)
		ALIN	ASIGL	ALIN	ALOGL	ASIGL	ALIN	ASIGL
Stat. signif. (p value)		+	+	−	+	+	+	−
		<0.001	<0.001	0.2573	<0.001	<0.001	<0.001	0.2579

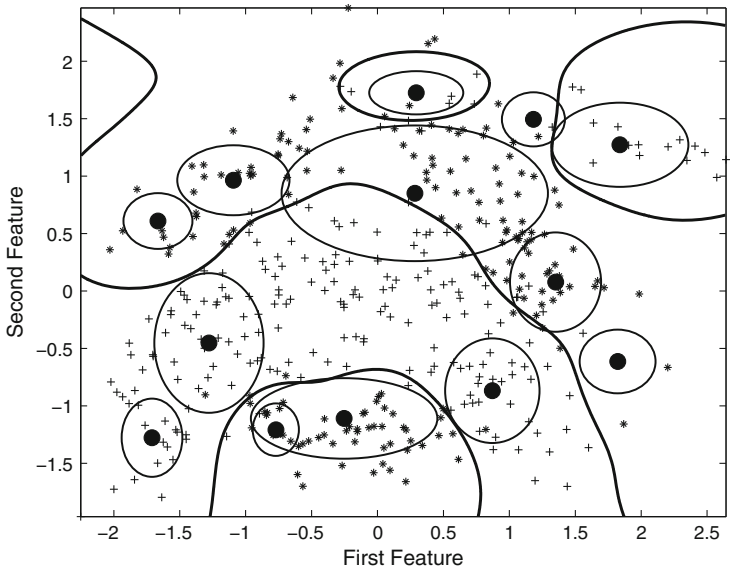


Fig. 1 Discrimination curve for the Ban database (the first training set)

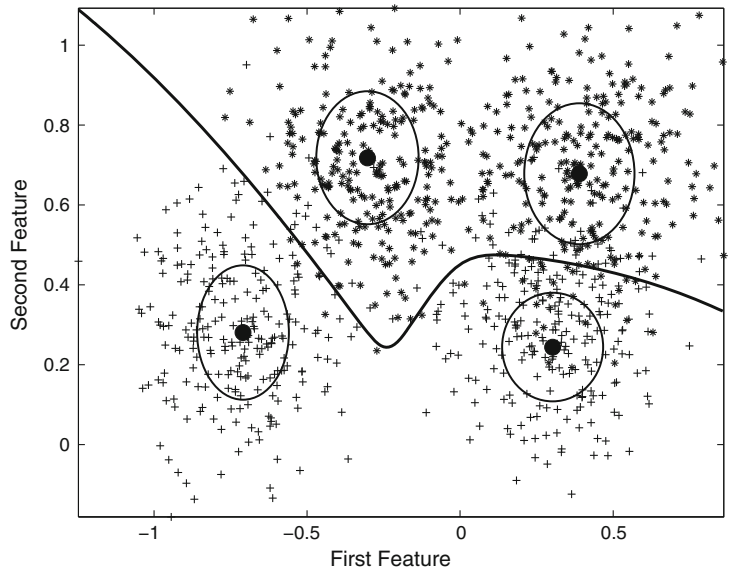


Fig. 2 Discrimination curve for the Syn database (the first testing set)

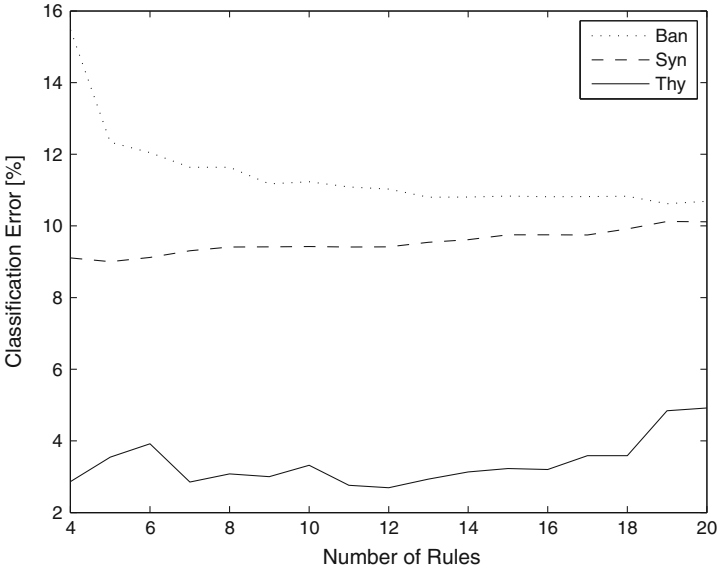


Fig. 3 The relation between the number of rules and the classification error

Figure 3 presents the relation between the number of rules and the classification quality (final result—for all 100 testing sets) for Ban, Syn and Thy. In case of Ban (Syn) rather clear relation is observed—the classification error decreases (increases) with the increase of the number of rules. For Thy the classification error first decreases and next increases. In the described figure, the final results obtained for each considered number of rules were presented. However, according to the procedure outlined at the beginning of this section, the final result is calculated for the number of rules chosen using the first 5 pairs of the training and the testing sets. Therefore, the chosen number of rules may not correspond to the best classification quality observed in Fig. 3. For example, for Ban (Syn) the chosen number of rules was 13(4), whereas the best classification quality is observed for 19(5) rules. In Figs. 1, 2 and 3, clustering and classification were performed using the parameters that correspond to the results in the second row of Table 2.

5 Conclusions

In the presented paper, a fuzzy clustering method for determining the rules of the fuzzy rule-based classifier was proposed. This approach uses fuzzy c-means method, but the new idea of adding one new prototype to the prototypes previously found was proposed. The usefulness of the proposed method was verified by the obtained classification quality. For six benchmark databases the obtained results were better

if compared to the Lagrangian SVM method. The plans for future embrace further modifications of the proposed clustering approach.

Acknowledgments This research was partially supported by statutory funds (BK-2014) of the Institute of Electronics, Silesian University of Technology. The work was performed using the infrastructure supported by POIG.02.03.01-24-099/13 grant: GeCONiI–Upper Silesian Center for Computational Science and Engineering.

References

1. Brouwer, R., Groenwold, A.: Modified fuzzy c-means for ordinal valued attributes with particle swarm for optimization. Fuzzy Sets Syst. **161**(13), 1774–1789 (2010)
2. Dave, R., Krishnapuram, R.: Robust clustering methods: a unified view. IEEE Trans. Fuzzy Syst. **5**(2), 270–293 (1997)
3. Fazendeiro, P., de Oliveira, J.: Observer-biased fuzzy clustering. IEEE Trans. Fuzzy Syst. **23**(1), 85–97 (2015)
4. Frigui, H., Krishnapuram, R.: Clustering by competitive agglomeration. Pattern Recognit. **30**(7), 1109–1119 (1997)
5. Havens, T., Bezdek, J., Leckie, C., Hall, L., Palaniswami, M.: Fuzzy c-means algorithms for very large data. IEEE Trans. Fuzzy Syst. **20**(6), 1130–1146 (2012)
6. Jezewski, M., Leski, J.: Nonlinear extension of the IRLS classifier using clustering with pairs of prototypes. In: Burduk, R., Jackowski, K., Kurzynski, M., Wozniak, M., Zolnierek, A. (eds.) CORES 2013, AISC, vol. 226, pp. 121–130. Springer, Switzerland (2013)
7. Jezewski, M., Leski, J.: Application of the conditional fuzzy clustering with prototypes pairs to classification. In: Gruca, A., Czachórski, T., Kozielski, S. (eds.) Man-Machine Interactions 3. AISC, vol. 242, pp. 397–405. Springer, Switzerland (2014)
8. Kannan, S., Ramathilagam, S., Chung, P.: Effective fuzzy c-means clustering algorithms for data clustering problems. Expert Syst. Appl. **39**(7), 6292–6300 (2012)
9. Leski, J.: An ε-margin nonlinear classifier based on fuzzy if-then rules. IEEE Trans. Syst. Man Cybern. Part B Cybern. **34**(1), 68–76 (2004)
10. Leski, J.: Iteratively reweighted least squares classifier and its ℓ_2- and ℓ_1-regularized kernel versions. Bull. Polish Acad. Sci. Tech. Sci. **58**(1), 171–182 (2010)
11. Leski, J.: Fuzzy (c+p)-means clustering and its application to a fuzzy rule-based classifier: toward good generalization and good interpretability. IEEE Trans. Fuzzy Syst. **23**(4), 802–812 (2015)
12. Mangasarian, O., Musicant, D.: Lagrangian support vector machines. J. Mach. Learn. Res. **1**, 161–177 (2001)
13. Mansoori, E.: FRBC: a fuzzy rule-based clustering algorithm. IEEE Trans. Fuzzy Syst. **19**(5), 960–971 (2011)
14. Pal, N., Sarkar, K.: What and when can we gain from the kernel versions of c-means algorithm? IEEE Trans. Fuzzy Syst. **22**(2), 363–379 (2014)
15. Parker, J., Hall, L.: Accelerating fuzzy-c means using an estimated subsample size. IEEE Trans. Fuzzy Syst. **22**(5), 1229–1244 (2014)

Imputation of Missing Values by Inversion of Fuzzy Neuro-System

Krzysztof Siminski

Abstract Incomplete data are common and require special techniques. The essential techniques are: marginalisation, imputation, and rough sets. The paper presents the imputation by inversion of the neuro-fuzzy system. First the neuro-fuzzy systems is trained with complete data. Next the system is inverted and the missing values are imputed. The complete and imputed data are used to train the final neuro-fuzzy system. The technique is limited to data items with one missing value. The paper is accompanied by numerical examples and statistical verification.

Keywords Evolutionary optimisation · Gradient descent · Memetic algorithm · Big-Bang-Big-crunch

1 Introduction

Missing values are common in real life data. The reasons of incompleteness are various: problems in data acquisition, random noise, invalid values, aggregation of data from various sources, difficulties in collecting all data (e.g. in medicine). The incomplete data may contain important information and are challenging task to handle.

There are three essential methods of handling incomplete data: marginalisation, imputation, and application of rough sets. Marginalisation is the simplest approach—this method deletes incomplete data items and leaves only complete ones. The deletion of missing data can be done in two ways: removal of incomplete data tuples (vectors)—reduction of dataset size [13], or removal of incomplete attributes—reduction of dimensionality of the task [5].

Imputation is more complicated than marginalisation but is used more frequently [14]. The missing values are imputed with zeroes, random numbers [31], average values for an attribute in the whole data set [19], average values for an attribute in the subset labelled with the same class as the incomplete data item [11], medians, all

K. Siminski (✉)
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: krzysztof.siminski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_49

possible values [12]. More sophisticated approaches have been also proposed [1, 2, 34]. Application of average values may impute missing values with nonexisting or unreasonable values. It may cause problems in some fields [31]. The third method of handling missing values is application of rough sets [4, 9, 15, 20, 25–28]. This method preserves the uncertainty of data.

The comparison of preprocessing techniques for incomplete data in clustering and neuro-fuzzy systems is presented in [18, 22]. The paper [22] compares various techniques of missing value handling in neuro-fuzzy systems. One of its conclusion is that for some data the best approach (resulting in the lowest error) is marginalisation. The incomplete data may represent the outliers and their deletion may improve the generalisation ability of the system. This is the anchor of our approach presented here. This paper proposes an imputation technique based on inversion of the fuzzy systems. The missing values are imputed with values elaborated with inverted neuro-fuzzy system.

2 Imputation by Inversion

The technique proposed in this paper may be described as follows:

1. First the incomplete train dataset is marginalised.
2. The resulting complete data set is used to create a fuzzy model for a neuro-fuzzy system.
3. The system is then inverted to impute missing values in the train dataset.
4. The imputed data are used to train the final neuro-fuzzy system.
5. The system is tested with complete test set.

2.1 Neuro-Fuzzy System

The neuro-fuzzy system we use in our experiments is ANNBFIS (Artificial Neural Network Based Fuzzy Inference System) [6]. It is a multiple input single output (MISO) system. Fuzzy rule base is a vital part of the system.

Rule base $\mathbb{L}$ contains fuzzy rules l (fuzzy implications). The rule's premise is built up with a fuzzy set $\mathbb{A}$ in D-dimensional space. For each dimension d the set $\mathbb{A}$ is described with the Gaussian membership function $\mu_d(x_d)$. The membership of the variable $\mathbf{x}$ to the fuzzy set $\mathbb{A}^{(l)}$ (in the lth rule is defined as a T-norm ($\star$, in ANNBFIS the product T-norm is used) of membership values of all attributes: $\mu_{\mathbb{A}^{(l)}}(\mathbf{x}) = \mu_{d_1^{(l)}}(x_1) \star \cdots \star \mu_{d_D^{(l)}}(x_D)$.

The rule's consequence is represented by a normal isosceles triangle fuzzy set $\mathbb{B}$ with the base width w. The localisation of the core of the triangle membership function is determined as a linear function of attribute values.

The output of the rule is a value of the fuzzy implication (logical interpretation of fuzzy rule): The value of the fuzzy implication is a fuzzy set. The answer of the system is an aggregation of these sets $\mu_{\mathbb{B}'}(\mathbf{x}) = \bigoplus_{l=1}^{L} \mu_{\mathbb{B}'^{(l)}}(\mathbf{x})$. In order to get a crisp answer y_0 the fuzzy set $\mathbb{B}'$ is defuzzified with modified indexed centre of gravity (MICOG) method [6]. The shape of the fuzzy set $\mathbb{B}'$ is usually quite complicated what causes expensive aggregation and defuzzyfication, but it has been proved [6] that the crisp system output can be expressed as:

$$y_0 = \frac{\sum_{l=1}^{L} g^{(l)}(\mathbf{x})\, y^{(l)}(\mathbf{x})}{\sum_{l=1}^{L} g^{(l)}(\mathbf{x})}. \tag{1}$$

The function g depends on the fuzzy implication. The Artificial Neural Network Based Fuzzy Interence System (ANNBFIS) system [6] uses Reichenbach implication.

2.2 Inversion of Fuzzy System

Inversion of a fuzzy system is widely used in automatic control. Typically a fuzzy system is provided with input values (input vector) and the task is to elaborate an answer for the presented data. In inverted paradigm desired output and input vector except one value (one attribute) are provided and the task of the inverted system is to elaborate the missing value in the input vector. There are two essential approaches to inversion: exact and approximate. The exact inversion requires special conditions of fuzzy systems to invert (zeroth order Takagi-Sugeno-Kang system triangular partition in premises [10], TSK with singletons in consequences [32], type-2 fuzzy systems with pairwise triangular function overlap in premises [17]). Approximate technique does not require special conditions for fuzzy systems. The researches proposed various methods: Levenberg-Marquadt approximate algorithm [7], genetic algorithm for iterative inversion [30], Big Bang Big Crunch (BB-BC) algorithm [16].

In our approach we apply approximate inversion of neuro-fuzzy systems with logical interpretation of rules. The Gaussian functions in premises make it impossible to invert the system analytically. Thus to invert the system we use fast Ridders algorithm [21]. It is an algorithm for finding solutions to one variable function. It has fast convergence and simple operations.

3 Experiments

The experiments were executed on real life data sets described below. Each data sets were split into train and test disjoint subsets. Each train data set was prepared in 6 version with various ratios of incomplete tuples (1, 2, 5, 10, 20, and 50 %). The data

tuple may miss at most one value. As reference methods we used marginalisation and imputation with average, median, kNN average, and kNN median ($k = 5$). For these methods first the incomplete train data sets are preprocessed, then the fuzzy models are created and finally tested with complete test sets. Because the values miss at random, for each experiment $n = 20$ incomplete train sets with values missing at random have been prepared. Statistical significance of the results was tested with dependent t-test for paired samples. Each pair holds errors elaborated with inversion and the other method. The null hypothesis states that there is no difference between results in the pair, the alternative hypothesis states that the error elaborated with inversion is lower than the one elaborated with other method (one-tailed hypothesis). At significance level $\alpha = 0.95$ the threshold of the statistics is $T(n-1, \alpha) = 1.729$, where $n = 20$.

3.1 Datasets

'Methane' The data set contains the real life measurements of air parameters (measured in 10 second intervals) in a coal mine in Upper Silesia (Poland). The task is to predict the concentration of the methane in 10 min [24]. The data set is split into train (tuples 1–511) and (512–1022) data sets.

CO_2 The dataset contains real life measurements of some air parameters (measured in 1 min intervals) in a pump deep shaft in one of Polish coal mines [23]. The task is to predict the concentration of the carbon dioxide in 10 min. The data set is split into train (tuples 1–2653) and (2654–5307) data sets.

'Concrete' It is a real life data set describing the parameters of the concrete sample and its strength [33]. The original data set can be downloaded from public repository [8]. The data set is split into train (tuples 1–515) and (516–1030) data sets.

'BoxJenkins' The popular data set describes the concentration of carbon dioxide in gas furnace [3]. The data set contains 290 data tuples prepared with template [6] $[y(n-1), \ldots, y(n-4), x(n-1), \ldots, x(n-6), y(n)]$. The data set is split into train (tuples 1–145) and (146–290) data sets.

3.2 Results

The neuro-fuzzy systems is evaluated with a root mean square error (RMSE)

$$E(\mathbb{X}) = \sqrt{\frac{1}{X}\sum_{i=1}^{X}[y_0(\mathbf{x}_i) - y(\mathbf{x}_i)]^2}, \tag{2}$$

where $\mathbb{X}$ stands for a set of data items (tuples), $X = |\mathbb{X}|$ is a number of data items, $y_0(\mathbf{x}_i)$ represents the system's answer for an ith data item $\mathbf{x}_i$ (elaborated with Eq. 1), and finally $y(\mathbf{x}_i)$ is the desired (expected) output for this data item.

For the brevity of the paper we do not present all results. The Tables 1, 2, 3, and 4 present the average error rate elaborated by the neuro-fuzzy systems with various types of preprocessing (marginalisation, imputation with average, median, kNN average, kNN median) with regard to missing ratio. The Table 5 compares the errors with regard to number of rules in the neuro-fuzzy system. The results marked with $^{(*)}$ are statistically significantly poorer than results elaborated with

Table 1 Comparison of root mean square errors (RMSE) for 'Gas Furnace' date set with regard to missing ratio (in %), neuro-fuzzy system with 5 rules

Missing ratio	Marginal	Average	Median	kNN avg.	kNN med.	Inversion
1	0.5692	0.5788	0.5783	0.5624	0.5623	0.5644
2	0.5552	0.5683	0.5614	0.5574$^{(*)}$	0.5541	0.5421
5	0.5645$^{(*)}$	0.5743$^{(*)}$	0.5683$^{(*)}$	0.5497$^{(*)}$	0.5522$^{(*)}$	0.5222
10	0.5954$^{(*)}$	0.5860$^{(*)}$	0.5921$^{(*)}$	0.5656$^{(*)}$	0.5560	0.5383
20	0.6240$^{(*)}$	0.6575$^{(*)}$	0.6354$^{(*)}$	0.5583$^{(*)}$	0.5914$^{(*)}$	0.4854
50	0.9622$^{(*)}$	0.7274$^{(*)}$	0.6989$^{(*)}$	0.6814$^{(*)}$	0.6400$^{(*)}$	0.5241

The mark $^{(*)}$ denotes results statistically poorer then results elaborated with inversion

Table 2 Comparison of root mean square errors (RMSE) for 'CO_2' date set with regard to missing ratio (in %), neuro-fuzzy system with 4 rules

Missing ratio	Marginal	Average	Median	kNN avg.	kNN med.	Inversion
1	1.0732$^{(*)}$	0.9968	1.0485	1.0368	1.0353	0.9560
2	0.9975$^{(*)}$	1.0810$^{(*)}$	1.0812$^{(*)}$	1.0856$^{(*)}$	0.9955$^{(*)}$	0.8638
5	0.9977$^{(*)}$	0.9446$^{(*)}$	1.0580$^{(*)}$	0.9754$^{(*)}$	1.0126$^{(*)}$	0.7828
10	0.9474$^{(*)}$	0.8823$^{(*)}$	0.9286$^{(*)}$	1.0670$^{(*)}$	0.9836$^{(*)}$	0.6001
20	1.0650$^{(*)}$	0.6864	0.7423	1.0482$^{(*)}$	1.0252$^{(*)}$	0.6907
50	1.1400$^{(*)}$	0.5062	0.5399	1.1165$^{(*)}$	1.0327$^{(*)}$	0.5964

The mark $^{(*)}$ denotes results statistically poorer then results elaborated with inversion

Table 3 Comparison of root mean square errors (RMSE) for 'Concrete' date set with regard to missing ratio (in %), neuro-fuzzy system with 4 rules

Missing ratio	Marginal	Average	Median	kNN avg.	kNN med.	Inversion
1	20.9261	21.0318	21.0102	20.8067	20.8395	20.8833
2	20.8836$^{(*)}$	20.8702$^{(*)}$	20.9125$^{(*)}$	20.8125$^{(*)}$	20.7739	20.6456
5	20.1917	20.7306	20.7973$^{(*)}$	20.8852$^{(*)}$	20.8292	20.1484
10	20.4786	20.5514	20.8417$^{(*)}$	20.8738$^{(*)}$	20.8232$^{(*)}$	20.4504
20	20.8643	20.8030	21.1662$^{(*)}$	20.9567$^{(*)}$	20.9847$^{(*)}$	20.4196
50	20.5521	20.6977	21.5249$^{(*)}$	20.7725	20.5672	19.5973

The mark $^{(*)}$ denotes results statistically poorer then results elaborated with inversion

Table 4 Comparison of root mean square errors (RMSE) for 'Methane' date set with regard to missing ratio (in %), neuro-fuzzy system with 3 rules

Missing ratio	Marginal	Average	Median	kNN avg.	kNN med.	Inversion
1	0.1231$^{(*)}$	0.1250$^{(*)}$	0.1246$^{(*)}$	0.1234	0.1234	0.1222
2	0.1245$^{(*)}$	0.1288$^{(*)}$	0.1281$^{(*)}$	0.1241$^{(*)}$	0.1238	0.1222
5	0.1241$^{(*)}$	0.1308$^{(*)}$	0.1316$^{(*)}$	0.1245$^{(*)}$	0.1235$^{(*)}$	0.1201
10	0.1262$^{(*)}$	0.1353$^{(*)}$	0.1321$^{(*)}$	0.1260$^{(*)}$	0.1241$^{(*)}$	0.1190
20	0.1268$^{(*)}$	0.1477$^{(*)}$	0.1433$^{(*)}$	0.1230$^{(*)}$	0.1230	0.1194
50	0.1691	0.1305	0.1616	0.1263	0.1230	0.1925

The mark $^{(*)}$ denotes results statistically poorer then results elaborated with inversion

Table 5 Comparison of root mean square errors (RMSE) for 'Gas Furnace' date set with 20 % tuples incomplete with regard to number of rules in the neuro-fuzzy system

# rules	Marginal	Average	Median	kNN avg.	kNN med.	Inversion
3	0.4865$^{(*)}$	0.5895$^{(*)}$	0.5681$^{(*)}$	0.5150$^{(*)}$	0.5426$^{(*)}$	0.4563
4	0.5738$^{(*)}$	0.6045$^{(*)}$	0.5968$^{(*)}$	0.5384$^{(*)}$	0.5697$^{(*)}$	0.4963
5	0.6239$^{(*)}$	0.6575$^{(*)}$	0.6354$^{(*)}$	0.5583$^{(*)}$	0.5914$^{(*)}$	0.4853
6	0.6561$^{(*)}$	0.6371$^{(*)}$	0.6377$^{(*)}$	0.5628	0.5876	0.5456
7	0.7012$^{(*)}$	0.6235	0.5963	0.5413	0.5550	0.5739
8	0.7476$^{(*)}$	0.6844$^{(*)}$	0.6713$^{(*)}$	0.6484$^{(*)}$	0.6810$^{(*)}$	0.5401
9	0.7589$^{(*)}$	0.6833	0.6821	0.6379	0.6640	0.6427
10	0.8596$^{(*)}$	0.7160	0.7011	0.7126	0.7102	0.6600
15	1.0614$^{(*)}$	0.8990$^{(*)}$	0.8886	1.0256$^{(*)}$	1.0242$^{(*)}$	0.8012
20	1.0048	1.0048	1.0022	1.0904$^{(*)}$	1.0827$^{(*)}$	0.9561

The mark $^{(*)}$ denotes results statistically poorer then results elaborated with inversion

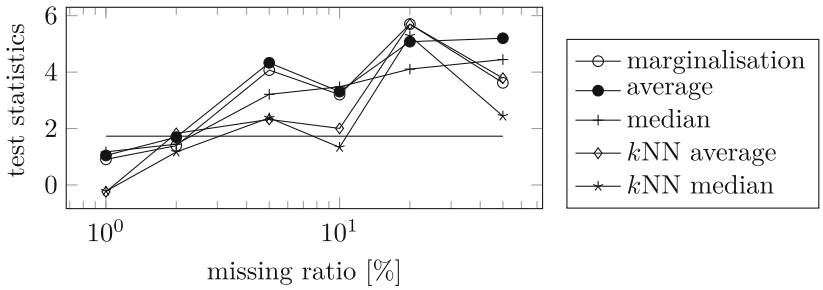


Fig. 1 The values of test statistics comparing the imputation by inversion with other techniques for 'Gas Furnace' data set and system with 5 rules. The *horizontal line* denotes the threshold value of the statistics. The results above it denote statistical significance of difference

imputation by inversion. The Figs. 1, 2, 3, and 4 present the values of test statistics with regard to missing ratio for the data sets. The horizontal line denotes the threshold value of the statistics. The points above it represent the statistical significance of

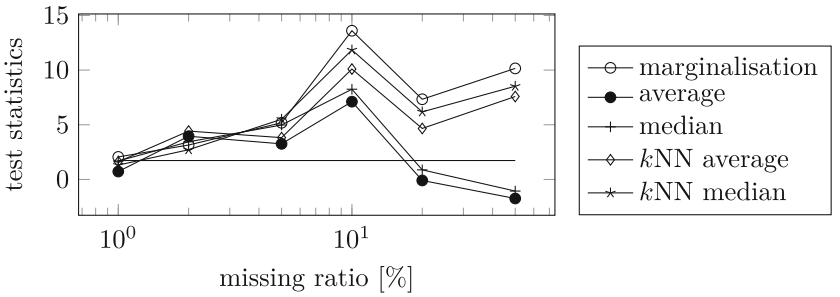


Fig. 2 The values of test statistics comparing the imputation by inversion with other techniques for 'CO_2' data set and system with 4 rules. The *horizontal line* denotes the threshold value of the statistics. The results above it denote statistical significance of difference

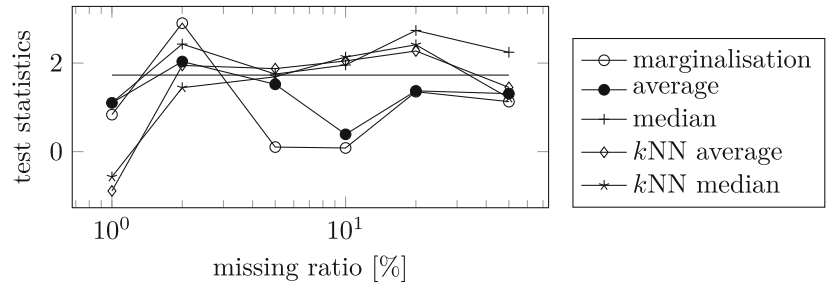


Fig. 3 The values of test statistics comparing the imputation by inversion with other techniques for 'Concrete' data set and system with 4 rules. The *horizontal line* denotes the threshold value of the statistics. The results above it denote statistical significance of difference

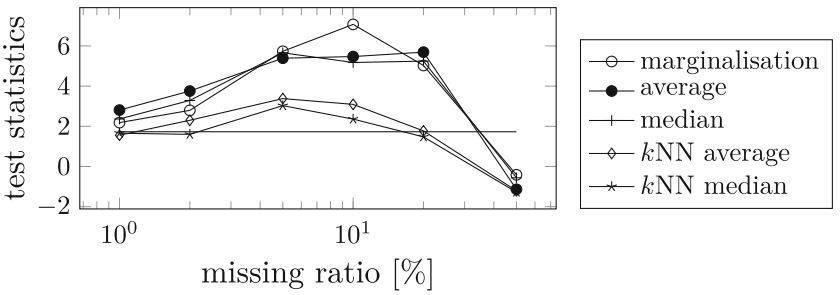


Fig. 4 The values of test statistics comparing the imputation by inversion with other techniques for 'Methane' data set and system with 3 rules. The *horizontal line* denotes the threshold value of the statistics. The results above it denote statistical significance of difference

differences between errors elaborated by systems with imputation by inversion and other methods. The Fig. 5 presents the value of the statistics with regard to number of rules in the fuzzy system.

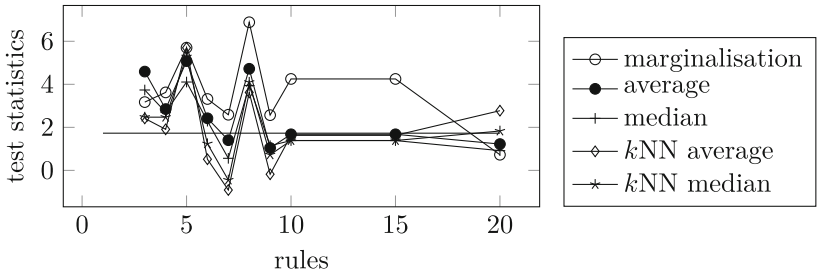


Fig. 5 The values of test statistics comparing the imputation by inversion with other techniques for 'Gas Furnace' data set and 20 % incomplete tuples. The *horizontal line*ZX denotes the threshold value of the statistics. The results above it denote statistical significance of difference

The results presented in the tables and figures mentioned above show that the proposed technique (imputation by inversion) leads to statistically significantly better results that other preprocessing techniques for data with 5, 10, and 20 % of incomplete data vectors. For almost complete data (1 % of incomplete data tuples) it is not possible to claim that this method is significantly better than other methods. Similar situation can be noticed for highly incomplete data (50 % of incomplete data).

Comparison with techniques from [22] reveals that for the 'BoxJenkins' data set imputation by inversion can elaborate lower error rates that neuro-fuzzy system with specialised clustering algorithms for incomplete data (as OCS: Optimal Completion Strategy, IFCM: Improved Fuzzy C Means [29], NPS: Nearest Prototype Stategy [13]).

The Table 5 and Fig. 5 present the typical results with regard to number of rules. It can be noticed that the proposed method leads to statistically significantly better results for neuro-fuzzy systems with low number of rules. It is a good news because neuro-fuzzy systems work with low number of rules to preserve the interpretability of the model.

4 Conclusions

Incomplete data are common and require special techniques. The essential ones are: marginalisation, imputation, and sophisticated methods. The paper presents the imputation of missing values by inversion of neuro-fuzzy system. First the neuro-fuzzy system is trained with marginalised data. Next the system is inverted and the missing values are imputed. The technique is limited to data tuples with one missing value. The complete and imputed data are used to train the final neuro-fuzzy system. The numerical experiments show that described technique is statistically significantly better results that other preprocessing techniques for data with moderate ratio of incomplete data vectors. For highly incomplete data (50 % of incomplete data) or almost complete data the technique is not statistically better.

References

1. Acuña, E., Rodriguez, C.: The treatment of missing values and its effect on classifier accuracy. In: Banks, D., McMorris, F., Arabie, P., Gaul, W. (eds.) Classification, Clustering, and Data Mining Applications, pp. 639–647. Studies in Classification, Data Analysis, and Knowledge Organisation, Springer, Berlin (2004)
2. Batista, G.E.A.P.A., Monard, M.C.: An analysis of four missing data treatment methods for supervised learning. Appl. Artif. Intell. **17**(5–6), 519–533 (2003)
3. Box, G.E.P., Jenkins, G.: Time Series Analysis Forecasting and Control. Holden-Day Incorporated, Oakland, California (1970)
4. Chen, J.Q., Xi, Y.G., Zhang, Z.J.: A clustering algorithm for fuzzy model identification. Fuzzy Sets Syst. **98**(3), 319–329 (1998)
5. Cooke, M., Green, P., Josifovski, L., Vizinho, A.: Robust automatic speech recognition with missing and unreliable acoustic data. Speech Commun. **34**, 267–285 (2001)
6. Czogala, E., Leski, J.: Fuzzy and Neuro-Fuzzy Intelligent Systems. Series in Fuzziness and Soft Computing, Physica-Verlag, A Springer-Verlag company, Heidelberg, New York (2000)
7. Filev, D.P.: Inversion of fuzzy models-practical issues. In: ICSFP, vol. 2, pp. 1658–1663. Anchorage, AK (1998)
8. Frank, A., Asuncion, A.: UCI machine learning repository (2010)
9. Gabriel, T.R., Berthold, M.R.: Missing values in fuzzy rule induction. In: SMC, vol. 2, pp. 1473–1476 (2005)
10. Galichet, S., Boukezzoula, R., Foulloy, L.: Explicit analytical formulation and exact inversion of decomposable fuzzy systems with singleton consequents. Fuzzy Sets Syst. **146**, 421–436 (2004)
11. Grzymala-Busse, J., Goodwin, L., Grzymala-Busse, W., Zheng, X.: Handling missing attribute values in preterm birth data sets. In: Slezak, D., Yao, J., Peters, J., Ziarko, W., Hu, X. (eds.) Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing. LNCS, vol. 3642, pp. 342–351. Springer, Berlin (2005)
12. Grzymala-Busse, J.W.: On the unknown attribute values in learning from examples. In: Ras, Z., Zemankova, M. (eds.) Methodologies for Intelligent Systems. LNCS, vol. 542, pp. 368–377. Springer, Berlin (1991)
13. Hathaway, R., Bezdek, J.: Fuzzy c-means clustering of incomplete data. IEEE Trans. Syst. Man Cybern. Part B: Cybern. **31**(5), 735–744 (2001)
14. Himmelspach, L., Conrad, S.: Fuzzy clustering of incomplete data based on cluster dispersion. In: Hüllermeier, E., Kruse, R., Hoffmann, F. (eds.) IPMU 2010. LNCS, vol. 6178, pp. 59–68. Springer, Berlin (2010)
15. Korytkowski, M., Nowicki, R., Scherer, R., Rutkowski, L.: Ensemble of rough-neuro-fuzzy systems for classification with missing features. In: FUZZ-IEEE, pp. 1745–1750. Hong Kong (2008)
16. Kumbasar, T., Eksin, İ., Güzelkaya, M., Yeşil, E.: Big bang big crunch optimization method based fuzzy model inversion. MICAI 2008: Advances in Artificial Intelligence. LNCS, pp. 737–740. Springer, Berlin (2008)
17. Kumbasar, T., Eksin, İ., Güzelkaya, M., Yeşil, E.: Exact inversion of decomposable interval type-2 fuzzy logic systems. Int. J. Approximate Reasoning **54**, 253–272 (2013)
18. Matyja, A., Simiński, K.: Comparison of algorithms for clustering incomplete data. Found. Comput. Decis. Sci. **39**(2), 107–127 (2014)
19. Mundfrom, D.J., Whitcomb, A.: Imputing missing values: the effect on the accuracy of classification. Multiple Linear Regres. Viewpoints **25**(1), 13–19 (1998)
20. Nowicki, R.K.: Rough-neuro-fuzzy structures for classification with missing data. IEEE Trans. Syst. Man Cybern. Part B: Cybern. **39**(6), 1334–1347 (2009)
21. Ridders, C.: A new algorithm for computing a single root of a real continuous function. IEEE Trans. Circ. Syst. **26**, 979–980 (1979)
22. Sikora, M., Simiński, K.: Comparison of incomplete data handling techniques for neuro-fuzzy systems. Comput. Sci. **15**(4), 441–458 (2014)

23. Sikora, M., Krzykawski, D.: Application of data exploration methods in analysis of carbon dioxide emission in hard-coal mines dewater pump stations. Mechanizacja i Automatyzacja Gornictwa **413**(6) (2005)
24. Sikora, M., Sikora, B.: Application of machine learning for prediction a methane concentration in a coal-mine. Arch. Min. Sci. **51**(4), 475–492 (2006)
25. Simiński, K.: Neuro-rough-fuzzy approach for regression modelling from missing data. Int. J. Appl. Math. Comput. Sci. **22**(2), 461–476 (2012)
26. Simiński, K.: Clustering with missing values. Fundamenta Informaticae **123**(3), 331–350 (2013)
27. Simiński, K.: Rough fuzzy subspace clustering for data with missing values. Comput. Inf. **33**(1), 131–153 (2014)
28. Simiński, K.: Rough subspace neuro-fuzzy system. Fuzzy Sets Syst. **269**, 30–46 (2015)
29. Timm, H., Kruse, R.: Fuzzy cluster analysis with missing values. In: NAFIPS, pp. 242–246. Pensacola Beach, FL (1998)
30. Varkonyi-Koczy, A., Almos, A., Kovacshazy, T.: Genetic algorithms in fuzzy model inversion. In: FUZZ-IEEE, vol. 3, pp. 1421–1426 (1999)
31. Wagstaff, K.L., Laidler, V.G.: Making the most of missing values: object clustering with partial data in astronomy. In: ADASS XIV, vol. 347, pp. 172–176. Pasadena, California, USA (2005)
32. Xu, C., Shin, Y.: A fuzzy inverse model construction method for general monotonic multi-input-single-output (MISO) systems. IEEE Trans. Fuzzy Syst. **16**(5), 1216–1231 (2008)
33. Yeh, I.C.: Modeling of strength of high-performance concrete using artificial neural networks. Cement Concr. Res. **28**(12), 1797–1808 (1998)
34. Zhang, S.: Parimputation: from imputation and null-imputation to partially imputation. IEEE Intell. Inf. Bull. **9**(1), 32–38 (2008)

Memetic Neuro-Fuzzy System with Big-Bang-Big-Crunch Optimisation

Krzysztof Siminski

Abstract The paper presents a memetic fuzzy inference system based on Big Bang Big Crunch (evolutionary optimisation) and gradient descent (local search) techniques. Tuning parameters of the fuzzy system with evolutionary optimisation failed to be successful, but application of both evolutionary and local optimisation achieved lower error rates than reference system (that uses only gradient descent optimisation). The results of experiments have been statistically verified.

Keywords Approximate inversion · Imputation · Incomplete data · Neuro-fuzzy system

1 Introduction

Neuro-fuzzy systems are commonly known for their ability to extract knowledge from the presented data ("neuro" part) and to model the imprecision of the data ("fuzzy" part). Extraction of knowledge results in fuzzy model. Extraction can be divided into two parts: identification of the model structure (number of rules, used attributes, etc.) and identification of parameter values. Many techniques have been proposed for creation of models as grid partition, clustering, hierarchical partition [14, 18], simulated annealing [10], genetic algorithms [2, 3]. Memetic genetic programming fuzzy inference system (MEMFIS) [19] applies backpropagation gradient descent for linguistic variables and least square for coefficients of the linear functions. The genetic algorithm uses a context-free grammar guidance. The paper [8] describes application of genetic algorithm to identification of structure of fuzzy system and compares this technique with LOLIMOT [14] algorithm (Local Linear Model Tree(s)).

The phrase *memetic algorithm* was coined in late 80 s of the 20th century. One of the first algorithms was a hybrid of a genetic algorithm and a simulated annealing technique. Evolutionary algorithms mimic the biological optimisation technique.

K.Siminski (✉)
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: krzysztof.siminski@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_50

This biological background is then joined with mathematical optimising techniques. The memetic algorithms can be summarized as adding local search optimising technique to an evolutionary algorithm. The memetic algorithms are sometimes called [9] hybrid genetic algorithms, genetic local search algorithms, Lamarckian evolutionary algorithm. The last name is a tribute to the Lamarckian theory than claimed that acquired characteristics can be inherited by the offsprings. The more intensively a giraffe stretches its neck, the longer necks have its calves. This phenomenon is not observed in nature, but the idea can be mimicked in computers.

The memetic algorithms are widely used to find solutions to NP problems (graph partitioning, travelling salesman problem, vehicle routing problem [12], parallel machine scheduling, timetable scheduling [1], etc.), machine learning [13], automatic control, and robotics, molecular optimisation, image processing [5], pattern recognition etc.

The novelty of the paper is an application of memetic optimisation to tuning parameters of a neuro-fuzzy system with logical interpretation of fuzzy rules.

The paper is organized as follows: Sect. 2.1 describes shortly the neuro-fuzzy system. Sect. 2.2 outlines main features of Big-Bang-Big-Crunch algorithm. Sect. 3 presents the experiments and finally Sect. 4 summarizes the paper.

2 Memetic Fuzzy Inference System

The memetic fuzzy inference system is based on fuzzy systems with logical interpretation of fuzzy rules. The general pattern of the training is following:

1. model identification: clustering of data
2. extraction of rules from the clusters
3. evolutionary tuning of model parameters
4. local gradient descent optimisation
5. repetition of steps 3-4

In step 1 the fuzzy c-mean (FCM) algorithm [6] is used for identification of cluster centres. Then these centres are transformed into premises of the rules (step 2). This initial fuzzy model is then multiplicated into the population of individuals. Then the evolutionary algorithm (step 3) is started. Then for each individual the local search is started (step 4). The evolutionary and local search optimisation are then repeated.

2.1 Neuro-Fuzzy System

The neuro-fuzzy system we use in our experiments is ANNBFIS (Artificial Neural Network Based Fuzzy Inference System) [4]. It is a multiple input single output (MISO) system. The crucial part of the systems is fuzzy rule base. Each fuzzy rule is a fuzzy logical implication. The value of the rule is the value of the fuzzy implication of

rule's premise and consequence. The premises have Gaussian membership, premises have triangular symmetric functions.

Rule base $\mathbb{L}$ contains fuzzy rules l (fuzzy implications)

$$l : \mathbf{x} \text{ is } \mathfrak{a} \rightsquigarrow y \text{ is } \mathfrak{b}, \tag{1}$$

where $\mathbf{x} = [x_1, x_2, \ldots, x_D]^{\mathrm{T}}$ and y are linguistic variables, $\mathfrak{a}$ and $\mathfrak{b}$ are fuzzy linguistic values. The squiggle arrow ($\rightsquigarrow$) stands for fuzzy implication.

The rule's premise is built up with a fuzzy set $\mathbb{A}$ in D-dimensional space. For each dimension d the set $\mathbb{A}$ is described with the Gaussian membership function:

$$\mu_d(x_d) = \exp\left(-\frac{(x_d - v_d)^2}{2s_d^2}\right), \tag{2}$$

where v_d is the core location for dth attribute and s_d is this attribute's deviation. The membership of the variable $\mathbf{x}$ to the fuzzy set $\mathbb{A}^{(l)}$ (in the lth rule is defined as a T-norm of membership values of all attributes:

$$\mu_{\mathbb{A}^{(l)}}(\mathbf{x}) = \mu_{d_1^{(l)}}(x_1) \star \cdots \star \mu_{d_D^{(l)}}(x_D) = \bigstar_{d \in \mathbb{D}} \mu_{d^{(l)}}(x_d), \tag{3}$$

where $\star$ denotes the T-norm and $\mathbb{D}$ stands for the set of attributes (in ANNBFIS the product T-norm is used).

The term $\mathfrak{b}$ (in the rule's consequence) is represented by an normal isosceles triangle fuzzy set $\mathbb{B}$ with the base width w. The localisation of the core of the triangle membership function is determined as a linear function of attribute values:

$$y^{(l)}(\mathbf{x}) = \left(\mathbf{p}^{(l)}\right)^{\mathrm{T}} \cdot \left[1, \mathbf{x}^{\mathrm{T}}\right]^{\mathrm{T}} = \left[p_0^{(l)}, p_1^{(l)}, \ldots, p_D^{(l)}\right] \cdot [1, x_1, \ldots, x_D]^{\mathrm{T}}, \tag{4}$$

where $\mathbf{p}^{(l)}$ is the parameter vector of the consequence for lth rule. The localisation of the fuzzy sets depends on the values of the input vector $\mathbf{x}$.

The output of the rule is a value of the fuzzy implication (logical interpretation of fuzzy rule).

$$\mu_{\mathbb{B}'^{(l)}}(\mathbf{x}) = \mu_{\mathbb{A}^{(l)}}(\mathbf{x}) \rightsquigarrow \mu_{\mathbb{B}^{(l)}}(\mathbf{x}). \tag{5}$$

The answer of the system is the aggregation of all rules:

$$\mu_{\mathbb{B}'}(\mathbf{x}) = \bigoplus_{l=1}^{L} \mu_{\mathbb{B}'^{(l)}}(\mathbf{x}). \tag{6}$$

In order to get a crisp answer y_0 the fuzzy set $\mathbb{B}'$ is defuzzified with modified indexed centre of gravity MICOG method [4]. The shape of the fuzzy set $\mathbb{B}'$ is usually quite complicated what causes expensive aggregation and defuzzification, but it has been

proved [4] that the crisp system output can be expressed as:

$$y_0 = \frac{\sum_{l=1}^{L} g^{(l)}(\mathbf{x})\, y^{(l)}(\mathbf{x})}{\sum_{l=1}^{L} g^{(l)}(\mathbf{x})}. \tag{7}$$

The function g depends on the fuzzy implication. The Artificial Neural Network Based Fuzzy Interence System (ANNBFIS) system [4] uses Reichenbach implication [15].

In local search optimisation parameters of the premises are tuned with backpropagation gradient descent method. The parameters of the linear models in conclusions are calculated with least square linear regression. The widths of the supports w of sets in consequences are tuned with backpropagation gradient descent method.

2.2 Big Bang Big Crunch Algorithm

Big-Bang-Big-Crunch (BBBC) algorithm [7] is a heuristics for optimisation. It is based on evolutionary approach. The algorithm has two phases in each iteration (generation): Big Bang and Big Crunch. The Big Bang phase creates randomly the population of individuals all over the search space. The Big Crunch is a convergence phase that shrinks all individuals into one representative. In original BBBC the representative is the 'center of mass' of all individuals in the population. The 'mass' is the inverse of the fitness function. For the next generation (starting with Big Bang phase) the new population has to be created. The new population is randomly selected from the neighbourhood of the representative of the previous Big Crunch phase. In order to maintain the global optimisation a certain ratio of individuals is taken from the whole search space. This ratio diminishes with number of generations.

In our implementation the ratio of item taken from the whole domain starts with 0.9. In each generation it is multiplied by 0.9, so it decreases, but never equals zero. Also in each generation the radius of neighbourhood decreases.

3 Experiments

The memetic neufo-fuzzy system first identifies the structure of the model. In experiments fuzzy model have $L = 6$ rules. In our experiments we do not aim at finding the best number of rules in the fuzzy model. We use constant number of rules. The identification of structure uses FCM algorithm [6] with 100 iterations.

For identification of parameters the population of individuals is created. Each individual represents one fuzzy model. The BBBC algorithm elaborated the best individual that is later tuned with gradient descent. The iteration of global (BBBC)—local (gradient) optimisation is called generation.

The reference system is an Artificial Neural Network Based Fuzzy Inference System (ANNBFIS)—fuzzy inference system with logical interpretation of rules. Each data set is split into train and test data sets. These data sets have no common part. The experiments were executed in two paradigms: data approximation (DA) and knowledge generalisation (KG). In DA the systems are trained and tested with the same train data set. In KG the systems are trained with train data set and tested with test data sets (unseen cases).

3.1 Data Sets

'**Methane**' The data set contains the real life measurements of air parameters in a coal mine in Upper Silesia (Poland). The parameters (measured in 10 s intervals) are: AN31—the flow of air in the shaft, AN32—the flow of air in the adjacent shaft, MM32—concentration of methane (CH_4), production of coal, the day of week. To the tuples the 10-minute sums of measurements of AN31, AN32, MM32 are added as dynamic attributes [17]. The task is to predict the methane concentration in 10 minute interval. The data is divided into train set (499 tuples) and test set (523 tuples).

'**Mackey-Glass**' The data sets represents chaotic time series generated by differential equation [11]:

$$\frac{dx}{dt}(t) = \frac{ax(t-\tau)}{1+x^{10}(t-\tau)} - bx(t), \tag{8}$$

where $a = 0.2$, $b = 0.1$ and $\tau = 17$. The data tuples are created with template

$$\mathbf{x} = \left[x\left(t_{-8}\right), x\left(t_{-7}\right), \ldots x\left(t_0\right), x\left(t_1\right)\right]. \tag{9}$$

The last item of the tuple $x\left(t_1\right)$ is the value predicted on the base of previous 9 items. The train data set is composed by 200 tuples $x(501) - x(700)$, the test set comprises tuples $x(701) - x(1000)$.

'$\mathbf{CO_2}$' The dataset contains real life measurements of some air parameters in a pump deep shaft in one of Polish coal mines. The parameters (measured in 1 minute intervals) are: CO_2—concentration of carbon dioxide, Ps—atmospheric pressure, RHOs—relative humidity of the air in the shaft, RHPs—relative humidity of the air near the pump, TOs—air temperature. The dynamic attributes (10-minute sums of measurements: DCO_2, DPs, DRHOs, DRHPs, DTOs) are added to the tuples. The task is to predict the concentration of the carbon dioxide in 10 minutes. The data are divided into train set (tuples 1–2653) and test set (tuples 2654–5307) [16].

Table 1 The results elaborated by ANNBFIS and mememtic BBBC NFS systems for 'Methane' data set

ANNBFIS				
DA		0.293842 ± 0.313697		
KG		0.538277 ± 0.357462		
Memetic BBBC NFS				
Number of individuals				
g		5	10	20
5	t	DA	DA	DA
	0	$177.634916 \pm 83.945770^{(*)}$	$108.325359 \pm 42.081164^{(*)}$	$77.039364 \pm 28.604570^{(*)}$
	5	$0.087386 \pm 0.000455^{(*)}$	$0.086709 \pm 0.000626^{(*)}$	$0.086796 \pm 0.000480^{(*)}$
	10	$0.086759 \pm 0.000609^{(*)}$	$0.086395 \pm 0.000575^{(*)}$	$0.086150 \pm 0.000861^{(*)}$
	20	$0.086174 \pm 0.000750^{(*)}$	$0.085913 \pm 0.000464^{(*)}$	$0.085294 \pm 0.000756^{(*)}$
	t	KG	KG	KG
	0	$222.945385 \pm 124.179084^{(*)}$	$291.779016 \pm 277.775444^{(*)}$	$186.143893 \pm 111.050100^{(*)}$
	5	$0.122174 \pm 0.018388^{(*)}$	$0.159853 \pm 0.041923^{(*)}$	$0.158456 \pm 0.053130^{(*)}$
	10	$0.121552 \pm 0.012671^{(*)}$	$0.138429 \pm 0.034359^{(*)}$	$0.124190 \pm 0.015648^{(*)}$
	20	$0.128870 \pm 0.014461^{(*)}$	$0.140134 \pm 0.028357^{(*)}$	$0.150247 \pm 0.029821^{(*)}$
10	t	DA	DA	DA
	0	$149.190853 \pm 36.475603^{(*)}$	$104.450647 \pm 41.700906^{(*)}$	$86.035442 \pm 28.965690^{(*)}$
	5	$0.086682 \pm 0.000520^{(*)}$	$0.086150 \pm 0.000653^{(*)}$	$0.086080 \pm 0.000652^{(*)}$
	10	$0.086208 \pm 0.000697^{(*)}$	$0.086219 \pm 0.000554^{(*)}$	$0.085780 \pm 0.000662^{(*)}$
	20	$0.085794 \pm 0.000661^{(*)}$	$0.085504 \pm 0.000810^{(*)}$	$0.085004 \pm 0.000629^{(*)}$
	t	KG	KG	KG
	0	$153.391511 \pm 100.505104^{(*)}$	$154.310243 \pm 68.031192^{(*)}$	$151.774478 \pm 114.144159^{(*)}$
	5	$0.125296 \pm 0.019454^{(*)}$	$0.128133 \pm 0.014175^{(*)}$	$0.186229 \pm 0.181251^{(*)}$
	10	$0.139198 \pm 0.032109^{(*)}$	$0.129102 \pm 0.016499^{(*)}$	$0.139591 \pm 0.039754^{(*)}$
	20	$0.134361 \pm 0.032139^{(*)}$	$0.132749 \pm 0.023820^{(*)}$	$0.127950 \pm 0.012156^{(*)}$
20	t	DA	DA	DA
	0	$73.659512 \pm 33.413561^{(*)}$	$75.351433 \pm 20.296595^{(*)}$	$58.585661 \pm 23.425983^{(*)}$
	5	$0.086437 \pm 0.001093^{(*)}$	$0.086235 \pm 0.000658^{(*)}$	$0.085754 \pm 0.000830^{(*)}$
	10	$0.086083 \pm 0.000804^{(*)}$	$0.086039 \pm 0.000543^{(*)}$	$0.085237 \pm 0.001251^{(*)}$
	20	$0.085669 \pm 0.000698^{(*)}$	$0.085491 \pm 0.001121^{(*)}$	$0.085059 \pm 0.000628^{(*)}$
	t	KG	KG	KG
	0	$146.276680 \pm 137.001866^{(*)}$	$173.487981 \pm 151.423942^{(*)}$	$123.187450 \pm 132.355318^{(*)}$
	5	$0.128336 \pm 0.021921^{(*)}$	$0.121239 \pm 0.012682^{(*)}$	$0.126753 \pm 0.016312^{(*)}$
	10	$0.122475 \pm 0.015563^{(*)}$	$0.134653 \pm 0.017374^{(*)}$	$0.135910 \pm 0.023069^{(*)}$
	20	$0.147295 \pm 0.031700^{(*)}$	$0.132629 \pm 0.016408^{(*)}$	$0.134578 \pm 0.028662^{(*)}$

Each experiment was repeated $n = 10$ times. The table holds the averages and standard deviations of the root mean square error (RMSE) for DA and KG paradigms. The notation $\mu \pm \sigma$ stand for average and standard deviation, g stands for number of generations, t stands for number of tuning iterations in memetic local search, symbol $^{(*)}$ denotes statistically significantly better results than those elaborated by ANNBFIS

3.2 Results

In statistical testing we use the Cochran-Cox test. We assume that the averages of RMSE elaborated by ANNBFIS (subscript $_1$) and proposed approach (subscript $_2$) have normal distributions $N(\mu_1, \sigma_1)$ i $N(\mu_2, \sigma_2)$ with unknown standard deviations σ_1 and σ_2. The null hypothesis states that the averages are equal: $H_0 : \mu_1 = \mu_2$.

Table 2 The results elaborated by ANNBFIS and mememtic BBBC NFS systems for 'Mackey-Glass' data set

ANNBFIS				
DA		0.863128 ± 0.275646		
KG		0.865955 ± 0.276285		
Memetic BBBC NFS				
Number of individuals				
g		5	10	20
5	t	DA	DA	DA
	0	$0.396090 \pm 0.115698^{(*)}$	$0.373270 \pm 0.065580^{(*)}$	$0.274305 \pm 0.071636^{(*)}$
	5	$0.000190 \pm 0.000000^{(*)}$	$0.000188 \pm 0.000004^{(*)}$	$0.000187 \pm 0.000006^{(*)}$
	10	$0.000190 \pm 0.000000^{(*)}$	$0.000190 \pm 0.000000^{(*)}$	$0.000190 \pm 0.000000^{(*)}$
	20	$0.292693 \pm 0.433951^{(*)}$	$0.011121 \pm 0.032793^{(*)}$	$0.000190 \pm 0.000000^{(*)}$
	t	KG	KG	KG
	0	$0.391608 \pm 0.119544^{(*)}$	$0.379342 \pm 0.069132^{(*)}$	$0.277960 \pm 0.073161^{(*)}$
	5	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$	$0.000179 \pm 0.000003^{(*)}$
	10	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$
	20	$0.294000 \pm 0.435134^{(*)}$	$0.012198 \pm 0.036051^{(*)}$	$0.000180 \pm 0.000000^{(*)}$
10	t	DA	DA	DA
	0	$0.291264 \pm 0.053009^{(*)}$	$0.282456 \pm 0.085875^{(*)}$	$0.238051 \pm 0.021195^{(*)}$
	5	$0.000189 \pm 0.000003^{(*)}$	$0.000190 \pm 0.000000^{(*)}$	$0.000184 \pm 0.000005^{(*)}$
	10	$0.000190 \pm 0.000000^{(*)}$	$0.000190 \pm 0.000000^{(*)}$	$0.000190 \pm 0.000000^{(*)}$
	20	$0.095672 \pm 0.286446^{(*)}$	$0.000190 \pm 0.000000^{(*)}$	$0.000190 \pm 0.000000^{(*)}$
	t	KG	KG	KG
	0	$0.288977 \pm 0.053060^{(*)}$	$0.280642 \pm 0.081879^{(*)}$	$0.236471 \pm 0.026710^{(*)}$
	5	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$	$0.000179 \pm 0.000005^{(*)}$
	10	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$
	20	$0.095967 \pm 0.287361^{(*)}$	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$
20	t	DA	DA	DA
	0	$0.000189 \pm 0.000003^{(*)}$	$0.000187 \pm 0.000005^{(*)}$	$0.000188 \pm 0.000004^{(*)}$
	5	$0.000189 \pm 0.000003^{(*)}$	$0.000187 \pm 0.000005^{(*)}$	$0.000188 \pm 0.000004^{(*)}$
	10	$0.000190 \pm 0.000000^{(*)}$	$0.000190 \pm 0.000000^{(*)}$	$0.000190 \pm 0.000000^{(*)}$
	20	$0.000190 \pm 0.000000^{(*)}$	$0.000190 \pm 0.000000^{(*)}$	$0.000190 \pm 0.000000^{(*)}$
	t	KG	KG	KG
	0	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$	$0.000181 \pm 0.000007^{(*)}$
	5	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$	$0.000181 \pm 0.000007^{(*)}$
	10	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$
	20	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$	$0.000180 \pm 0.000000^{(*)}$

Each experiment was repeated $n = 10$ times. The table holds the averages and standard deviations of the root mean square error (RMSE) for DA and KG paradigms. The notation $\mu \pm \sigma$ stand for average and standard deviation, g stands for number of generations, t stands for number of tuning iterations in memetic local search, symbol $^{(*)}$ denotes statistically significantly better results than those elaborated by ANNBFIS

Table 3 The results elaborated by ANNBFIS and mememtic BBBC NFS systems for 'CO_2' data set

ANNBFIS				
DA		$1.996250 \pm 2.220446 \cdot 10^{-16}$		
KG		$1.437843 \pm 4.582576 \cdot 10^{-6}$		
Memetic BBBC NFS				
Number of individuals				
g		5	10	20
5	t	DA	DA	DA
	0	$869.150111 \pm 322.795806^{(*)}$	$700.284266 \pm 438.377436^{(*)}$	$326.971723 \pm 132.864212^{(*)}$
	5	$0.448547 \pm 0.007374^{(*)}$	$0.437251 \pm 0.007521^{(*)}$	$0.428424 \pm 0.009808^{(*)}$
	10	$0.445802 \pm 0.012510^{(*)}$	$0.436952 \pm 0.009721^{(*)}$	$0.433476 \pm 0.008498^{(*)}$
	20	$0.445592 \pm 0.006471^{(*)}$	$0.439342 \pm 0.012347^{(*)}$	$0.427798 \pm 0.014001^{(*)}$
	t	KG	KG	KG
	0	$739.864647 \pm 411.062665^{(*)}$	$640.567645 \pm 423.316846^{(*)}$	$380.725436 \pm 280.127754^{(*)}$
	5	$0.293922 \pm 0.073710^{(*)}$	$0.263820 \pm 0.033492^{(*)}$	$0.500412 \pm 0.615798^{(*)}$
	10	$0.326969 \pm 0.130471^{(*)}$	$0.293039 \pm 0.182352^{(*)}$	$0.239589 \pm 0.016691^{(*)}$
	20	$0.240517 \pm 0.021726^{(*)}$	$0.287596 \pm 0.052693^{(*)}$	$0.317454 \pm 0.150329^{(*)}$
10	t	DA	DA	DA
	0	$761.141745 \pm 462.113554^{(*)}$	$448.461913 \pm 258.019972^{(*)}$	$382.140964 \pm 214.351591^{(*)}$
	5	$0.431601 \pm 0.008825^{(*)}$	$0.433031 \pm 0.007660^{(*)}$	$0.429413 \pm 0.006995^{(*)}$
	10	$0.448662 \pm 0.014646^{(*)}$	$0.435524 \pm 0.006983^{(*)}$	$0.426259 \pm 0.012191^{(*)}$
	20	$0.438805 \pm 0.011802^{(*)}$	$0.429682 \pm 0.016127^{(*)}$	$0.422714 \pm 0.007612^{(*)}$
	t	KG	KG	KG
	0	$678.248728 \pm 517.222434^{(*)}$	$397.491552 \pm 390.135759^{(*)}$	$348.335782 \pm 327.473318^{(*)}$
	5	$0.293250 \pm 0.100369^{(*)}$	$0.255574 \pm 0.045647^{(*)}$	$0.289932 \pm 0.150674^{(*)}$
	10	$0.248512 \pm 0.025484^{(*)}$	$0.401160 \pm 0.383967^{(*)}$	$0.261042 \pm 0.050629^{(*)}$
	20	$0.320489 \pm 0.144001^{(*)}$	$0.442849 \pm 0.426908^{(*)}$	$0.241791 \pm 0.023383^{(*)}$
20	t	DA	DA	DA
	0	$709.185201 \pm 364.450670^{(*)}$	$253.200967 \pm 139.999364^{(*)}$	$220.369890 \pm 48.345527^{(*)}$
	5	$0.436218 \pm 0.011490^{(*)}$	$0.432581 \pm 0.009851^{(*)}$	$0.419521 \pm 0.009096^{(*)}$
	10	$0.440210 \pm 0.009881^{(*)}$	$0.429594 \pm 0.008036^{(*)}$	$0.425722 \pm 0.008002^{(*)}$
	20	$0.432299 \pm 0.011355^{(*)}$	$0.416614 \pm 0.009155^{(*)}$	$0.423023 \pm 0.009819^{(*)}$
	t	KG	KG	KG
	0	$548.671734 \pm 352.225822^{(*)}$	$209.128162 \pm 153.660934^{(*)}$	$164.275872 \pm 105.844956^{(*)}$
	5	$0.302179 \pm 0.166924^{(*)}$	$0.290029 \pm 0.051890^{(*)}$	$0.281745 \pm 0.103065^{(*)}$
	10	$0.463412 \pm 0.441842^{(*)}$	$0.294566 \pm 0.148971^{(*)}$	$0.327736 \pm 0.269785^{(*)}$
	20	$0.289729 \pm 0.097846^{(*)}$	$0.272017 \pm 0.077856^{(*)}$	$0.339797 \pm 0.146052^{(*)}$

Each experiment was repeated $n = 10$ times. The table holds the averages and standard deviations of the root mean square error (RMSE) for DA and KG paradigms. The notation $\mu \pm \sigma$ stand for average and standard deviation, g stands for number of generations, t stands for number of tuning iterations in memetic local search, symbol $^{(*)}$ denotes statistically significantly better results than those elaborated by ANNBFIS

The one-tailed alternative hypothesis states that the error elaborated by ANNBFIS is greater: $H_1 : \mu_1 > \mu_2$. The significance level $1 - \alpha = 0.95$ leads to critical interval $[1.812, +\infty)$.

The results are gathered in Tables 1 ('Methane'), 2 ('Mackey-Glass'), and 3 ('CO_2'). The symbol $^{(*)}$ denotes statistically significantly better results elaborated by memetic approach than by ANNBFIS.

The first important conclusion is that the evolutionary approach (BBBC algorithm) without local search does not lead to smaller errors than reference systems (ANNBFIS). Intertwining the evolutionary algorithm with local search (memetic approach) elaborates lower errors than reference system with gradient descent optimisation.

The important conclusion is that the number of generations is more important factor than number of individuals in each generation. To achieve lower errors it is better to increase number of generations than number of individuals in generation. These conclusions are valid both for knowledge generalisation (KG) and data approximation (DA) paradigms.

4 Conclusions

The paper presents a memetic fuzzy inference system based on Big-Bang-Big-Crunch (global evolutionary optimisation) and gradient descent (local search) techniques. Tuning parameters of the fuzzy system with evolutionary optimisation failed to be successful, but application of both evolutionary and local optimisation achieved lower error rates than reference system (that uses only gradient descent optimisation). Because the evolutionary technique applies random values the experiments were repeated and statistically verified. The important conclusion from the experiments is that the number of generations is more important than number of individuals in each generations.

References

1. Abbaszadeh, M., Saeedvand, S., Mayani, H.A.: Solving university scheduling problem with a memetic algorithm. Int. J. Artif. Intell. **1**(2), 79–90 (2012)
2. Cordón, O., Herrera, F.: Identification of linguistic fuzzy models by means of genetic algorithms. In: Hellendoorn, H., Driankov, D. (eds.) Fuzzy model Identification, pp. 215–250. Springer, Berlin (1997)
3. Cordón, O., Herrera, F.: A three-stage evolutionary process for learning descriptive and approximate fuzzy-logic-controller knowledge bases from examples. Int. J. Approx. Reason. **17**(4), 369–407 (1997)
4. Czogala, E., Leski, J.: Fuzzy and neuro-fuzzy intelligent systems. Series in fuzziness and soft computing. Physica-Verlag, A Springer-Verlag Company, Heidelberg, New York (2000)
5. Di Gesu, V., Lo Bosco, G., Millonzi, F., Valenti, C.: A memetic algorithm for binary image reconstruction. In: Brimkov, V.E., Barneva, R.P., Hauptman, H.A. (eds.) Combinatorial Image Analysis, LNCS, vol. 4958, pp. 384–395. Springer, Berlin Heidelberg (2008)
6. Dunn, J.C.: A fuzzy relative of the ISODATA process and its use in detecting compact, well separated clusters. J. Cybern. **3**(3), 32–57 (1973)
7. Erol, O.K., Eksin, I.: A new optimization method: Big Bang-Big Crunch. Adv. Eng. Softw. **37**(2), 106–111 (2006)
8. Hoffmann, F., Nelles, O.: Genetic programming for model selection of TSK-fuzzy systems. Inf. Sci. **136**(1), 7–28 (2001)

9. Krasnogor, N., Aragón, A., Pacheco, J.: Memetic algorithms. In: Alba, E., Marti, R. (eds.) Metaheuristic procedures for training neutral networks. Operations Research/Computer Science Interfaces Series, vol. 36, pp. 225–248. Springer, US (2006)
10. Leski, J., Czogala, E.: A neuro-fuzzy inference system optimized by deterministic annealing. In: Hampel, R., Wagenknecht, M., Chaker, N. (eds.) Fuzzy Control, Advances in Soft Computing, vol. 6, pp. 287–293. Physica-Verlag HD (2000)
11. Mackey, M.C., Glass, L.: Oscillation and chaos in physiological control systems. Science **197**(4300), 287–289 (1977)
12. Nalepa, J., Blocho, M.: Adaptive memetic algorithm for minimizing distance in the vehicle routing problem with time windows. Soft Comput. 1–19 (2015)
13. Nalepa, J., Kawulok, M.: A memetic algorithm to select training data for support vector machines. In: GECCO 2014. pp. 573–580. Vancouver, Canada (2014)
14. Nelles, O., Fink, A., Babuška, R., Setnes, M.: Comparison of two construction algorithms for Takagi-Sugeno fuzzy models. Int. J. Math. Comput. Sci. **10**(4), 835–855 (2000)
15. Reichenbach, H.: Wahrscheinlichkeitslogik. Erkenntnis **5**, 37–43 (1935)
16. Sikora, M., Krzykawski, D.: Application of data exploration methods in analysis of carbon dioxide emission in hard-coal mines dewater pump stations. Mechanizacja i Automatyzacja Gornictwa 413(6) (2005)
17. Sikora, M., Krzystanek, Z., Bojko, B., Śpiechowicz, K.: Application of a hybrid method of machine learning for description and on-line estimation of methane hazard in mine workings. J. Min. Sci. **47**(4), 493–505 (2011)
18. Siminski, K.: Patchwork neuro-fuzzy system with hierarchical domain partition. In: Kurzyński, M., Woźniak, M. (eds.) Computer recognition systems 3, advances in intelligent and soft computing, vol. 57, pp. 11–18. Springer-Verlag, Berlin, Heidelberg (2009)
19. Tsakonas, A.: Local and global optimization for Takagi-Sugeno fuzzy system by memetic genetic programming. Expert Syst. Appl. **40**, 3282–3298 (2013)

Part X
Algorithms and Optimisation

Statistical Methods of Natural Language Processing on GPU

Dariusz Banasiak

Abstract The following work investigates the subject of using GPGPU technology for natural language processing. Natural language processing involves analysing very large volumes of data based on sophisticated algorithms. This process can only be performed on computers with significant computing power. Parallel computing and utilisation of the processing capacity of graphics cards can help achieve the above requirements. The work presents the problem of building n-gram models of natural language based on specific text. Two algorithms were developed: a sequential one for a typical CPU and a parallel one, which uses the capacity of a GPU. The GPU algorithm was prepared using Nvidia CUDA technology. Experiments were carried out in order to compare the effectiveness of the developed algorithms depending on the size of the analysed text and the number of words in the n-grams. The results showed that a parallel type algorithm is better for a GPU environment.

Keywords Natural language processing · N-grams · GPGPU · CUDA

1 Introduction

Parallel programming has recently become one of the more important ideas in modern information technology. Previously used methods of increasing the capacity of computer systems based on single core architecture have practically achieved the limit of its potential. It proved difficulty to break the barrier of 5 GHz, when increasing processor clocks. It is also difficult to further improve the degree of task parallelism achieved through the use of pipeline architecture. The situation is similarly difficult in the area of data bus width. Therefore, the only way to potentially improve the performance of computer systems is to use methods based on multiprocessing. As a result, the currently manufactured CPUs actually consist of a number of processors (called cores) in one chip. All the cores are identical and they communicate one

D. Banasiak (✉)
Department of Computer Engineering, Wroclaw University of Technology,
Wroclaw, Poland
e-mail: dariusz.banasiak@pwr.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_51

with another by means of a shared, internal system bus. Depending on the utilised architecture, the typical contemporary CPUs can have from 2 to 16 cores (certainly there are CPUs with more cores).

Multiprocessing is also used in modern graphics cards. The process of creating realistic graphics in computer games is associated with fast processing large amounts of data. As a result video adapters must have high computing power. This requirement was met by using parallel architecture. Modern graphics cards typically have several hundred or even several thousand cores. Consequently, processing power of the Graphics Processing Unit (GPU) is a number of times higher than the capacity of a traditional CPU (even if it has more than one core). It was therefore natural, that attempts were made to utilise GPUs for computing that was not related to image generation. Major GPU manufacturers, i.e. Nvidia and ATI supported this trend and created tools which facilitate using graphics cards for general-purpose computing. Nvidia's CUDA environment is particularly interesting in this context [1, 5].

GPU could potentially be used for Natural Language Processing (NLP). Natural language processing involves analysis of very large volumes of data based on sophisticated algorithms. This process can only be performed on computers with significant computing power. Literature provides an increasing number of studies illustrating the use of GPUs to solve selected NLP problems. Speech recognition is a suitable example (Nvidia provides CUFFT library with a Fast Fourier Transform (FFT) implementation for CUDA environment [6]). Additional application examples include: morphological analysis [7], lexical scan and text segmentation [2, 8], text analysis using regular expressions [4] or context-free grammars [9].

This work focuses on the problem of creating n-gram models of natural language based on sample texts. N-grams make it possible to represent contextual knowledge by determining incidence of specific word sequences. They can be used in systems for speech recognition, text correction etc.

2 Comparison of CPU and GPU Architectures

Graphics Processing Unit is the main element of every video adapter. Initially, GPUs were specialised circuits that were intended to support CPUs only in the area of computer graphics. They featured fixed graphics pipeline functionality. Increasing requirements related to graphics processing resulted in more powerful video adapters. Consequently, computing capacity of GPUs became higher than that of CPUs (even the ones with several cores). Introduction of Nvidia's GeForce 3 Series cards in 2001 was crucial in the context of using GPUs for parallel computing. They were the first units, that in accordance with DirectX 8.0 requirements, allowed programming of pixel and vertex shaders. Since then, graphics cards were increasingly utilised for general-purpose computing.

The architecture of GPUs is different from that of a traditional CPUs. CPUs are used for sequential processing of large volumes of data. CPUs must provide advanced control flow functionalities, that allow frequent use of conditional

Table 1 The comparison of program running time (in seconds) in a CPU and GPU environment

N	1	10	100	500	1000	5000	10,000	50,000
CPU	0.002	0.017	0.185	0.965	1.926	9.664	19.377	96.352
GPU	0.099	0.189	0.270	0.322	0.577	0.587	1.162	4.035

statements and loops. CPU should have full access to system memory. Initially, CPUs were designed based on Single Instruction, Single Data (SISD) architecture. Modern processing units tend to be multi-core. This means that Multiple Instruction, Multiple Data (MIMD) architecture is more frequent.

GPU is a specialised unit realising functions related to graphics processing. The method of generating image in a graphics card (performing the same instruction sequence on many data streams) makes it necessary to design units optimised for running as many threads simultaneously as possible. GPUs are, therefore, an example of SIMD architecture. Instruction sequence run on data is usually predefined, so control flow management is not that important in GPUs. High processing output of GPUs results from the fact, they consist of a large number of processing units, called cores (e.g. Tesla K20X has 2688 cores).

The difference between programs intended for CPU and GPU environments is well illustrated by the following example. Let's analyse the incidence of particular words in a text (let N be the number of those words). In the most basic version of the algorithm, the procedure determining the number of occurrences of a given word in a text has to be executed N times. On a CPU the program will be run in a sequential manner: procedures for individual words will be executed one after another. This means that the higher the N value, the longer the program running time. On a GPU the program will be run in a parallel manner. Occurrences of individual words can be counted in separate threads running in parallel. Increase of the N value does not necessarily mean longer program running time. Table 1 shows the comparison of program running times for a CPU and GPU version depending on the value of N.

3 Statistical Methods in Natural Language Processing

Statistical approach to natural language processing means analysing large amounts of data (e.g. text corpora). Currently, statistical methods are often used in NLP systems, because they generate good results and significantly reduce the required human input. N-grams are one of the methods used in natural language processing systems. They are created as a result of statistical analysis of suitably large language data assets, which are called corpora. N grams are mainly used to predict the next element in a sequence (e.g. a phoneme during speech recognition or a word during text analysis). The process of creating an n-gram model starts by counting the number of occurrences of element sequences (phonemes, words, etc.) that have a specific length N in the

analysed language assets. Sequence length N is usually one (1-grams, unigrams), two (2 grams, bigrams) and three (3-grams, trigrams). The process of creating longer n-grams requires very large amounts of language data, which is not always possible.

In order to create an n-gram model of a natural language, one must define the occurrence probability for a specific word sequence (especially for full sentences). Let $w = w_1, w_2, \ldots, w_n$ be a sequence consisting of n words, and $p(w_1, w_2, \ldots, w_n)$ be the occurrence probability for this sequence. If Bayes' law is used (also called chain rule), occurrence probability for sequence w can be calculated using the following formula:

$$p(w_1, w_2, \ldots, w_n) = p(w_1)p(w_2|w_1)p(w_3|w_1, w_2) \ldots p(w_n|w_1, \ldots, w_{n-1}) \tag{1}$$

Probability $p(w_2|w_1)$ in the formula (1) is the conditional probability that word w_2 will occur, if the preceding word is w_1 (the left-hand side context). Similarly, probability $p(w_n|w_1, \ldots, w_{n-1})$ is the occurrence probability for word w_n on condition that the preceding word sequence was $w_1, \ldots, w_{n-1}$. Probability $p(w_n|w_1, \ldots, w_{n-1})$ can, for example, be determined by calculating incidence of the analysed word sequences in a given text corpus. Let $c(w_1, w_2, \ldots, w_n)$ be the number of occurrences of sequence $w_1, w_2, \ldots, w_n$ in a corpus. Then, probability $p(w_n|w_1, \ldots, w_{n-1})$ can be estimated using the following formula:

$$p(w_n|w_1, \ldots, w_{n-1}) = \frac{c(w_1, w_2, \ldots, w_n)}{c(w_1, w_2, \ldots, w_{n-1})} \tag{2}$$

In practice, it is impossible to determine probability for a context of any length. First, very large corpora are needed for long word sequences. Even if those were available it is not certain that a given word sequence will be found. Second, using context that is too long may be inadvisable. In case of very long sentences, it may be important to consider only local context for a given word.

As a result, certain approximation of the probability defined by formula (1) is used. Only $N - 1$ elements of the left-hand side context are taken into consideration. The probability, defined as described above, is an approximation of Markov chain of the order of $N - 1$ and is called an n-gram language model. Then, the following dependence is used to calculate formula (1):

$$p(w_n|w_1, \ldots, w_{n-1}) \approx p(w_n|w_{n-N+1}, \ldots, w_{n-1}) \tag{3}$$

Table 2 shows a fragment of data acquired during Berkeley Restaurant project [3]. The table shows incidence of bigrams, created by mutually combining 8 sample words, in an available corpus. The corpus was created on the basis of 9332 sentences in the form of questions about a restaurant in Berkeley. The corpus included 1446 various word forms. Words at the beginning of a row are the first element of a bigram. Words at the beginning of a column are the second element a bigram. Based on the above information we can verify that "I want" occurred 827 times in that corpus.

Table 2 Incidence of selected bigrams in the Berkeley Restaurant corpus [3]

	I	Want	To	Eat	Chinese	Food	Lunch	Spend	
I	5	827	0	9	0	0	0	2	2533
Want	2	0	608	1	6	6	5	1	927
To	2	0	4	686	2	0	6	211	2417
Eat	0	0	2	0	16	2	42	0	746
Chinese	1	0	0	0	0	82	1	0	158
Food	15	0	15	0	1	4	0	0	1093
Lunch	2	0	0	0	0	1	0	0	341
Spend	1	0	1	0	0	0	0	0	278

Zero means a bigram was not present in the corpus. The last column is a number of occurrences in the corpus for a word that begins a bigram (value 2533 in the last column of the second row is the number of occurrences of the word "I" in the analysed corpus).

4 Description of the Algorithms Used in Experiment

The primary aim of this work is to compare the effectiveness of algorithms (implemented in CPU and GPU environments) that determine the incidence of n-grams (with preset N) in a given text.

Literature provides examples of many approaches to building an n-gram model. One of the more typical ones is based on the assumption that all fundamental elements that will be used to build n-grams (alphabet) are known. Depending on the intended use of the system, alphabet can be defined by single letters (e.g. in a speech recognition system) or by words of a given language (e.g. in a text analysis system). Then, m^N size matrix is created, where m is the number of alphabet elements and N is the size of the n-gram. An example of such a matrix for $N = 2$ is shown in Table 2. Individual elements of the matrix represent the number of occurrences of a given n-gram in a text; e.g. position [2, 4] corresponds to "to eat" bigram. Unfortunately, this solution has two drawbacks. First, for high m and N values it requires large memory size (e.g. building a model for English alphabet describing sequences consisting of 10 letters requires memory for $26^{10} \approx 1,4 \times 10^{14}$ matrix elements). Second, a lot of matrix elements take a value of 0 (it is not possible to combine some of the elements of the alphabet). Work [4] suggests an effective method creating an n-gram model for high N values, that eliminates the above limitations.

This paper presents a different approach to building an n-gram model. Let's introduce the following definitions: N is the number of words comprising an n-gram (n-gram size), WORDS_NUM is the number of words in a text, POS is the position (index) of a word in a text, WORD(POS) is a word in a text, the position of

which is POS, CURR_NGRAM is a sequence of N successive words in a text starting from position POS, and NGRAMS_TAB is the structure that stores the n-grams found in a text. NGRAMS_TAB consists of two tables. The first one contains all the non-repeating n-grams in a text, and the second one the number of occurrences of a given n-gram in the text. NGRAMS_TAB is dynamically created (it starts with no elements). The operation of an algorithm building an n-gram model in a CPU environment can be described as follows:

```
1. POS = 0;
2. while (POS ≤ WORDS_NUM − N)
3.   CURR_NGRAM = WORDS(POS) & .. &WORDS(POS+N−1)
4.   IF (CURR_NGRAM is in NGRAMS_TAB)
5.     increase by 1 counter for CURR_NGRAM in NGRAMS_TAB
6.   ELSE
7.     add CURR_NGRAM to NGRAMS_TAB
8.     set at 1 counter for CURR_NGRAM in NGRAMS_TAB
```

Symbol & used in the description indicates concatenation (combining several character chains into one chain).

The algorithm that counts n-grams in a GPU environment should take full advantage of the available computing potential. This can be achieved through parallel computing. The easiest method of achieving algorithm parallelism is to make separate threads count individual n-grams. If a text consists of n words, $n - N + 1$ threads should be used. A single thread should count the number of occurrences of a single n-gram in a text. If the index variable is the thread number, the n-gram will consist of the words located from index to index $+ N - 1$ in a text. Since a given n-gram can exist in a text several times, a number of threads can count the number of occurrences of the same n-gram. This is not optimal. In order to reduce this effect (it cannot be eliminated completely), the following modifications were introduced into the kernel function:

- if an n-gram, for which the thread was executed is positioned at index in the text, it is compared to succeeding n-grams, starting from index $+ 1$,
- if an n-gram, for which the thread was executed was found before in the text, the thread will be stopped (this is achieved, using an auxiliary table).

In the described case, the correct number of occurrences of a given n-gram is determined by a thread that counted the first occurrence of it. Other thread results for the same n-gram (if started) will be ignored. Initial experiments show that the total time of building a n-gram model in a GPU environment is mainly affected by the computing time for the stage of determining the number of occurences of individual n-grams in text. The time required for other stages (e.g. allocation of data structures, transfer data from CPU to GPU) is inconsequential.

5 Experiment Results

The aim of the experiments was to compare the effectiveness of algorithms (implemented in CPU and GPU environments) that create n gram models. The following hardware environment was used during the experiment:

1. CPU: AMD FX(tm)-4100 (4 cores, f = 3.60 GHz),
2. graphics card: Nvidia GeForce GTX 560.

The GPU available on the graphics adapter is characterised by the following parameters: the number of CUDA cores—336, GPU core clock—1,66 GHz, memory clock—2,004 GHz, maximum number of threads in a block—1024, number of threads allowed for every block dimension—1024 × 1024 × 64, number of blocks allowed for every grid dimension —65535 × 65535 × 65535, compute capability—2.1.

Algorithm test were performed on texts of various length. Digital versions of books in Polish and English were used during experiments. The basic data on the analysed texts are presented in Table 3.

The first phase of the experiments was to determine the effect the number of concurrently running threads has on the performance of the algorithm in the GPU environment. Earlier experiments showed that it is an important parameter that significantly influences the total time required to run an algorithm. The experiments were performed for all the 7 texts, and the following N values were assumed as n-gram size: 2, 3 and 4. The results for N = 2 are shown in Table 4. The results for N = 3 and N = 4 are similar.

Figure 1 shows the results for N = 2 in a graphical form. As expected, the time required to create an n-gram does not decrease, if the number of running threads exceeds a some value. Therefore, it was decided that 1024 blocks and 1024 threads would be running during future experiments.

The fundamental aim of the performed experiments was to compare algorithms (implemented in CPU and GPU environments) that determine the n-gram model

Table 3 Basic data on the texts used during experiments

text	Number of characters	Total number of words	Number of various words	Number of various 2 grams	Number of various 3 grams	Number of various 4 grams
t1	92,780	16,343	3254	11,159	14,820	15,672
t2	207,702	36,177	4350	21,332	32,238	35,099
t3	605,541	120,219	8493	52,513	97,551	114,315
t4	1,058,488	159,992	28,362	112,389	150,903	158,540
t5	1,573,153	272,772	33,962	165,950	248,304	267,644
t6	1,935,559	260,251	45,940	193,412	247,555	257,536
t7	3,440,425	447,208	56,387	298,249	412,162	438,415

Table 4 Time required to create n-grams (N = 2), depending on the number of blocks and the number of threads in a block [in seconds]

text	1/ 512	1/ 1024	4/ 1024	8/ 1024	16/ 1024	64/ 1024	512/ 1024	1024/ 1024	4096/ 1024
t1	0.26	0.23	0.07	0.04	0.04	0.04	0.04	0.04	0.04
t2	1.31	1.13	0.31	0.25	0.20	0.18	0.18	0.18	0.18
t3	13.55	11.80	3.06	2.83	2.22	1.86	1.78	1.78	1.78
t4	24.80	20.73	5.31	5.04	3.87	3.22	3.05	3.05	3.05
t5	73.00	61.37	15.51	15.14	11.56	9.69	9.07	9.06	9.07
t6	63.52	52.50	13.36	12.89	9.83	8.22	7.70	7.70	7.70
t7	182.80	151.30	38.03	37.53	28.53	23.89	22.27	22.26	22.27

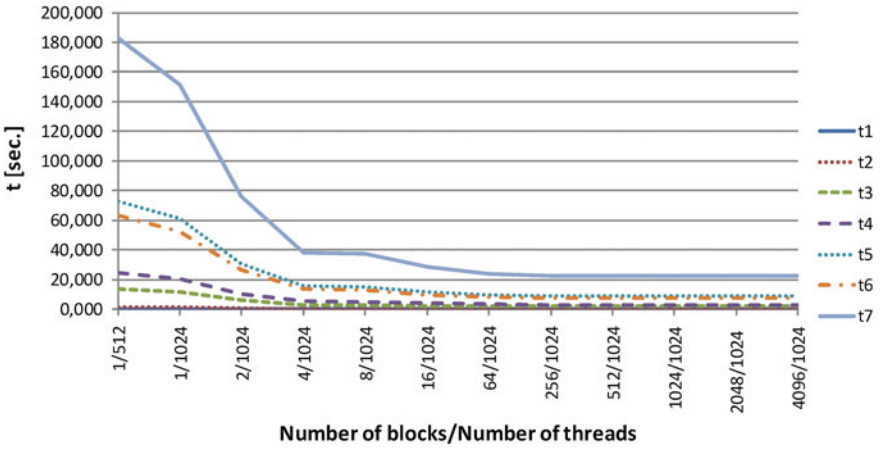


Fig. 1 Time required to create an n-gram (for N = 2), depending on the number of blocks and the number of threads in a block

based on a given text. Tests were run for three different n-gram sizes (N value of 2, 3 and 4). The obtained results are shown in Table 5.

The table only presents time for the phase of building the n-gram model, i.e. isolating individual n-grams from a text, and determining the number of occurrences of each of those n-grams in the analysed text. The times required to read data from a file and write the results to a file were omitted. Additionally, in the case of the GPU algorithm, the time required to allocate memory on the GPU and copy data from the CPU memory to the GPU memory was omitted.

The obtained results lead to the following conclusions. First, implementing the algorithm in a GPU environment results in much faster computing. For example, the time required to execute the algorithm in the GPU environment for the longest text and N = 4 is over 100 times shorter than it is in the CPU environment. Additionally, the algorithm execution time in the CPU environment for the same text depends on

Table 5 Time required to create an n-gram, depending on the used algorithm [in seconds]

N	Alg.	t1	t2	t3	t4	t5	t6	t7
2	CPU	0.568	2.191	19.076	95.243	244.396	408.871	1109.20
	GPU	0.037	0.175	1.779	3.051	9.065	7.704	22.263
3	CPU	0.865	4.395	67.130	171.429	609.792	651.977	2033.96
	GPU	0.036	0.177	1.926	3.084	9.362	7.679	22.582
4	CPU	0.931	5.739	89.578	204.936	743.713	722.729	2325.43
	GPU	0.035	0.172	1.900	2.990	9.054	7.395	21.819

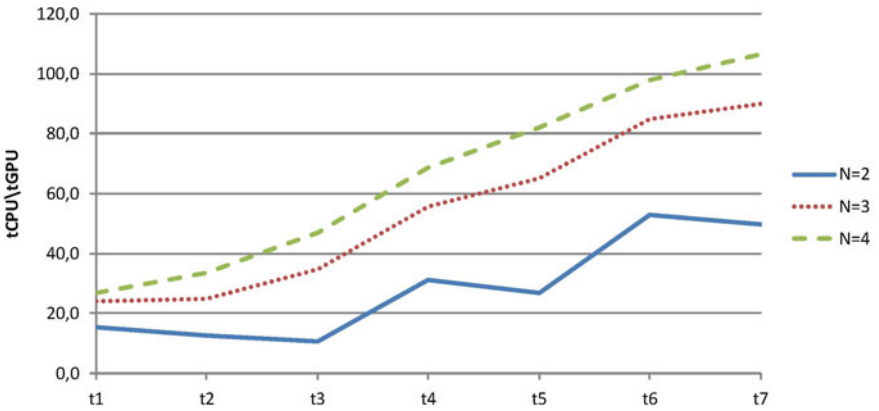


Fig. 2 The relation of tCPU/tGPU times for various texts and N values

the value of N. This dependence is not present for the algorithm implemented in the GPU environment (execution time for a given text does not depend on the value of N).

It can be noticed that the advantage of the GPU in relation to the CPU increases with more complex problems (longer texts or higher N values). It is well illustrated in the graph in Fig. 2. It shows algorithm execution time quotient tCPU in the CPU environment and algorithm execution time tGPU in the GPU environment in relation to the size of text and N value.

6 Conclusions

This paper covers the problem of building an n-grams model based on an analysed text. Two algorithms were developed: a serial one for a traditional CPU and a parallel one for a GPU. The algorithms were tested for performance on various length texts for a number of n gram sizes (for N value of 2, 3 and 4). The performed experiments confirmed suitability of the GPGPU technology for problems related to natural

language processing. If the size of n-grams was equal to 4 and the longest text was analysed, the GPU version of algorithm was over 100 times faster than the CPU version. Additionally, it was found that the times required to run auxiliary operations (reading and writing data to a file, memory allocation and data copying) can be omitted in the context of total time needed for creating an n-gram.

The experiments and results are a useful starting point for further study related to using GPGPU technology for natural language processing.

Acknowledgments This work was financed by Ministry of Science and Higher Education in Poland (research project no. N N516 499139).

References

1. Ghorpade, J., Parande, J., Kulkarni, M., Bawaskar, A.: GPGPU processing in CUDA architecture. Adv. Comput.: Int. J. **3**(1), 105–120 (2012)
2. Gupta, S., Rajasekhara, M.B.: Performance analysis of GPU compared to single-core and multi-core CPU for natural language applications. Int. J. Adv. Comput. Sci. Appl. **2**(5), 50–53 (2011)
3. Jurafsky, D., Martin, J.H.: Speech and Language Processing. Pearson Prentice Hall, New Jersey (2008)
4. Nagao, M., Mori, S.: A new method of N-gram statistics for large number of n and automatic extraction of words and phrases from large text data of Japanese. In: COLING 1994. vol. 1, pp. 611–615. Kyoto, Japan (1994)
5. NVidia: CUDA C Programming Guide ver. 5.0 (2012)
6. NVidia: CUFFT Library User Guide ver. 5.0 (2012)
7. Shiwon, C., Dong-Wook, L.: High-performance Korean morphological analyzer using the mapreduce framework on the GPU. J. Electr. Eng. Technol. **6**(4), 573–579 (2011)
8. Xiwu, G., Ruixuan, L., Kunmei, W., Bei, P., Weijun, X.: A GPU-based accelerator for Chinese word segmentation. In: Sheng, Q.Z., Wang, G., Jensen, C.S., Xu, G. (eds.) Web Technologies and Applications. LNCS, vol. 7235, pp. 231–242. Springer, Berlin (2012)
9. Youngmin, Y., Chao-Yue, L., Slav, P., Keutzer, K.: Efficient parallel CKY parsing on GPUs. In: IWPT 2011. pp. 175–185. Dublin, Ireland (2011)

Profitability Analysis of PV Installation in Combination with Different Time-of-Use Strategies in Poland

Agnieszka Brachman and Robert Wojcicki

Abstract Increasing the participation of energy originated from renewable sources is one of the most important issues for polish and EU economy. It is also required by all targets of the 2020 climate and energy package for EU. The usage of micro photovoltaic (PV) instalments for residential and commercial buildings in Poland has been increasing for several years. Introduction of the Renewable Energy Sources Act, will raise profitability of this type of installations even more. The paper presents an energetic and economical analysis of exemplary PV system for domestic production, under polish climatic and economic conditions. Moreover an overview of DSR strategies is presented, some of them are already available for customers, others require additional regulations or technology development.

Keywords Photovoltaic · Smart grid · Solar electricity · Demand response · Electricity market

1 Introduction

Over 70 % of all residential buildings in Poland, use coal as the main source of heat. Since several years, the public organizations emphasize the negative influence of such instalments for environment and the overall comfort of living. The problem of low emission induced by coal furnaces is the main cause for toxic substances contained in the air, which results in higher cardiac and pulmonary diseases among society. Heating and cooling respond for 58 % of final energy consumed in Poland, therefore they represent the largest potential to energy savings, carbon reduction and development of renewable energy sources [5].

Increasing the participation of energy originated from renewable sources is one of the most important issues for polish and EU economy. It is also required by all

A. Brachman (✉) · R. Wojcicki
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: agnieszka.brachman@polsl.pl

R. Wojcicki
e-mail: robert.wojcicki@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_52

targets of the 2020 climate and energy package for EU. These targets known as the 20–20–20 targets set the three key objectives for 2020, connected with:

- the reduction of greenhouse gas emission (20 %),
- raising the share of energy consumption produced from renewable sources (20 %),
- improvement in the energy efficiency (20 %).

Undeniable advantage of renewable sources are the minor influence on natural environment resulting in the reduction of the low emission. The commonly accessible renewable sources of energy, applicable for residential and smaller commercial purposes, include: biogas (originating from sewage treatment plants, landfills, etc.), water, solar radiation and wind power. Current regulations allow both: production of the electricity for self needs and selling back the surplus to the grid to the owners' benefit.

The most simply accessible instalments are systems that use the solar radiation. The usage of micro photovoltaic (PV) instalments for residential and commercial buildings in Poland has been increasing for several years. Introduction of the Renewable Energy Sources Act [6], which is already scheduled, will raise profitability of this type of installations even more. The key objective of the aforementioned act is guaranteeing the price of the bought back kilowatt-hour [kWh] for next 15 years. The purchase price depends on the energy source and the overall power of the installed system, nevertheless it is higher then the current retail price, regardless on the customer's tariff.

The main flaw of the PV systems is the incongruity that exists between the timing of electricity generated from the PV installation and the peak demand hours [2]. At households, the peak electricity demand occurs usually in the evening and the highest production from PV installation is at midday hours when solar irradiation is the highest [1]. Moreover, the energy produced during winter, from PV installation laying on the latitude specific to polish region, is insufficient for the heating purposes.

Providing high system reliability in power grid is serious and complicated concern [8]. Power disturbances and quality issues cause huge material losses to other industries. Due to the growing demand and limited supply, the energy price rates are growing. Alongside, residential consumers seek for opportunities to minimize their own energy consumption as well as benefit to the most from their own renewable energy sources. It seems that the best approach to maintain and improve the energy usage curve, is to allow the consumers to actively participate (directly or indirectly) in the electricity market. It can be achieved by variable electricity tariffs or other incentives [3, 4, 8].

Demand Side Response (DSR) programs are intended to allow consumers actively participate in the electricity market. The DSR strategies are supposed to influence the consumption patterns in response to fluctuations in the electricity prices over time or incentive payments, which affects the overall energy consumption and may shift and/or flatten peak loads. The main potential of DSR comes for daily and seasonal fluctuations in requirements for power. User reaction is induced by different strategies, through applying financial benefits or penalties for energy consumption during and off peak periods. Determination of peak and off-peak hours may be

dynamic and should induce lower electricity use at times of high wholesale market prices or when system reliability is jeopardized. These problems are addressed by may EU projects, e.g. [7].

The paper presents an overview of several DSR strategies, some of them are already available for customers, others require additional regulations or technology development. Section 2 provides detailed summary of different types of DSR programs. In Sect. 3 exemplary, residential PV system is presented alongside its performance evaluation with reference to the household energy usage. Section 4 presents discussion of economical benefits of PV installation when applying different DSR strategies. Finally Sect. 5 concludes the paper.

2 Classification of DSR Strategies

Every DSR strategy concerns both managing and influencing the demand side i.e. end-user in power grid. DSR concerns also the energy providers since provider and consumer must cooperate at the energy consumption level and the way it is consumed. The main benefits from facilitating the DSR strategies would be:

- Reduction of maximum peak load, which usually last a few hours, when the price of energy is extremely high due to unexpected events such as blackouts, destruction of the power lines, unexpected and excessive requirements;
- Increasing the load when the energy price is low (among others—in order to correct the power coefficient);
- Shifting the power consumption from peak hours and seasons;
- Fitting the power consumption to current efficiency of the energy production system;
- Reducing the overall energy consumption.

There are two main categories of DSR schemes: Incentive-Based Programs (IBP) and Price-Based Programs [8]. The detailed programs are as follows:

- Incentive-Based Programs
  - Direct Load Control (DLC),
  - Interruptible/Curtailable Rates (ICR),
  - Demand Bidding Programs (DBP),
  - Emergency Demand Response Programs (EDRP),
  - Capacity Market Programs (CMP),
  - Ancillary services market programs (ASMP).
- Price-Based Programs
  - Time-of-Use (TOU),
  - Critical-Peak Pricing (CPP),
  - Real-Time Pricing (RTP).

Application of Incentive Based Programs requires interaction between end user and energy provider. The consumer voluntarily agrees to participate in certain schemes by allowing the operator to control some of the electric appliances. Since Incentive Based Programs are not available for residential customers in Poland, details of the particular programs are not provided and they can be found in [4, 8]

In general all Price-Based programs are based on dynamic pricing rates. For energy providers, the main purpose is to flatten the demand curve by offering higher prices during peak load periods and lower prices during off-peak periods. The Price-Based programs doesn't require the interaction between the operator and consumer. The participant voluntarily adjusts the energy consumption basing on the fixed or variable rates. The operator believes that variable price rates will sway users to power conservation in peak periods and potential energy deficits.

Time-of-Use (TOU) The TOU program is based on several fixed rates for different periods of day, weak and/or season of the year. The tariffs are set a priori and are not efficient for temporary fluctuations in energy supplies, however they force users to decreasing their energy consumption in periods with the highest price rates. The higher differences among price rates during peak and off-peak periods, the higher reduction of energy consumption during peak periods. Those tariffs are willingly and commonly applied by private end users, especially if they use heat pumps or accumulation heating or other aggregators. Because of their simplicity, they don't require sophisticated systems or any other smart, additional appliances.

Critical-Peak Pricing (CPP) The CPP program introduces one or more very high price rates for peak load periods, when the price of energy at the wholesale market is the highest. The participants are informed with short advance that there will be rate revaluation. The scheme may be an addition to the Time-of-Use program which allows reaction to the dynamic changes in the energy market. This program requires short reaction time between the information concerning the incoming price increase and expected user reaction, therefore introducing this program requires suitable IT system and smart controlling of home appliances.

Real-Time Pricing (RTP) assumes dynamic variations in price rates basing on the relationship between supply and demand. The energy price rate would change similarly to the wholesale market prices. The participants are informed about the energy price rates on hour-ahead to day-ahead basis. Applying this scheme requires similar IT system as in the previous case. The time duration between sending price rate information and actual price may change. The RTP program is highly suitable if renewable energy sources are present since they cause high variations in the energy production process. The program is also believed to be the most direct and efficient in terms of proper energy curve management, for competitive electricity markets.

DSR programs are supposed to reduce the disadvantages of uneven and sometimes high energy consumption [3]. Lowering energy peaks, reduces the mean costs of energy provisioning which results from higher efficiency of power production and distribution. DSR schemes are an answer to the energy supply problems, especially when considering the potential capabilities of the smart grids. In Poland, there are many different strategies available for different types of customers. Residential user

may choose between two TOU tariffs, namely G11 and G12 and their variations. Details are provided in Sect. 4.

3 PV System Analysis

3.1 PV System Configuration

For the purpose of the further analysis of benefits when applying the demand response scheme, the following PV microgrid installation was considered: Eighteen monocrystal, silicon panels gathered in two sets, distributed in two different south-oriented surfaces of residential building: the first set containing eleven panels, is placed on south-east surface, tilt degrees 30° and output power 2695 W_p, the second set containing seven panels, placed on south-west surface of the roof, tilt degrees of 45° with power 1715 W_p. The overall installation size is 4.4 kW_p, which is around the most popular installation size in many single family households. For the heating purposes of the analysed building, the heat pump is used. The installation is presented in Fig. 1.

Solar system is designed to produce energy for a single family household and to supply the generated electricity to the grid. Each PV collector has an aperture area of 1.2 m^2. The PV collector used for the means of this study is a commercially available product. The design parameters are shown in Table 1.

There are two energy meters for registering the produced and utilized energy. Moreover there is system for PV monitoring consisting of two bi-directional energy meters provided by the grid operator. Some additional equipment was installed to monitor the PV production, household energy usage and some environmental monitoring (outside and inside temperature).

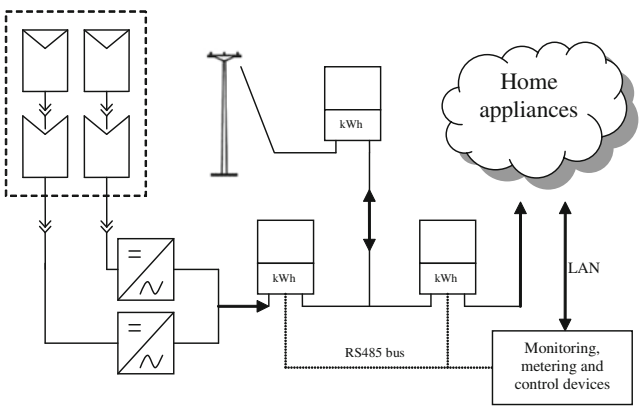


Fig. 1 The PV system configuration

Table 1 PV system design parameters

Parameter	1st set	2nd set
Collector aperture area (m^2)		
Nominal electric power of the system (W_p)	2695	1715
Array tilt angle (°)	30	45

Table 2 Photovoltaic system production summary

Month	Demand (kWh)	PV supply (kWh)	PV usage (kWh)	Peak (kWh)	Off-peak (kWh)
May 14′	436	379	177	71	188
June	384	521	211	53	120
July	451	548	229	70	151
August	411	422	182	76	153
September	409	338	157	98	154
October	461	269	141	108	213
November	668	122	77	102	489
December	1016	72	51	138	826
January 15’	1047	78	58	130	859
February	904	201	146	112	647
March	733	359	260	93	379
April	626	449	259	80	287
All	7546	3757	1949	1131	4466

3.2 PV System Efficiency

The Table 2 presents the summary of the demanded, produced and utilized electricity, in the analysed, residential building from May 2014 to April 2015. During this period the PV system produced 3757 kWh of electricity, however only 51 % on average was used for self demands, i.e. 1949 kWh and the remaining 1807 kWh was sold back to the grid. In the analysed period 5597 kWh was taken from the power grid, which makes the overall consumption 7546 kWh. From the 5597 kWh of bought energy, 1131 kWh was bought during peak hours and the rest off peak hours. Increased consumption can be observed since October, when the heating season began.

These data indicate the difficulties with the current use of electricity for self needs, resulting in the need to buy the additional energy from the grid, despite the daily and monthly overproduction from the PV system. This problem stems both from the manufacturing source of instability photovoltaic and lack of coordination between production and energy consumption as well as no smart controlling appliances, that would increase the ratio of electricity consumed for own needs for electricity taken from the network. The main conclusion is that PV energy cannot be fully utilized

for self demands, which results in using grid electricity despite daily and monthly surpluses coming from PV installation. The main reasons for this are instability of photovoltaic source, no coordination among PV supply and current demands, and first and most of all—no control of home appliances, which could improve the percentage of self-produced electricity. The efficiency evaluation of the depicted PV system and analysis under different DSR strategies are presented in the next section.

4 DRM Strategies Analysis in Context of Home Installation

Analysing Incentive Based programs is hard due to specific parameters of these tariffs and their influence on the consumer behaviour, therefore in further analysis the Price-based in combination to Time-based strategies were analysed. In Poland, there are two schemes (tariffs) for individual customers: G11 and G12. G11 sets fixed price for kWh for the whole day during the whole week. The G12 tariff is the example of Time-of-Use strategy, and it sets peak hours from 6 a.m. to 1 p.m. and 3 p.m. to 10 p.m on weekdays. There are different prices for kWh depending on the region of Poland, however the differences among the available tariffs are similar, therefore further analysis is provided for prices for the south region of Poland, where the presented PV system is installed.

4.1 G11 Tariff—No DSR Strategy

The annual demand for energy in analysed period of time was 7546 kWh. The overall production of PV system was 3757 kWh, which is almost 50 % of the total demand, however only 1949 kWh was used for self needs, which is only 25 % of the total demand. If the presented system was used without any pricing program the annual costs of the system would be price for 5597 kWh, the current price for kWh is 0.4979, which is overall 2787 PLN.

The data presented in Table 2 show that the use of electricity from PV system is around 50 % on average, but despite greater demand for energy in individual months, up to 51 % of the electricity was sold back to the grid. Similar proportions for their own consumption are reported in other studies.

Despite the daily electricity deficits, some part of the produced electricity is always sold back to the grid. The largest part of the production was returned to the grid in the summer and reached 58 %, while in the winter the input to the grid was only around 26–29% of energy produced, as shown in Fig. 2.

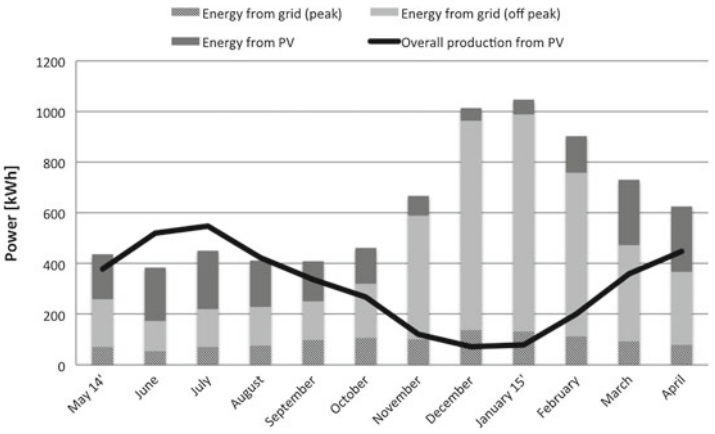


Fig. 2 The total electricity usage originating from grid and PV system, in comparison to the overall PV system production

4.2 G12 Tariff—TOU Program

For the analysed installation the overall energy taken from grid was 5597 kWh, from which 4466 kWh was taken during off-peak hours and 1131 kWh during peak hours. The total electricity consumption with division to peak and off-peak usage is presented in Fig. 2. Prices for kWh in G12 tariffs are: 0.6556 PLN during peak period and 0.2492 PLN off peak period, which results in total costs 1855 PLN.

4.3 Selling Energy Back to the Grid

During summer months, PV system production exceeds the household demand, when comparing the 24-h load and production. None the less 20–50% of the electricity must be bought from the grid due to the discrepancy between demand and PV system output. The imbalance may be decreased by shifting switching on and off some devices, however it cannot be fully compensated. The existing regulations allow selling back the produced energy back to the grid. This section presents analysis in accordance to different Price-Based approaches to selling and buying back the generated electricity to the grid.

The current reselling price is set to 80 % of the price on the wholesale market which results in price 0.12 PLN per kWh. The newly introduced Renewable Energy Sources Act defines prices for the bought back energy, depending on the installation type and its overall power. For the analysed PV system, the price would be 0.65 PLN per kWh. Additionally, we analyse the theoretical approach when customer can take back the introduced energy 1:1, however he pays the distributional costs, which are

Table 3 Comparison of annual costs of energy with different Time-Based and Price-Based schemes

Selling to the grid approach	G11	G12
No resale	2787	1855
Current regulations	2570	1638
RES Act	1612	679
1:1	2248	1306–1765

around 0.20 PLN per kWh. The annual costs for all of aforementioned approached, in combination with the available TOU programs, are gathered in Table 3. For 1:1 strategy depending on the possibility to reuse energy in peak or off-peak periods the minimum and maximum price is given.

For the analysed household with PV system and heat pump, which is used only in off-peak period, it is clear that TOU strategy allows significant reduction of the overall costs. Current regulations concerning prices for the produced energy, which is sold to the grid doesn't provide much financial benefits, however new regulation will significantly change those proportions. The RES Act provides fixed prices for bought energy for the next 15 years and the provided analysis shows, that installing PV system may be very beneficial. It is also visible that possibility to buy later the produced and sold energy, could also provide significant savings and should be considered as future solution.

The purpose of the provided analysis was to give insight to the realm of polish electricity market as well as to evaluate the best approach for using the residential PV system. The size of the PV installation is typical, however its production highly depends on the solar activity therefore fluctuation between consecutive years are certain. None the less, it shouldn't exceed the 10–15%, which is also indicated in similar works [1]. Question arises if the household demands for electricity are referential. The house is equipped with heat pump, which is less popular and generates higher electricity usage during cold seasons, however average demand during remaining period is around 400 kWh per month which is also quite typical for a single-family household [9].

5 Conclusions and Future Work

The presented study determines that PV systems are an interesting and potentially profitable option for residential applications even in polish climatic conditions. The system can cover around 50 % of the self needs, while simultaneously drain the excess energy to the grid. Depending on the different approaches to the reselling scheme and available DSR programs, the beard costs may be significantly reduced. New regulations in the energy market are supposed to increase the implementation of renewable energy sources. The main advantage of new regulations is that end user

may sell the produced energy to the grid for the reasonable price, therefore those installations may be economically beneficial.

Electricity prices are not the only drivers for changes in consumers behaviour since only few users will spend time to analyse consumption decisions and micro management of home appliances. Smart technologies will be highly beneficial for introducing DSR program and will increase the responsiveness of residential energy consumers. Any smart system that will shift and curtail demands according to the current tariffs and electricity prices will improve the energy demand and supply balance without user involvement. The results from applying DSR strategy should alter timing and level of instantaneous demands as well as the overall consumption.

Acknowledgments This material is based upon work supported by The National Centre for Research and Development and The National Fund for Environmental Protection and Water Management under grant GEKON1/02/213877/31/2015.

References

1. Axaopoulos, P.J., Fylladitakis, E.D.: Performance and economic evaluation of a hybrid photovoltaic/thermal solar system for residential applications. Energy Buildings **65**, 488–496 (2013)
2. Casares, F.J., Lopez-Luque, R., Posadillo, R., Varo-Martinez, M.: Mathematical approach to the characterization of daily energy balance in autonomous photovoltaic solar systems. Energy **72**, 393–404 (2014)
3. Fischer, D., Hartl, A., Wille-Haussmann, B.: Model for electric load profiles with high time resolution for german households. Energy Buildings **92**, 170–179 (2015)
4. Logenthiran, T., Srinivasan, D., Shun, T.Z.: Demand side management in smart grid using heuristic optimization. IEEE Trans. Smart Grid **3**(3), 1244–1252 (2012)
5. Ministry of Economy: National action plan for energy from renewable sources (2010)
6. Ministry of Economy: Renewable energy sources act (2015)
7. Origin: Origin project webpage, http://www.origin-concept.eu/
8. Shariatzadeh, F., Mandal, P., Srivastava, A.K.: Demand response for sustainable energy systems: a review, application and implementation strategy. Renew. Sustain. Energy Rev. **45**, 343–350 (2015)
9. Widen, J., Wackelgard, E.: A high-resolution stochastic model of domestic activity patterns and electricity demand. Appl. Energy **87**(6), 1880–1892 (2010)

Bees Algorithm for the Quadratic Assignment Problem on CUDA Platform

Wojciech Chmiel and Piotr Szwed

Abstract With the proliferation of graphics processing units (GPU) supporting general-purpose computing (GPGPU), many computationally demanding applications are being redesigned to exploit the capabilities offered by massively parallel computing platforms. This paper presents a Bees Algorithm (BA) for the Quadratic Assignment Problem (QAP) implemented on the CUDA platform. The motivations for our work were twofold: firstly, we wanted to develop a dedicated algorithm to solve the QAP showing both time and optimization performance, secondly, we planned to check if the capabilities offered by popular GPUs can be exploited to accelerate hard optimization tasks requiring high computational power. The paper describes both sequential and parallel algorithm implementations, as well as reports results of tests.

Keywords QAP · Bees algorithm · CUDA · GPU calculation · GPGPU · Discrete optimization

1 Introduction

With the proliferation of graphics processing units (GPU) supporting general-purpose computing (GPGPU), many computationally demanding applications are being redesigned to exploit the capabilities offered by massively parallel computing platforms. They include such tasks as: physically based simulations, signal processing, ray tracing, geometric computing and data mining [20]. More recently several attempts have been made to develop various population based optimization algorithms on GPUs including: the particle swarm optimization [25, 26, 29], the ant colony optimization [27], the genetic [15] and memetic algorithm [13]. The described implementations benefit from the capabilities offered by GPUs by processing whole

W. Chmiel (✉) · P. Szwed
AGH University of Science and Technology, Kraków, Poland
e-mail: wch@agh.edu.pl

P. Szwed
e-mail: pszwed@agh.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_53

populations by fast GPU cores running in parallel. In this paper we research an implementation of the Bees Algorithm (BA) for the Quadratic Assignment Problem (QAP) on the NVIDIA CUDA platform.

The basic QAP formulation is the following: given a set of n *facilities* and n *locations*, the goal is to find an assignment of facilities to locations that minimizes the objective function, which is calculated as a sum of flows between facilities multiplied by distances between locations. As there are $n!$ possible assignments, QAP is one of the most difficult combinatorial problems belonging to the *NP-hard* class. Therefore, only approximation algorithms can be used for the case, where the n is bigger than 30 [3, 5, 6].

The Bees Algorithm (BA) is an approximation algorithm, inspired by the behavior of swarms of honey bees [21, 22]. The BA is a metaheuristics, that can be mapped on various domains and can vary in implementation details. Some modifications of BA are the Artificial Bee Colony (ABC) [1] and the Bee Colony Optimization [9].

Luo et al. [14] proposed the CUBA algorithm (CUDA based Bees Algorithm) dedicated for the CUDA platform. Threads are divided into blocks corresponding to different colonies. Each thread is assigned to a honey bee and performs search on behalf of its colony, so a number of colonies run the Bees Algorithm in parallel. The authors evaluated the performance of CUBA by conducting numerous experiments for continuous optimization problems.

Our work had two goals. The first was to develop an efficient bees algorithm for the QAP problem. In contrast to the previously reported sequential implementations, e.g. [14, 17], we intended to build a parallel application. Secondly, we intended to check, if the capabilities offered by popular GPUs can be exploited to accelerate hard optimization tasks requiring high computational power. In order to enable a comparison, two versions of algorithm were developed: a sequential executed on Java platform and a hybrid, whose parts were executed in parallel on CUDA.

The paper is organized as follows: next Sect. 2 gives the definition of the QAP. It is followed by Sect. 3, which presents the bees algorithm for solving the QAP. The parallel algorithm version is discussed in Sect. 4. Experiments performed and their results are presented in Sect. 5. Section 6 provides concluding remarks.

2 Quadratic Assignment Problem

The Quadratic Assignment Problem was introduced by Koopmans and Beckman in 1957, as a mathematical model of assigning indivisible economic activities to a set of locations.

For the given set $N = \{1, \ldots, n\}$ we define two $n \times n$ non-negative matrices $F = \left[f_{i,k}\right]$, $D = \left[d_{j,l}\right]$. In the terminology of facilities-location the set N is a set of facilities indexes and $\pi\colon N \rightarrow N$ defines locations, to which the facilities are assigned. Matrix D defines distances between locations, whereas matrix F defines flows between pairs of facilities. Matrix B describes a linear part of the assignment cost and in most cases is omitted. A solution of QAP (also denoted as $QAP(F, D)$)

can be defined in permutation form $\pi = (\pi(1), \ldots, \pi(n))$ of the set of n elements (facilities). In the Koopman-Beckman's [12] model the goal is to find the permutation π^* which minimizes the objective function:

$$\varphi(\pi^*) = min_{\pi \in \Pi} \sum_{i=1}^{n} \sum_{j=1}^{n} f_{ij} d_{\pi(i),\pi(j)} + \sum_{i=1}^{n} b_{i,\pi(i)} \tag{1}$$

The objective function $\varphi(\pi)$, $\pi \in \Pi$ describes a global cost of system realization and exploitation. Π is a set of permutations of the set of natural numbers $1, \ldots, n$. In most cases matrix D and F are symmetric: distances $d_{i,j}$ and $d_{j,i}$ between two locations i and j are equal, the same applies to flows: $f_{i,j}$ and $f_{j,i}$.

The QAP problem found application various areas including transportation [2], scheduling, electronics (wiring problem), distributed computing, statistical data analysis (reconstruction of destroyed soundtracks), balancing of turbine running [16], chemistry, genetics [23], creating the control panels and manufacturing [8].

In 1976 Sahni and Gonzalez [24] proved that the QAP is strongly $\mathcal{NP}$*-hard*, by showing that an existence of a polynomial time algorithm for solving the QAP implies the existence of a polynomial time algorithm for an $\mathcal{NP}$*-complete* decision problem—the Hamiltonian cycle (HC).

Chakrapani and Skorin-Kapov [4] proposed a parallel taboo search algorithm for the QAP problem, which included dynamically changing tabu list sizes, aspiration criteria and long term memory. An intensification strategy based on intermediate term memory was proposed and occurred promising, especially, while solving large QAPs. Taillard [28] proposed a taboo search algorithm where a length of taboo list is randomly changed in the limited scope. He tested two parallelization methods: the first consisted in dividing the neighborhood of the current solution into p parts of the same size and evaluating them on p processors; another way was to perform many independent searches starting from different solutions.

3 Bees Algorithm

The Bees Algorithm (BA) imitates the food foraging behaviour of swarms of honey bees [21]. In its basic version, the algorithm performs a kind of neighborhood search combined with the random search. A colony of bees searches space surrounding the hive in several directions in the distance of ten kilometers. Near places plentiful of nectar or pollen are visited more frequently than the other. At the beginning of the search process, the scouts are sent from the hive into promising paths. The scouts search randomly the space surrounding the hive and on return provide the information about the food sources found. The colony makes a decision about the number of bees sent to particular sites, assigning more bees to sources richer in food. The wealth of the food source is continuously monitored by the returning bees. It allows to react, if the amount of food decreases. In this case new scouts are sent to explore the space surrounding the hive and to find new promising food sources.

The bees algorithm [7] can be implemented in many ways, depending on various mappings of bees behavior on the elements used to model the optimization problem:

- creation of an initial bees' population,
- methods for selecting search directions (choice of solution to examine),
- choosing the numbers of scouts and locations,
- definition of the stop condition.

The proposed implementation of the bees algorithm adapted to the QAP (called the BA-QAP) is presented in Algorithm 1. At the beginning the bees population is created randomly (in the presented experiments the compared algorithms used identical initial population). Key elements, which determine the algorithm effectiveness are: the method of sites selection for the neighborhood search and the neighborhood size. The number of examined solutions in the selected sites should proportional to their quality. A population of l_e best solutions (called *elite* sites) and l_b good solutions are chosen from the whole population. The sizes of the search neighborhood for elite sites and other good solutions are defined by n_e and n_b coefficients, respectively. The remaining low quality solutions (having high values of criteria function) are ignored in the next search phase. Both l_e, l_b and n_e, n_b are the algorithm parameters.

To create the neighborhood solutions (coded as permutations) genetic unary operators dedicated to the QAP problem were used: *shift* the randomly chosen facility to a random position (other facilities are moved in the reverse direction) and *swap* the two randomly chosen facilities. The number of times those operators were executed is one of the algorithm parameters.

The next population of the solutions is created by choosing the best solution from the elite and good localizations. To keep the size of the population fixed, missing solutions are randomly created.

In the BA-QAP algorithm special mechanisms to prevent stagnancy by getting stuck at local minima were implemented. A solution can exist in the population only for a predefined number of iterations called the *life expectancy*. If the value of this parameter is exceeded, a new solution is randomly generated and replaces the old one. The best solution, which has been found so far is kept in the memory. The algorithm terminates after examining a predefined number of solutions.

Algorithm 1 uses the following variables: λ—swarm size, l_e—number of solution in elite (elite localization), l_b—number of good solutions (good localization), n_e—neighborhood size for the elite localization, n_b—neighborhood size for the good localization, π_{best}—the best found solution, $\varphi(\cdot)$—objective function value, μ—number of solutions which exist in swarm longer than maximum lifetime of solution, LT—solution life expectancy.

4 Implementation of Bees Algorithm on CUDA Platform

CUDA (*Compute Unified Device Architecture*) is hardware and software co-processing architecture created by NVIDIA corporation enabling decomposition of developed programs into two parts: sequential executed on CPU and parallel

Algorithm 1 BA-QAP algorithm.

Require $\lambda, l_e, l_b, n_e, n_b, \varphi(\cdot), LT$

Step 1. Initialize population with λ random solutions:

1. Create random population comprising λ individuals.
2. Evaluate fitness of the population members.
3. Sort the population (from best to worse).
4. Save the best solution.

Step 2. From the whole population select L_e: $|L_e| = l_e$ elite and L_b: $|L_b| = l_b$ best solutions (locations):

1. Define neighborhoods: $\forall \pi \in L_e : N(\pi)$ where $|N(\pi)| = n_e$ and $\forall \pi \in L_b : N(\pi)$ where $|N(\pi)| = n_b$.
   Calculate the objective function values for the solutions from neighborhoods.
2. Choose the best solution from the explored neighborhoods
   $\forall \pi \in L_e : \pi^* = arg \max_{\pi \in N(\pi)} \varphi(\pi)$ and $\forall \pi \in L_b : \pi^* = arg \max_{\pi \in N(\pi)} \varphi(\pi)$.

Step 3. Create new population:

1. Replace the best locations $L_e \cup L_b$ by the new solutions obtained in **Step 2.2**.
2. Remove μ solutions, which exist in population (swarm) longer than the predefined number of iterations LT (maximal lifetime of solution).
3. Randomly create μ new solutions (missing solutions to fit the population size).
4. Sort population (from best to worse).

Step 4. If the newly formed population comprises a solution with better value of criteria function than solution π_{best}, update π_{best}.

Step 5. Check the stop condition

1. If the stop condition is reached, then return π_{best} and $\varphi(\pi_{best})$.
2. Otherwise, return to **Step 2.**

running on NVIDIA graphics processing units (GPUs). Programs for GPU (called *kernels*) can be executed by thousands of concurrent threads and assigned to hundreds of processor cores [11, 18]. Kernels are developed in such sequential languages as C, C++ or Fortran. The CUDA programming library [19] provide some basic mechanisms supporting parallelism (assignment of threads to data, local barriers), task management and functions supporting communication between the host (CPU) and the device (GPU).

The architecture of GPU differs from CPU because it is designed following a few general ideas: simple decomposition, simple execution, simple synchronization and simple communication. In contrast to CPU, GPU threads are very lightweight, what assures small creation overhead and very fast switching. A powerful feature of CUDA is cooperation of threads based on shared memory (Fig. 1). In order to make threads cooperation more scalable, they are split into blocs (batches), where they can synchronize and communicate using shared memory. However, the threads in different blocks cannot cooperate.

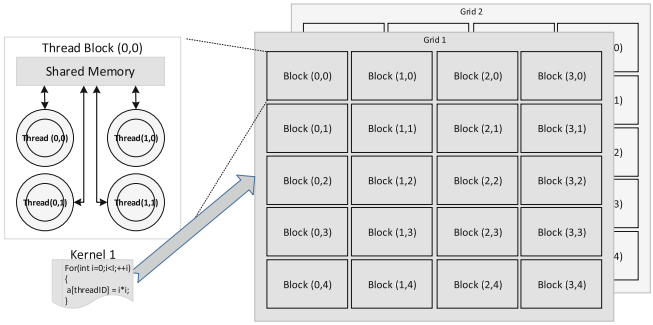


Fig. 1 CUDA architecture model

Three types of kernel memory access can be distinguished:

- *per-thread*: a thread can have access to registers (on-chip) and local memory (off-chip, uncached) to store its own variables used during calculation.
- *per-block*: threads from the block can communicate with one another using shared memory located on-chip. Usually this memory is small but fast.
- *per-device*: global memory located off-chip, large, uncached and persistent across kernel launches. It is usually used as a communication mechanism between the host and the device

CUDA implementation of the BA algorithm presented in this paper is based on JCuda [10], a Java library with bindings to CUDA runtime and NVIDIA driver API. JCuda include several libraries: JCublas—CUDA implementation of linear algebra, JCufft—provides the Fast Fourier Transformation, JCudpp—a bridge to CUDA Data Parallel Primitives Library and JCurand—offers GPU accelerated random number generator. In particular, the parallel algorithm implementation relies on JCurand (generation of random solutions), JCudpp (fast sorting algorithms) and JCublas (matrix multiplication to calculate goal function values).

Two BA algorithms were developed to compare the performance of the CPU and GPU implementations. The CPU version is presented in Algorithm 1. The GPU version is given in Fig. 2. The activities marked with gray are executed as kernels on the device (GPU). They include *init*—random solution initialization performed at the beginning and during global search, *goal*—calculation of the objective function value and *sort*—sorting solutions according to goal function values.

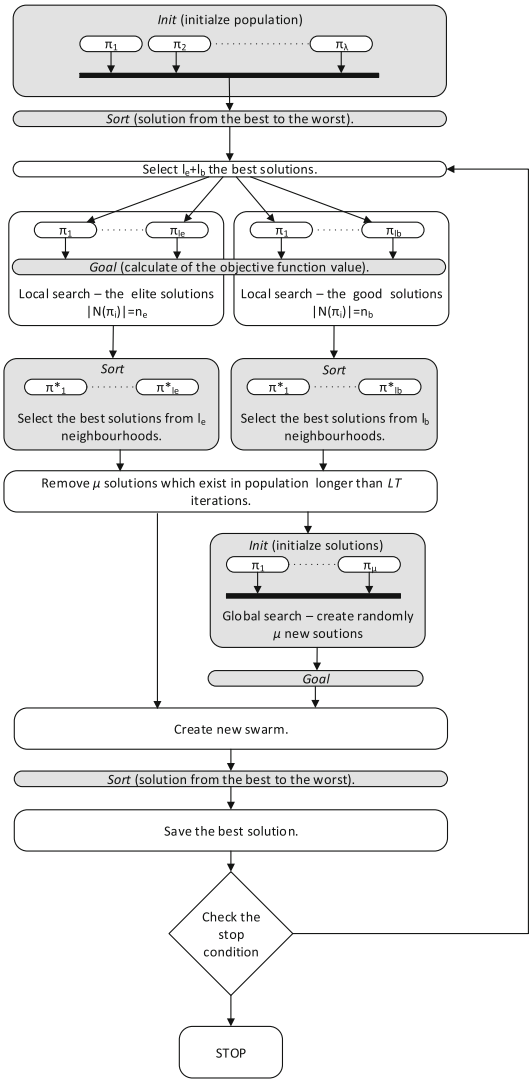


Fig. 2 Parallel BA-QAP algorithm. Activities marked in *gray* are executed on GPU

5 Experiments and Results

We have conducted two types of experiments. The first aimed at evaluating the time performance of GPU based implementation for various problems and population sizes. The goal of the second group of tests was to evaluate the algorithm performance for QAP instances published in QAPLIB problem library [3].

Table 1 Execution times (100 iterations, $\lambda = 100$) for selected problems from QAPLIB

Name	n	t_{CUDA} (ms)	t_{CPU} (ms)	$\frac{t_{CUDA}}{t_{CPU}}$
Chr12a	12	1044.79	371	2.81
Chr15b	15	830.89	424	1.96
Had16	16	938.12	474	1.98
Bur26a	26	1613.8	869	1.86
Esc32a	32	1929.36	1151	1.68
Ste36c	36	2204.92	1474	1.50
Lipa70b	70	4413.96	4448	0.99

5.1 Time Performance

The goal of the time performance tests was to compare two algorithm implementations: the parallel executed partially on the CUDA platform and the sequential running on CPU. During the tests the following platform was used: Intel Core i3-2350M CPU@2.30GHz, NVIDIA GeForce 630M, 4GB RAM, Windows 7, NVIDIA CUDA 6.5.

Table 1 summarizes execution times for selected problems from the QAPLIB library. In all cases algorithms performed 100 iterations for population size $\lambda = 100$. It can be observed that for small problems the communication overhead (copying data between the host and GPU memory) hinders gains from parallel execution. Execution times become comparable for larger problems.

The second group of tests were performed on the same problem Lipa50a (problem size 50). In all cases 100 iterations were performed for various population sizes: 20, 50, 100, 200, 500, 1000, 2000, 5000, 10000. To perform comparison, we have collected data on total execution times of three operations: *init*, *goal* and *sort* discussed in Sect. 4. Surprisingly, for the parallel implementation the execution times were nearly constant, regardless of the population size. They were about 50 ms for *init*, 243 ms for *goal* and 147 ms for *sort*. However, for the sequential implementation those times grew exponentially with the problem size. Figure 3 shows the ratio of the measured execution times for CUDA and CPU platforms (observe that the logarithmic scale is used). The results show clearly, that to exploit the leverage of the parallel platform, population based algorithms should rather process larger populations, in order of thousands of solutions.

5.2 Optimization Performance

The tests aiming at verifying the algorithm optimization performance were performed for selected problems from QAPLIB library. In all cases the same set of control parameters were used: number of scouts or swarm size ($\lambda = 100$), number of the best

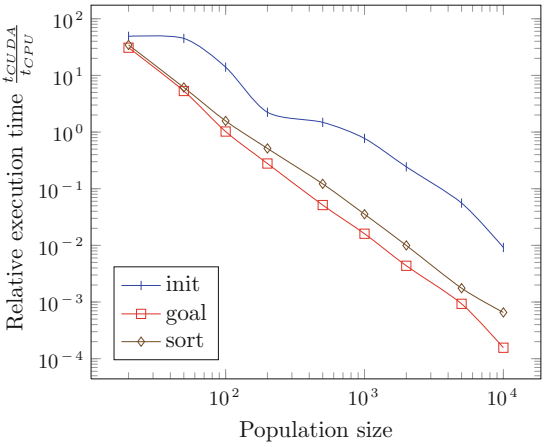


Fig. 3 Relative execution time of *init*, *goal* and *sort* operations on CUDA and CPU for various population sizes

locations ($_b = 7$), number of elite locations ($l_e = 3$), number of workers assigned to elite locations ($n_e = 40$), number of workers assigned to the best locations ($n_b = 30$), the maximal lifetime solution ($LT = 0.2 * \lambda$), maximum number of iterations ($I_{max} = 1000$). Table 2 summarizes results of the tests, giving the reference value of

Table 2 Results of performance tests for various problems from QAPLIB

Name	n	φ_{ref}	φ_{best}	I_{best}	t (ms)	E (%)
Chr12a	12	9552	9552	289	375	0.00
Chr15b	15	7990	7990	222	452	0.00
Esc16a	16	68	68	2	514	0.00
Had16	16	3720	3720	24	468	0.00
Chr18a	18	11,098	11,118	170	530	0.18
Chr20c	20	14,142	14,142	543	609	0.00
Rou20	20	725,522	727,322	700	590	0.25
Tai20a	20	703,482	724,472	446	671	2.98
Nug21	21	2438	2442	603	747	0.16
Chr25a	25	3796	3796	974	1156	0.00
Bur26a	26	5,426,670	5,431,640	832	826	0.09
Bur26b	26	3,817,852	3,817,948	335	796	0.00
Kra30b	30	91,420	93,350	903	1201	2.11
Esc32a	32	130	144	902	1092	10.77
Ste36c	36	8,239,110	8,752,218	981	1650	6.23
Lipa40a	40	31,538	32,091	917	1715	1.75
Wil50	50	48,816	50,216	991	2371	2.87
Esc64a	64	116	116	340	3027	0.00
Lipa70b	70	4,603,200	5,204,730	826	3769	13.07
Tai80a	80	13,499,184	14,784,380	737	4087	9.52

the objective function value for the selected problem from QAPLIB library (column φ_{ref}), the best objective function value determined (φ_{best}), iteration, in which the best solution was reached (I_{best}), algorithm execution time (t [ms]) and the relative percentage gap ($E = 100\,\% \cdot (\varphi_{best} - \varphi_{ref})/\varphi_{ref}$) between the reached solution and the reference value. It can be observed that in most cases the gap is relatively small, however it is higher for greater problem sizes.

6 Conclusions

In this paper we describe the BA-QAP bees algorithm designed for solving the QAP problem, as well as its parallel implementation on the CUDA platform.

We report results of tests aiming at evaluating the implementation in terms of execution times and optimization capability. The tests targeting time performance revealed that benefits of GPU calculations can be observed, if large swarms are processed in parallel. According to analyses reported in [18], for an application with a moderate number of parallel tasks (25 %) reduction of execution time by 25 % may be expected, whereas for parallel intensive program such reduction may reach 99.3 %. In this light, the BA-QAP implementation is still a "mostly sequential program", however with growing swarm size it may move towards massive parallel. The results of tests revealed that within the range of swarm sizes 10–10000, the execution time of parallel tasks is practically constant.

Tests of optimization performance performed on several QAP instances showed that the algorithm behaves correctly. It can be expected that better results for larger problems can be obtained with growing swarm sizes.

References

1. Bansal, J.C., Sharma, H., Nagar, A., Arya, K.V.: Balanced artificial bee colony algorithm. Int. J. Artif. Intell. Soft Comput. **3**(3), 222–243 (2013)
2. Bermudez, R., Cole, M.H.: A genetic algorithm approach to door assignments in breakbulk terminals. Technical Report MBTC-1102, Mack-Blackwell Transportation Center, University of Arkansas, Fayetteville, Arkansas (2001)
3. Burkard, R., Karisch, S., Rendl, F.: QAPLIB—a quadratic assignment problem library. J. Glob. Optim. **10**(4), 391–403 (1997)
4. Chakrapani, J., Skorin-Kapov, J.: Massively parallel tabu search for the quadratic assignment problem. Ann. Oper. Res. **41**(4), 327–341 (1993)
5. Chmiel, W.: Evolution Algorithms for optimisation of task assignment problem with quadratic cost function. Ph.D. thesis, AGH Technology University, Kraków, Poland (2004)
6. Chmiel, W., Kadłuczka, P., Packanik, G.: Performance of swarm algorithms for permutation problems. Automatyka **15**(2), 117–126 (2009)
7. Chong, C.S., Sivakumar, A.I., Low, M.Y.H., Gay, K.L.: A bee colony optimization algorithm to job shop scheduling. In: WSC 2006. Monterey, USA (2006)
8. Grötschel, M.: Discrete mathematics in manufacturing. In: Malley, R.E.O. (ed.) ICIAM 1991, pp. 119–145 (1991)

9. Huang, Y.M., Lin, J.C.: A new bee colony optimization algorithm with idle-time-based filtering scheme for open shop-scheduling problems. Expert Syst. Appl. **38**(5), 5438–5447 (2011)
10. jcuda.org: JCuda–Java bindings for CUDA (2015), http://www.jcuda.org
11. Kirk, D.B., Hwu, W.M.: Programming Massively Parallel Processors: A Hands-on Approach, 1st edn. Morgan Kaufmann Publishers, San Francisco, USA (2010)
12. Koopmans, T.C., Beckmann, M.J.: Assignment problems and the location of economic activities. Econometrica **25**, 53–76 (1957)
13. Krüger, F., Maitre, O., Jiménez, S., Baumes, L.A., Collet, P.: Generic local search (memetic) algorithm on a single GPGPU chip. In: Tsutsui, S., Collet, P. (eds.) Massively Parallel Evolutionary Computation on GPGPUs, pp. 63–81. Natural Computing Series, Springer, Berlin (2013)
14. Luo, G.H., Huang, S.K., Chang, Y.S., Yuan, S.M.: A parallel bees algorithm implementation on GPU. J. Syst. Arch. **60**(3), 271–279 (2014)
15. Maitre, O.: Genetic programming on GPGPU cards using EASEA. In: Tsutsui, S., Collet, P. (eds.) Massively Parallel Evolutionary Computation on GPGPUs, pp. 227–248. Natural Computing Series, Springer, Berlin (2013)
16. Mason, A., Rönnqvist, M.: Solution methods for the balancing of jet turbines. Comput. Oper. Res. **24**(2), 153–167 (1997)
17. Mirzazadeh, M., Shirdel, G.H., Masoumi, B.: A honey bee algorithm to solve quadratic assignment problem. J. Optim. Ind. Eng. **9**, 27–36 (2011)
18. Nickolls, J., Dally, W.J.: The GPU computing era. IEEE Micro **30**(2), 56–69 (2010)
19. NVIDIA Corporation: CUDA toolkit documentation v6.5 (2015), http://docs.nvidia.com/cuda/index.html#axzz3T4PFSm60
20. Owens, J.D., Luebke, D., Govindaraju, N., Harris, M., Krüger, J., Lefohn, A.E., Purcell, T.J.: A survey of general-purpose computation on graphics hardware. Comput. Graph. Forum **26**(1), 80–113 (2007)
21. Pham, D.T., Castellani, M.: Benchmarking and comparison of nature-inspired population-based continuous optimisation algorithms. Soft Comput. **18**, 1–33 (2013)
22. Pham, D.T., Ghanbarzadeh, A., Koc, E., Otri, S., Rahim, S., Zaidi, M.: The bees algorithm, a novel tool for complex optimisation problems. IPROMS **2006**, 454–459 (2006)
23. Phillips, A.T., Rosen, J.B.: A quadratic assignment formulation of the molecular conformation problem. J. Glob. Optim. **4**, 229–241 (1994)
24. Sahni, S., Gonzalez, T.: P-complete approximation problems. J. ACM **23**(3), 555–565 (1976)
25. Szwed, P., Chmiel, W.: Multi-swarm PSO algorithm for the quadratic assignment problem: a massive parallel implementation on the OpenCL platform. Comput. Res. Repos. 1504.05158 (2015)
26. Szwed, P., Chmiel, W., Kadłuczka, P.: OpenCL implementation of PSO algorithm for the quadratic assignment problem. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) Artificial Intelligence and Soft Computing, LNCS, vol. 9120, pp. 223–234. Springer, Switzerland (2015)
27. Tadeusiewicz, R., Lewicki, A.: The ant colony optimization algorithm for multiobjective optimization non-compensation model problem staff selection. In: Cai, Z., Hu, C., Kang, Z., Liu, Y. (eds.) Advances in Computation and Intelligence, LNCS, vol. 6382, pp. 44–53. Springer, Berlin (2010)
28. Taillard, E.: Robust taboo search for the quadratic assignment problem. Parallel Comput. **17**(4–5), 443–455 (1991)
29. Zhou, Y., Tan, Y.: GPU-based parallel particle swarm optimization. In: IEEE CEC 2009, pp. 1493–1500. Trondheim, Norway (2009)

Grammatical Inference in the Discovery of Generating Functions

Wojciech Wieczorek and Arkadiusz Nowakowski

Abstract In this paper an algorithm for the induction of a context-free grammar is proposed, and its application in obtaining a generating function for the number of certain combinatorial objects is demonstrated. In particular, two problems classified in The On-Line Encyclopedia of Integer Sequences (http://oeis.org/) under entries A000073 and A000108, as well as a problem from the domain of chemoinformatics, are solved as an illustration of our method.

Keywords Grammatical inference · CFG induction · Constraint satisfaction · Generating functions

1 Introduction

The induction of automata or string-rewriting systems has been the subject of scientific experiments and theoretical research for over 30 years [6, 9, 14]. From the practical viewpoint, i.e., getting a model from a finite set of labeled strings, it is known that the task is intractable. Specifically, Gold proved that given a finite alphabet Σ, two finite sets of strings S_+ and S_- built from symbols taken from Σ, and an integer k, then determining whether there is a k-state DFA (deterministic finite automaton) that recognizes L such that every string from S_+ is also in L and no string from S_- is in L, is an NP-complete problem [11]. Furthermore, Angluin showed in her PhD thesis that there is no polynomial time algorithm for finding a shortest compatible regular expression for arbitrary given data [2]. As regards moving up from the regular world to the context-free world, we are faced with a whole set of new difficulties (see Chap. 15 of [14] for a discussion of this topic). However, the development of heuristic methods has helped to apply inductive inference to such fields

W. Wieczorek (✉) · A. Nowakowski
Institute of Computer Science, University of Silesia, Sosnowiec, Poland
e-mail: wojciech.wieczorek@us.edu.pl

A. Nowakowski
e-mail: arkadiusz.nowakowski@us.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_54

as computational linguistics, pattern recognition, machine learning, bio-informatics, and others [13, 20].

This study will be especially concerned with explaining how the induction of a context-free grammar can support discovering ordinary generating functions (also called OGFs). In combinatorics, the closed form of an OGF is often the basic way of representing infinite sequences. Suppose that we are given a description of the construction of some structures we wish to count. The idea is as follows. First, define a one-to-one correspondence (a bijection) between the structures and the language over a fixed alphabet. Next, determine some examples (also called positive strings) and counterexamples (also called negative strings). Infer an unambiguous context-free grammar consistent with the sample. Via classical Schützenberger methodology, give a set of function equations. Finally, solve it and establish a hypothesis for the OGF. Our main contribution is to provide a procedure for inferring the smallest context-free grammar which accepts all examples and none of the counterexamples. The grammar is likely to be unambiguous, which is crucial in the Schützenberger methodology. The most closely related work to our study is by Imada and Nakamura [16]. Their work differs from ours in four respects. Firstly, they translated the learning problem for a CFG into an SAT, which is then solved by an SAT solver. We did the translation into 0–1 NP (zero-one nonlinear programming[1]), which is then solved by a CSP (constraint satisfaction programming) solver. Secondly, they minimize the number of rules, while we minimize the sum of the lengths of the rules. Thirdly, their goal is to obtain a grammar in Chomsky normal form. In contrast, we get a grammar in quadratic Greibach normal form. And fourthly, every grammar we obtain is unambiguous with respect to the examples, which is not the case in the comparable approach. The last two points are essential in our application, because they favor unambiguity. If we obtained an ambiguous grammar, the sequence determined by an OGF would be only upper bounds on the number of objects.

This paper is organized into six sections. Section 2 introduces the notion of ordinary generating functions and discusses the basic techniques for manipulating them. Section 3 translates the task of induction into a 0–1 NP, and describes the procedure of using it. In Sect. 4 the relation between grammars and generating functions is explained. Section 5 discusses the experimental results. Conclusions and research perspectives are contained in Sect. 6.

2 Generating Functions

In this section, the concept of ordinary generating functions and ways to manipulate them is introduced. For a more detailed presentation and advanced applications of generating functions, the reader is referred to [5, 12].

[1] Our translation can be further re-formulated as an integer linear program, but the number of variables increases so much that this is not profitable.

Let $a_0, a_1, \ldots$ be a sequence (especially of non-negative integers). Then the power series

$$A(z) = \sum_{i=0}^{\infty} a_i z^i \tag{1}$$

is called the *ordinary generating function* (OGF) associated with this sequence. For simple sequences, the closed forms of their OGFs can be obtained fairly easily. Take the sequence 1, 1, 1, ..., for instance:

$$A(z) = 1 + z + z^2 + z^3 + \cdots = 1 + z(1 + z + z^2 + \cdots) \ . \tag{2}$$

After a few transformations we see that $A(z) = 1/(1-z)$. There are many tools from algebra and calculus to obtain information from generating functions or to manipulate them. Partial fractions are a good case in point, since they have been shown to be valuable in solving many recursions. Herein only the basic techniques—those that will be helpful in the later investigation—are recalled.

To shift $A(z)$ to the right by m places, that is, to produce the OGF for the sequence $\langle 0, \ldots, 0, a_0, a_1, \ldots\rangle = \langle a_{n-m}\rangle$ with m leading 0's, we simply multiply by z^m:

$$z^m A(z) = \sum_{n\geq 0} a_n z^{n+m} = \sum_{n\geq 0} a_{n-m} z^n \ , \quad \text{integer } m \geq 0 \ . \tag{3}$$

And to shift $A(z)$ to the left by m places, that is, to form the OGF for the sequence $\langle a_m, a_{m+1}, a_{m+2}, \ldots\rangle = \langle a_{n+m}\rangle$ with the first m elements discarded, we subtract the first m terms and then divide by z^m:

$$\frac{A(z) - a_0 - \cdots - a_{m-1} z^{m-1}}{z^m} = \sum_{n\geq m} a_n z^{n-m} = \sum_{n\geq 0} a_{n+m} z^n \ . \tag{4}$$

Let $A(z)$ be the OGF for the sequence $\langle a_0, 0, a_2, 0, a_4, 0, \ldots\rangle$. In order to get rid of these odd-positioned zeros, i.e., to obtain the OGF for the sequence $\langle a_0, a_2, a_4, a_6, \ldots\rangle$, we may substitute z for z^2 in $A(z)$. For example[2]:

$$\frac{1}{1-3z^2} = 1 + 3z^2 + 9z^4 + 27z^6 + 81z^8 + \cdots \tag{5}$$

whereas

$$\frac{1}{1-3z} = 1 + 3z + 9z^2 + 27z^3 + 81z^4 + \cdots \ . \tag{6}$$

[2]To expand this fraction in a power series, one can use Maclaurin's formula:

$$f(z) = f(0) + \frac{z}{1!} f'(0) + \frac{z^2}{2!} f''(0) + \cdots + \frac{z^n}{n!} f^{(n)}(0) + \cdots \ .$$

3 Grammar Induction

This section will be devoted to our main result: the translation of CFG identification into a 0–1 non-linear programming problem. In a classical, Gold's model of *identification in the limit*, the input of a learning algorithm is an infinite sequence of examples (including counterexamples) of the unknown grammar [10]. The setting of this model is that of on-line, incremental learning. After each new example the learner (the algorithm) must return some hypothesis (a grammar). Identification is achieved when the learner returns a correct answer and does not change its decision afterwards. In the present paper, however, the problem is treated as a problem of combinatorial optimization, which is constituted by the following task: given two disjoint finite sets $S_+, S_- \subset \Sigma^+$ of words and an integer $k > 0$, build the smallest context-free grammar with k variables that accepts the language S_+ and does not accept any word from the set S_-. We also try to find as small a k as possible, in accordance with Occam's razor, demanding that we should take the simplest possible theoretical explanation for the existing data.

We assume the reader is familiar with elementary formal language theory. The best general reference here is Hopcroft et al. [15] or Lothaire [18] (words and languages), and Du and Ko [7] or Hopcroft et al. [15] (context-free grammars). For a deeper discussion of normal forms, we refer the reader to [21].

3.1 The Formulation of the Grammar Induction Problem

A zero-one non-linear programming problem (0–1 NP) deals with the question of whether there exists an assignment to binary variables $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ that satisfies the constraints $f_i(\mathbf{x}) = 0, i \in I$, and (optionally) simultaneously minimizes (or maximizes) some expression involving $\mathbf{x}$, where $f_i(\mathbf{x})$ $(i \in I)$ are given non-linear functions. Binary variables $(x_i \in \{0, 1\})$ are usually used for such purposes as modeling yes/no decisions, enforcing disjunctions, enforcing logical conditions, etc.

Let Σ be an alphabet, let S_+ (examples) and S_- (counterexamples) be two disjoint finite sets of words over Σ, and let $k > 0$ be an integer. The goal of CFG induction is to determine a grammar $G = (V, \Sigma, P, v_1)$ such that $L(G)$ contains S_+ and is disjoint from S_-. Moreover, the following criteria have to be fulfilled: G is in quadratic form, $|V| \leq k$, every word $w \in S_+$ has a unique leftmost derivation, and the sum of the lengths of the right hand sides of the rules is minimal.

3.2 Encoding

Let $S = S_+ \cup S_-$ $(S_+ \cap S_- = \emptyset, \epsilon \notin S)$, and let F be the set of all proper factors of all words of S. An alphabet $\Sigma \subseteq F$ is determined by S, and variables V are determined by k: $V = \{v_1, v_2, \ldots, v_k\}$. The binary variables will be $w[n, f]$,

$x[n, a, i, j]$, $y[n, a, i]$, and $z[n, a]$, where $i, j, n \in V$, $a \in \Sigma$, and $f \in F$. The value of $w[n, f]$ is 1 if $n \Rightarrow^* f$ holds in a grammar G, and $n[w, f] = 0$ otherwise. The value of $x[n, a, i, j]$ is 1 if $n \to a\, i\, j \in P$, and $x[n, a, i, j] = 0$ otherwise. The value of $y[n, a, i]$ is 1 if $n \to a\, i \in P$, and $y[n, a, i] = 0$ otherwise. Finally, we let $z[n, a] = 1$ if $n \to a \in P$ and zero if not. Let us now see how to describe the constraints of the relation between a grammar G and a set F in terms of non-linear equations.

Naturally, every example has to be accepted by the grammar, but no counterexample should be. This can be written as

$$w[v_1, s] = 1 \qquad s \in S_+ \,, \tag{7}$$

$$w[v_1, s] = 0 \qquad s \in S_- \,. \tag{8}$$

We want to express the fact that whenever $w[n, f] = 1$ for $f = abc$ or $f = ab$ or $f = a, a \in \Sigma, b, c \in F$, we have exactly one[3] product $x[n, a, i, j] \cdot w[i, b] \cdot w[j, c]$ or $y[n, a, i] \cdot w[i, b]$ or $z[n, a]$ equal to 1. And vice versa, if, for instance, a factor $f = ab$ and there is a rule $n \to a\, i$ and b can be derived from a variable i, then $w[n, f] = 1$ has to be fulfilled. We can guarantee this by requiring

$$\begin{aligned} w[n, f] = & \sum_{\substack{i,j \in V \\ abc = f}} x[n, a, i, j] \cdot w[i, b] \cdot w[j, c] \\ & + \sum_{\substack{i \in V \\ ab = f}} y[n, a, i] \cdot w[i, b] \\ & + (z[n, f] \text{ if } f \in \Sigma) \end{aligned} \tag{9}$$

for each $(n, f) \in V \times F$. Obviously, we are to find the minimum value of the linear expression

$$3 \sum_{\substack{a \in \Sigma \\ i,j,n \in V}} x[n, a, i, j] + 2 \sum_{\substack{a \in \Sigma \\ i,n \in V}} y[n, a, i] + \sum_{\substack{a \in \Sigma \\ n \in V}} z[n, a] \,. \tag{10}$$

3.3 Usage

Suppose that the learning process is based on the existence of an *Oracle*, which can be seen as a device that:

1. Knows the language and has to answer correctly.
2. Can answer *equivalence queries*. They are made by proposing some hypothesis to the Oracle. The hypothesis is a grammar representing the unknown language. The Oracle just answers YES in the positive case. In the negative case, the Oracle

[3] Recall that we hope to obtain grammars that are likely to be unambiguous.

has to return the shortest string in the symmetric difference between the target language and the submitted hypothesis.

Then the following procedure can be applied. Start from a small[4] sample S and $k = 1$. Run a non-linear program. Every time it turns out that there exists no solution that satisfies all of the constraints, increase k by 1. As long as the Oracle returns a word w in response to an equivalence query, add w to S and run a new non-linear program. Stop after the answer is YES. Unfortunately, there is no guarantee that this will terminate in a polynomial number of steps even when the target language is regular [3].

This procedure imperfectly, but practically, matches the man–machine interaction scheme. Provided that there is an algorithm for generating objects and associated words, the man can play the role of the Oracle. The equivalence checking may be done by random sampling. The positive answer could be incorrect, but this probability decreases if the sampling is repeated. The shortest witness w can be generated manually or by a simple computer program as well. Note also that the ambiguity of a context-free grammar might be checked by means of the LR(k) test or other methods [4].

4 The Schützenberger Methodology

The idea of constructing a bijection between a class of combinatorial objects and the words of a language in order to deduce the generating function of the sequence of some parameter p on these objects is known as the Schützenberger methodology and has been developed since the 1960s. As a good bibliographical starting point, see [8, 17].

Let us briefly describe this theory. We will denote by $G = (V, \Sigma, P, S)$ a context-free unambiguous grammar, and by a_i the number of words of length i in $L(G)$. Let Θ be a map that satisfies the following conditions:

1. for every $a \in \Sigma$, $\Theta(a) = z$,
2. $\Theta(\epsilon) = 1$,
3. for every $N \in V$, $\Theta(N) = N(z)$.

If for every set of rules $N \to \alpha_1 \mid \alpha_2 \mid \ldots \mid \alpha_m$ we write $N(z) = \Theta(\alpha_1) + \Theta(\alpha_2) + \cdots + \Theta(\alpha_m)$, then

$$S(z) = \sum_{i=0}^{\infty} a_i z^i \ . \tag{11}$$

[4] We are aware of this imprecision. The number of words and their lengths should allow of executing a program in a reasonable amount of time. This, in turn, depends on many circumstances.

Thus, to take one example, let us try to find the number of ways in which one can go from point $(0, 0)$ to point $(n, 0)$ in the Cartesian plane using only the two types of moves, $u = (1, 1)$ and $d = (1, -1)$, crossing neither the $y = 1$ line nor the $y = -1$ line. It is easy to see that every valid sequence of moves is accepted by the grammar $S \to \epsilon \mid u\, d\, S \mid d\, u\, S$. This clearly forces $S(z) = 1 + z^2 S(z) + z^2 S(z)$, and consequently $S(z) = 1/(1 - 2z^2)$. The Maclaurin series coefficients of $S(z)$ determine the number of words of a fixed length, so we can look up the number of unique paths from the starting to the ending point: $\langle 1, 0, 2, 0, 4, 0, 8, \ldots \rangle$ for consecutive lengths $n = 0, 1, 2, \ldots$.

5 Applications

In this section, three examples of problems from enumerative combinatorics will be solved by means of our grammar induction method. In every problem, we are given an infinite class of finite sets S_i where i ranges over some index set I (such as the nonnegative integers $\mathbb{N}$), and we wish to count the number $f(i)$ of elements of each S_i "simultaneoously," i.e., give a generating function. In the first and third examples we will define the necessary bijections on our own, while in the second example, we will rely on the well-known correspondence between trees and words.

5.1 Compositions of a Natural Number

Compositions are merely partitions in which the order of summands is considered. For example, there are four compositions of 3: 3, 1+2, 2+1, 1+1+1. The problem we are dealing with in this subsection is to count the number of compositions of n with no part greater than 3. Fortunately, there is a straightforward one-to-one correspondence between the compositions and a language over the alphabet $\Sigma = \{a, b, c\}$. The sign a is associated with 1, bb with 2, and ccc with 3. So we have, for example, $acccbb$ corresponding with $1 + 3 + 2$, while abc does not belong to the language.

Using[5] the method described in Sect. 3, the following grammar is obtained:

$$
\begin{aligned}
v_1 &\to a \mid a\, v_1 \mid b\, v_3 \mid c\, v_4 \\
v_2 &\to c \mid c\, v_1 \\
v_3 &\to b \mid b\, v_1 \\
v_4 &\to c\, v_2
\end{aligned}
$$

[5] To model and solve our non-linear program we make use of the Optimization Modeling Language (OML) and Microsoft Solver Foundation 3.1 development tools.

Employing the methodology specified in Sect. 4 gives the following set of equations:

$$v_1(z) = z + zv_1(z) + zv_3(z) + zv_4(z)\,, \qquad v_2(z) = z + zv_1(z)\,,$$
$$v_3(z) = z + zv_1(z)\,, \qquad v_4(z) = zv_2(z)$$

which yields

$$v_1(z) = \frac{z + z^2 + z^3}{1 - z - z^2 - z^3}\,. \tag{12}$$

This is an OGF for the sequence $\langle 0, 1, 2, 4, 7, 13, 24, 44, 81, 149, \ldots\rangle$, and as we can see in the OEIS under entry A000073, our conjecture is indeed correct.

5.2 Rooted Plane Trees

If T is a connected undirected graph without any cycles, then T is called a *tree*. A pair (T, r) consisting of a tree T and a specified vertex r is called a *rooted tree* with *root* r. If, additionally, for each vertex, the children of the vertex are ordered, the result is a *rooted plane tree*.

Let w be a word formed with the letters x and y. We will say that w is a *Dyck word* if w is either an empty word or a word $xuyv$ where u and v are Dyck words. A word w is a 1-dominated sequence if and only if there exists a prefix of a Dyck word u such that $w = xu$.

Let s_n be the number of unlabeled rooted plane trees with n vertexes. To find an OGF for s_n, we can take advantage of a mapping g that maps a 1-dominated word that has n letters x and $n - 1$ letters y to a tree T with n nodes [1]. The algorithm for generating $g(w)$ is given below as a pseudo-code:

```
create an empty stack
read the letters of w to be transformed
    create a new vertex v
    while the next letter of the sequence is y
        read this letter
        pop the top tree from the stack
        add it as the left child of v
    push the tree with root v onto the stack
in the end the stack contains the tree g(w)
```

By means of our procedure and the Schützenberger methodology, we obtain

$$\left.\begin{aligned} v_1 &\to x \mid x\, v_1\, v_2 \\ v_2 &\to y \mid y\, v_1\, v_2 \end{aligned}\right\} \Longrightarrow v_1(z) = \frac{1 - \sqrt{1 - 4z^2}}{2z}\,. \tag{13}$$

This is an OGF for the sequence $\langle 0, 1, 0, 1, 0, 2, 0, 5, 0, 14, 0, 42, \ldots \rangle$. However, when $|w| = 2n - 1$ the number of nodes is n, so we need every second element from the right shifted sequence. Employing (3), (5) and (6), finally we have

$$s_n = [z^n] \frac{1 - \sqrt{1 - 4z}}{2}, \quad n \in \mathbb{N} . \tag{14}$$

5.3 Secondary Structure of Single-Stranded tRNAs

A transfer RNA (abbreviated tRNA) is an adaptor molecule composed of RNA, typically 76 to 90 nucleotides in length, that serves as the physical link between the nucleotide sequence of nucleic acids and the amino acid sequence of proteins. The structure of tRNA can be decomposed into its primary structure (the exact sequence of nucleotides that make up the whole molecule), its secondary structure (usually visualized as a cloverleaf structure), and its tertiary structure (the three-dimensional structure defined by the atomic coordinates).

One of the problems considered in chemoinformatics and combinatorics is to count the number of secondary structures of single-stranded tRNAs having a certain size. To this end, the special graph theory of secondary structures has been developed. For the convenience of the reader, we repeat the relevant material from [19], thus making our exposition self-contained. We call a graph on the set of n labeled points $\{1, 2, \ldots, n\}$ a *secondary structure* if its adjacency matrix $A = (a_{ij})$ has the following three properties:

1. $a_{i,i+1} = 1$ for $1 \leq i \leq n - 1$.
2. For each fixed i, $1 \leq i \leq n$, there is at most one $a_{ij} = 1$ where $j \neq i \pm 1$.
3. $a_{ij} = a_{kl} = 1$, where $i < k < j$, implies $i \leq l \leq j$.

Let $S(n)$ be the number of secondary structures for n points. It has been proved that $S(1) = S(2) = 1$, and for $n > 2$, $S(n)$ satisfies

$$S(n + 1) = S(n) + \sum_{k=0}^{n-2} S(k)S(n - k - 1) , \tag{15}$$

where $S(0) \equiv 1$. If the OGF $\phi(z)$ is defined by $\phi(z) = \sum_{n \geq 0} S(n)z^n$, the recursion formula (15) can be multiplied by z^{n+1} and summed to obtain

$$\phi(z) = \frac{1 - z - z^2 - \left[1 + z(z^3 - 2z^2 - z - 2)\right]^{1/2}}{2z^2} . \tag{16}$$

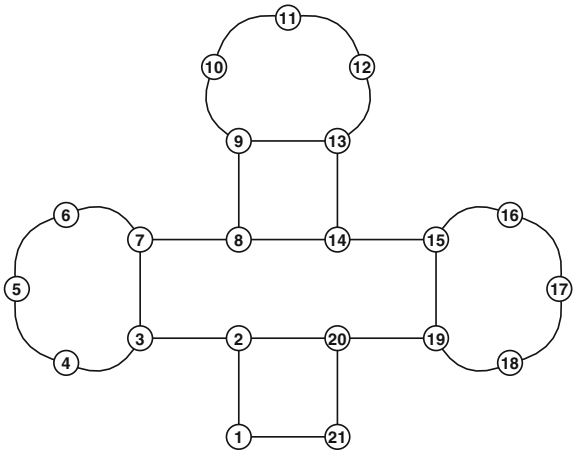


Fig. 1 The secondary structure for the word `(((aaa)((aaa))(aaa)))`

We can obtain the same result, (16), by means of our method by using the following bijection (see Fig. 1 as an illustration of this):

- A secondary structure on n points is represented by a word $w \in \{a, (,)\}^n$ in which the parentheses are well-balanced.
- If $a_{ij} = 1$ for $1 \leq i < j - 1 < n$, then $w_i = ($ and $w_j =)$, for the remaining positions k let $w_k = a$.

The productions of the resulting unambiguous grammar are $v_1 \to a \mid a\, v_1 \mid (v_1\, v_2$ and $v_2 \to) \mid)\, v_1$.

6 Conclusions

In the present paper, the way in which grammar induction may support finding generating functions has been revealed. This subject is especially important for such problems where obtaining an OGF by standard methods is hard. The proposed idea is not free from objections. Among the most serious complications are: (a) uncertainty about the possibility of describing combinatorial objects by a context-free grammar, (b) uncertainty about the total covering of the objects by a finding grammar, (c) it is undecidable whether a CFG is ambiguous in the general case [15].

Acknowledgments This research was supported in part by PL-Grid Infrastructure, and by Grant No. DEC-2011/03/B/ST6/01588 from National Science Center of Poland.

References

1. Alonso, L., Schott, R.: Random Generation of Trees: Random Generators in Computer Science. Springer, New York (1995)
2. Angluin, D.: An application of the theory of computational complexity to the study of inductive inference. Ph.D. thesis, University of California (1976)
3. Angluin, D.: Negative results for equivalence queries. Mach. Learn. **5**, 121–150 (1990)
4. Basten, H.J.S.: The usability of ambiguity detection methods for context-free grammars. Electron. Notes Theor. Comput. Sci. **238**(5), 35–46 (2009)
5. Bender, E.A., Williamson, S.G.: Foundations of Combinatorics with Applications. Dover Books on Mathematics Series. Dover Publications, Dover (2006)
6. Book, R., Otto, F.: String-Rewriting Systems. Springer, New York (1993)
7. Du D.Z., Ko, K.: Problem Solving in Automata, Languages, and Complexity. Wiley, New York (2001)
8. Delest, M.: Algebraic languages: a bridge between combinatorics and computer science. DIMACS: Ser. Discrete Math. Theor. Comput. Sci. **24**, 71–87 (1994)
9. Eyraud, R., de la Higuera, C., Janodet, J.C.: Lars: a learning algorithm for rewriting systems. Mach. Learn. **66**, 7–31 (2007)
10. Gold, E.M.: Language identification in the limit. Inf. Control **10**, 447–474 (1967)
11. Gold, E.M.: Complexity of automaton identification from given data. Inf. Control **37**, 302–320 (1978)
12. Graham, R., Knuth, D., Patashnik, O.: Concrete Mathematics: A Foundation for Computer Science. Addison-Wesley, New York (1994)
13. de la Higuera, C.: A bibliographical study of grammatical inference. Pattern Recogn. **38**, 1332–1348 (2005)
14. de la Higuera, C.: Grammatical Inference: Learning Automata and Grammars. Cambridge University Press, Cambridge (2010)
15. Hopcroft, J.E., Motwani, R., Ullman, J.D.: Introduction to Automata Theory, Languages, and Computation, 2nd edn. Addison-Wesley, New York (2001)
16. Imada, K., Nakamura, K.: Learning context free grammars by using sat solvers. In: ICMLA 2009, pp. 267–272. IEEE Computer Society (2009)
17. Kuich, K., Salomaa, A.: Semirings, Automata, Languages. Springer, Berlin (1985)
18. Lothaire, M.: Algebraic Combinatorics on Words, Encyclopedia of Mathematics and its Applications, vol. 90. Cambridge University Press, Cambridge (2002)
19. Waterman, M.S.: Secondary structure of single-stranded nucleic acids. In: Studies on Foundations and Combinatorics. Advances in Mathematics Supplementary Studies, vol. 1, pp. 167–212. Academic Press, New York (1978)
20. Wieczorek, W., Unold, O.: Induction of directed acyclic word graph in a bioinformatics task. JMLR Workshop Conf. Proc. **34**, 207–217 (2014)
21. Wood, D.: A generalised normal form theorem for context-free grammars. Comput. J. **13**(3), 272–277 (1970)

Optimization of Decision Rules Relative to Coverage—Comparison of Greedy and Modified Dynamic Programming Approaches

Beata Zielosko

Abstract In the paper, a modification of a dynamic programming algorithm for optimization of decision rules relative to coverage is proposed. Experimental results with decision tables from UCI Machine Learning Repository are presented.

Keywords Decision rules · Coverage · Dynamic programming · Greedy algorithm

1 Introduction

Decision rules are popular form of knowledge representation. They are used in many areas connected with knowledge discovery and data mining [6, 14]. Exact decision rules can be overfitted, i.e., dependent essentially on the noise or adjusted too much to the existing examples. Approximate rules have smaller number of attributes usually, so they are better from the point of view of understanding. Classifiers based on approximate decision rules have often better accuracy than classifiers based on exact decision rules. Therefore, approximate decision rules are studied intensively last years [2, 4, 9, 10, 12].

There are different approaches for construction of decision rules, for example, separate and conquer approach [4, 5], Boolean reasoning [9, 11], greedy algorithms [8], dynamic programming approach [2, 17].

There are different rule quality measures that are used for induction or classification tasks [12, 13]. In the paper, the coverage of decision rules is studied. It is a rule's evaluation measure that allows to discover major patterns in the data. Construction and optimization of rules relative to coverage can be considered as important task for knowledge representation.

In the paper, a modification of a dynamic programming algorithm for optimization of decision rules relative to coverage is presented. Dynamic programming approach allows one to obtain optimal decision rules, i.e., rules with the maximum coverage or minimum length. Proposed method of rule induction is based on the analysis of

B. Zielosko (✉)
Institute of Computer Science, University of Silesia, Sosnowiec, Poland
e-mail: beata.zielosko@us.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_55

the directed acyclic graph constructed for a given decision table. Such graph can be huge for larger data sets. The aim of the paper, is to find a heuristic, modification of a dynamic programming algorithm that allows us to obtain values of coverage of decision rules close to optimal ones [17], and the size of the graph (the number of nodes and edges) should be smaller than in case of dynamic programming algorithm.

In [7], it was shown that under some natural assumptions on the class NP, the greedy algorithm is close to the best polynomial approximate algorithms for the minimization of length of decision rules. There is an intuition, that in case of coverage we can have similar situation, so greedy algorithm is considered also in this work. It is obvious, that greedy approach is simpler than dynamic programming approach, the aim is to compare how close is proposed and greedy solution to optimal solution.

To work with approximate decision rules, an uncertainty measure $G(T)$ is used. It is a difference between number of rows in a given decision table and the number of rows labeled with the most common decision for this table divided by the number of rows in decision table. A threshold γ, $0 \leq \gamma < 1$, is fixed and so-called γ-decision rules, that localize rows in subtables which uncertainty is at most γ, are studied.

Presented algorithm is based on a dynamic programming algorithm for decision rules optimization relative to coverage [17]. For a given decision table T, a directed acyclic graph $\Delta_\gamma(T)$ is constructed. Nodes of this graph are subtables of a decision table T described by descriptors (pairs attribute = value). The partitioning of a subtable is finished when its uncertainty is at most γ. In [17], subtables of the directed acyclic graph were constructed for each value of each attribute from T. In the presented approach, subtables of the graph $\Delta_\gamma(T)$ are constructed for one attribute from T with the minimum number of values, and for the rest of attributes from T—the most frequent value of each attribute (value of an attribute attached to the maximum number of rows), is chosen. So, the size of the graph $\Delta_\gamma(T)$ (the number of nodes and edges) is smaller than the size of the graph constructed by the dynamic programming algorithm. This fact is important from the point of view of scalability. Based on the graph $\Delta_\gamma(T)$, sets of γ-decision rules for rows of table T, are described. Then, using procedure of optimization of the graph $\Delta_\gamma(T)$ relative to coverage, it is possible to find, for each row r of T, a γ-decision rule with the maximum coverage. These values of coverage are compared with the optimal ones obtained using dynamic programming algorithm. In [15], modified dynamic programming algorithm was studied but another uncertainty measure $R(T)$, which is the number of unordered pairs of rows with different decisions in the decision table T, was used. This paper contains also comparison of the coverage of exact decision rules based on another modification of the dynamic programming algorithm. In [16], modified dynamic programming algorithm was studied but uncertainty measure $J(T)$, which is a difference between number of rows in a given decision table and the number of rows labeled with the most common decision for this table, was used. In the present work, modified dynamic programming algorithm is studied and uncertainty measure $G(T)$ is used. The paper contains comparison of values of coverage of decision rules constructed by modified algorithm and greedy algorithm, and comparison of the size of the directed acyclic constructed by dynamic programming algorithm and proposed algorithm.

The paper consists of six sections. Section 2 contains main notions connected with a decision table and decision rules. In Sect. 3, proposed algorithm for construction of a directed acyclic graph is presented. Section 4 contains a description of a procedure of optimization relative to coverage. In Sect. 5, a greedy algorithm for γ-decision rules construction is presented. Section 6 contains experimental results with decision tables from UCI Machine Learning Repository, and Sect. 7—conclusions.

2 Main Notions

In this section, notions corresponding to decision tables and decision rules are presented.

A *decision table* T is a rectangular table with n columns labeled with conditional attributes $f_1, \ldots, f_n$. Rows of this table are filled with nonnegative integers that are interpreted as values of conditional attributes. Rows of T are pairwise different and each row is labeled with a nonnegative integer (decision) that is interpreted as a value of a decision attribute.

A minimum decision value which is attached to the maximum number of rows in T will be called the *most common decision for* T.

By $N(T)$ the number of rows in table T is denoted and by $N_{mcd}(T)$ the number of rows in the table T labeled with the most common decision for T is denoted. The value $G(T) = N(T) - N_{mcd}(T)/N(T)$ will be interpreted as *uncertainty* of the table T.

The table T is called *degenerate* if T is empty or all rows of T are labeled with the same decision, in this case, $G(T) = 0$.

A table obtained from T by the removal of some rows is called a *subtable* of the table T. Let T be nonempty, $f_{i_1}, \ldots, f_{i_s} \in \{f_1, \ldots, f_n\}$ and $a_1, \ldots, a_s$ be nonnegative integers. By $T(f_{i_1}, a_1) \ldots (f_{i_s}, a_s)$ the subtable of the table T is denoted. It contains only rows that have numbers $a_1, \ldots, a_s$ at the intersection with columns $f_{i_1}, \ldots, f_{i_s}$. Such nonempty subtables (including the table T) are called *separable subtables* of T.

An attribute $f_i \in \{f_1, \ldots, f_n\}$ is *not constant* on T if it has at least two different values. For the attribute that is not constant on T it is possible to find *the most frequent value*. It is an attribute's value attached to the maximum number of rows in T.

The set of attributes from $\{f_1, \ldots, f_n\}$ which are not constant on T is denoted by $E(T)$. For any $f_i \in E(T)$, the set of values of the attribute f_i in T is denoted by $E(T, f_i)$. If $f_i \in E(T)$ is the attribute with the most frequent value then $E(T, f_i)$ contains only one element.

The expression

$$f_{i_1} = a_1 \wedge \ldots \wedge f_{i_s} = a_s \rightarrow d \tag{1}$$

is called a *decision rule over* T if $f_{i_1}, \ldots, f_{i_s} \in \{f_1, \ldots, f_n\}$, and $a_1, \ldots a_s, d$ are nonnegative integers. It is possible that $s = 0$. In this case (1) is equal to the rule

$$\rightarrow d. \tag{2}$$

Let $r = (b_1, \ldots, b_n)$ be a row of T. The rule (1) will be called *realizable for* r, if $a_1 = b_{i_1}, \ldots, a_s = b_{i_s}$. If $s = 0$ then the rule (2) is realizable for any row from T.

Let γ be a nonnegative real number, $0 \leq \gamma < 1$. The rule (1) is called γ-*true for* T if d is the most common decision for $T' = T(f_{i_1}, a_1) \ldots (f_{i_s}, a_s)$ and $G(T') \leq \gamma$. If $s = 0$ then the rule (2) is γ-true for T if d is the most common decision for T and $G(T) \leq \gamma$.

If the rule (1) is γ-true for T and realizable for r, it will be called a γ-*decision rule for* T *and* r. Note that if $\gamma = 0$ it is an exact decision rule for T and r.

Let τ be a decision rule over T and τ be equal to (1). The *coverage* of τ is the number of rows in T for which τ is realizable and which are labeled with the decision d. It is denoted by $c(\tau)$. If $s = 0$ then $c(\tau)$ is equal to the number of rows in T which are labeled with decision d.

3 Algorithm for Directed Acyclic Graph Construction

In this section, a modification of the dynamic programming algorithm that construct, for a given decision table T, a *directed acycylic graph* $\Delta_\gamma(T)$ is presented. Based on this graph it is possible to describe the set of decision rules for T and each row r of T. Nodes of the graph are separable subtables of the table T. At each step, the algorithm processes one node and marks it with the symbol *. At the first step, the algorithm constructs a graph containing a single node T that is not marked with *.

Let the algorithm have already performed p steps. Now, the step $(p + 1)$ will be described. If all nodes are marked with the symbol * as processed, the algorithm finishes its work and presents the resulting graph as $\Delta_\gamma(T)$. Otherwise, choose a node (table) Θ, that has not been processed yet. Let d be the most common decision for Θ. If $G(\Theta) \leq \gamma$ label the considered node with the decision d, mark it with symbol * and proceed to the step $(p+2)$. If $G(\Theta) > \gamma$ then for each attribute $f_i \in E(\Theta)$, draw a bundle of edges from the node Θ if f_i is the attribute with the minimum number of values. If f_i is the attribute with the most frequent value draw one edge from the node Θ. Let f_i be the attribute with the minimum number of values and $E(\Theta, f_i) = \{b_1, \ldots, b_t\}$. Then draw t edges from Θ and label them with pairs $(f_i, b_1) \ldots (f_i, b_t)$ respectively. These edges enter to nodes $\Theta(f_i, b_1), \ldots, \Theta(f_i, b_t)$. For the rest of attributes from $E(\Theta)$ draw one edge, for each attribute, from the node Θ and label it with pair (f_i, b_1), where b_1 is the most frequent value of the attribute f_i. This edge enters to a node $\Theta(f_i, b_1)$. If some of nodes $\Theta(f_i, b_1), \ldots, \Theta(f_i, b_t)$ are absent in the graph then add these nodes to the graph. Each row r of Θ is labeled with the set of attributes $E_{\Delta_\gamma(T)}(\Theta, r) \subseteq E(\Theta)$, some attributes from this set can be removed

later during a procedure of optimization. The node Θ is marked with the symbol * and proceed to the step $(p + 2)$.

The graph $\Delta_\gamma(T)$ is a directed acyclic graph. A node of such graph will be called *terminal* if it does not have outgoing edges. Note that a node Θ of $\Delta_\gamma(T)$ is terminal if and only if $G(\Theta) \leq \gamma$.

In the next section, a procedure of optimization of the graph $\Delta_\gamma(T)$ relative to the coverage will be described. As a result, a graph G is obtained with the same sets of nodes and edges as in $\Delta_\gamma(T)$. The only difference is that any row r of each nonterminal node Θ of G is labeled with a nonempty set of attributes $E_G(\Theta, r) \subseteq E(\Theta)$. It is possible also that $G = \Delta_\gamma(T)$.

Now, for each node Θ of G and for each row r of Θ, a set of γ-decision rules $Rul_G(\Theta, r)$ will be described. The algorithm starts from terminal nodes of G and moves to the node T.

Let Θ be a terminal node of G labeled with the most common decision d for Θ. Then $Rul_G(\Theta, r) = \{\rightarrow d\}$.

Let now Θ be a nonterminal node of G such that for each child Θ' of Θ and for each row r' of Θ', the set of rules $Rul_G(\Theta', r')$ is already defined. Let $r = (b_1, \ldots, b_n)$ be a row of Θ. For any $f_i \in E_G(\Theta, r)$, the set of rules $Rul_G(\Theta, r, f_i)$ is defined as follows:

$$Rul_G(\Theta, r, f_i) = \{f_i = b_i \wedge \sigma \rightarrow k : \sigma \rightarrow k \in Rul_G(\Theta(f_i, b_i), r)\}.$$

Then $Rul_G(\Theta, r) = \bigcup_{f_i \in E_G(\Theta, r)} Rul_G(\Theta, r, f_i)$.

To illustrate the presented algorithm a decision table T_0 depicted on the top of Fig. 1 is considered. The value $\gamma = 0.5$, so during the construction of the graph $\Delta_{0.5}(T_0)$ the partitioning of a subtable Θ of T_0 ends if $G(\Theta) \leq 0.5$. The graph is denoted by $G = \Delta_{0.5}(T_0)$. Now, for each node Θ of the graph G and for each row r of Θ the set $Rul_G(\Theta, r)$ will be described, starting from terminal nodes. Terminal nodes of the graph G are $\Theta_1, \Theta_2, \Theta_3, \Theta_4$. For these nodes: $Rul_G(\Theta_1, r_1) = Rul_G(\Theta_1, r_2) = Rul_G(\Theta_1, r_3) = \{\rightarrow 2\}$; $Rul_G(\Theta_2, r_4) = Rul_G(\Theta_2, r_5) = \{\rightarrow 1\}$; $Rul_G(\Theta_3, r_1) = Rul_G(\Theta_3, r_3) = Rul_G(\Theta_3, r_4) = \{\rightarrow 3\}$; $Rul_G(\Theta_4, r_2) = Rul_G(\Theta_4, r_3) = Rul_G(\Theta_4, r_4) = Rul_G(\Theta_4, r_5) = \{\rightarrow 3\}$.

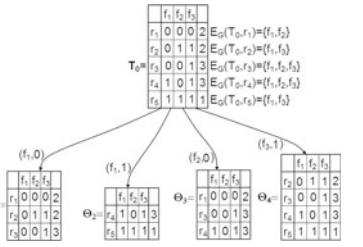


Fig. 1 Directed acyclic graph $G = \Delta_{0.5}(T_0)$

Now, the sets of rules attached to rows of T_0 are described: $Rul_G(T_0, r_1) = \{f_1 = 0 \to 2, f_2 = 0 \to 3\}$; $Rul_G(T_0, r_2) = \{f_1 = 0 \to 2, f_3 = 1 \to 3\}$; $Rul_G(T_0, r_3) = \{f_1 = 0 \to 2, f_2 = 0 \to 3, f_3 = 1 \to 3\}$; $Rul_G(T_0, r_4) = \{f_1 = 1 \to 1, f_2 = 0 \to 3, f_3 = 1 \to 3\}$; $Rul_G(T_0, r_5) = \{f_1 = 1 \to 1, f_3 = 1 \to 3\}$.

4 Procedure of Optimization Relative to Coverage

In this section, a procedure of optimization of the graph G relative to the coverage c is presented. The algorithm moves from the terminal nodes of the graph G to the node T. It will assign to each row r of each table Θ the set $Rul^c_G(\Theta, r)$ of γ-decision rules with the maximum coverage from $Rul_G(\Theta, r)$, the number $Opt^c_G(\Theta, r)$ – the maximum coverage of a γ-decision rule from $Rul_G(\Theta, r)$, and it will change the set $E_G(\Theta, r)$ attached to the row r in the nonterminal table Θ. The obtained graph is denoted by G^c.

Let Θ be a terminal node of G and d be the most common decision for Θ. Then the number $Opt^c_G(\Theta, r)$ is assigned to each row r of Θ. It is equal to the number of rows in Θ which are labeled with the decision d.

Let Θ be a nonterminal node of G and all children of Θ have already been treated. Let $r = (b_1, \ldots, b_n)$ be a row of Θ. The number $Opt^c_G(\Theta, r) = \max\{Opt^c_G(\Theta(f_i, b_i), r) : f_i \in E_G(\Theta, r)\}$ is assigned to the row r in the table Θ and set $E_{G^c}(\Theta, r) = \{f_i : f_i \in E_G(\Theta, r), Opt^c_G(\Theta(f_i, b_i), r) = Opt^c_G(\Theta, r)\}$.

Below you can find sets $Rul^c_G(T_0, r_i)$, $i = 1, \ldots, 5$, of γ-decision rules for T_0 (depicted on the top of Fig. 1) and r_i, with the maximum coverage, and the value $Opt^c_G(T, r_i)$. It is equal to the maximum coverage of γ-decision rule for T_0 and r_i, and it was obtained during the procedure of optimization of the graph G relative to the coverage.

$Rul_G(T_0, r_1) = \{f_1 = 0 \to 2, f_2 = 0 \to 3\}$, $Opt^c_G(T_0, r_1) = 2$;
$Rul_G(T_0, r_2) = \{f_1 = 0 \to 2, f_3 = 1 \to 3\}$, $Opt^c_G(T_0, r_2) = 2$;
$Rul_G(T_0, r_3) = \{f_1 = 0 \to 2, f_2 = 0 \to 3, f_3 = 1 \to 3\}$, $Opt^c_G(T, r_3) = 2$;
$Rul_G(T_0, r_4) = \{f_2 = 0 \to 3, f_3 = 1 \to 3\}$, $Opt^c_G(T, r_4) = 2$;
$Rul_G(T_0, r_5) = \{f_3 = 1 \to 3\}$, $Opt^c_G(T, r_5) = 2$.

5 Greedy Algorithm

In this section, a greedy algorithm for γ-decision rule construction is presented (see Algorithm 1). At each iteration, an attribute $f_i \in \{f_1, \ldots, f_n\}$ with minimum index is selected, such that uncertainty $G(T)$ of corresponding subtable is minimum. The algorithm is applied sequentially to each row r of the table T. As a result, for each row r of T, a γ-decision rule is obtained.

Algorithm 1 Greedy algorithm for γ-decision rule construction

Require: Decision table T with conditional attributes $f_1, \ldots, f_n$, row $r = (b_1, \ldots, b_n)$ of T, and real number γ, $0 \leq \gamma < 1$.
Ensure: γ-decision rule for T and r.
$Q \leftarrow \emptyset$;
$T' \leftarrow T$;
while $G(T') > \gamma$ **do**
  select $f_i \in \{f_1, \ldots, f_n\}$ with the minimum index such that $G(T'(f_i, b_i))$ is minimum;
  $T' \leftarrow T'(f_i, b_i)$;
  $Q \leftarrow Q \cup \{f_i\}$;
end while
$\bigwedge_{f_i \in Q}(f_i = b_i) \rightarrow d$, where d is the most common decision for T'.

6 Experimental Results

Experiments were done on decision tables from UCI Machine Learning Repository [3]. Some decision tables contain conditional attributes that take unique value for each row. Such attributes were removed. In some tables there were equal rows with, possibly, different decisions. In this case, each group of identical rows was replaced with a single row from the group with the most common decision for this group. In some tables there were missing values. Each such value was replaced with the most common value of the corresponding attribute. Let T be one of these decision tables. For this table values of γ from the set $\Gamma(T) = \{G(T) \times 0.001, G(T) \times 0.01, G(T) \times 0.1, G(T) \times 0.2\}$, were considered.

For each such decision table T, using modified dynamic programming algorithm, the directed acyclic graph $\Delta_\gamma(T)$ was constructed. Then, optimization relative to coverage was applied. For each row r of T, the maximum coverage of a γ-decision rule for T and r was obtained. After that, for rows of T, the average coverage of rules with the maximum coverage—one for each row, was calculated.

Tables 1 and 2 present the average coverage of γ-decision rules. Column *attr* contains the number of attributes in T, column *rows*—the number of rows in T. Values of the average coverage of γ-decision rules constructed by the proposed algorithm are contained in a column *mod*, values of γ-decision rules constructed by the dynamic programming algorithm are contained in the column *dp*, and values of γ-decision rules constructed by the greedy algorithm are contained in the column *greedy*. Comparisons with optimal values (obtained by the dynamic programming algorithm), for modified and greedy algorithms, are presented in columns *diff mod* and *diff greedy* respectively. It is a relative difference which is equal to (*Opt_Coverage - Coverage)/Opt_Coverage*, where *Opt_Coverage* denotes the average coverage of γ-decision rules constructed by the dynamic programming algorithm, *Coverage* denotes the average coverage of γ-decision rules constructed by the proposed algorithm (in a case of column *mod*) and greedy algorithm (in a case of column *greedy*). Based on the average relative difference it is possible to see how close on average coverage is proposed and greedy solution to optimal solution. Values in bold

Table 1 Average coverage of γ-decision rules for $\gamma \in \{G(T) \times 0.001, G(T) \times 0.01\}$

Decision table	Attr	Rows	$\gamma = G(T) \times 0.001$					$\gamma = G(T) \times 0.01$				
			Mod	dp	Greedy	Diff mod	Diff greedy	Mod	dp	Greedy	Diff mod	Diff greedy
Adult-stretch	4	16	6.25	7.00	6.25	0.11	0.11	6.25	7.00	6.25	0.11	0.11
Balance-scale	4	625	3.07	4.21	3.71	0.27	0.12	3.07	4.21	3.71	0.27	0.12
Breast-cancer	9	266	6.15	9.53	4.26	0.35	0.55	6.15	9.53	4.26	0.35	0.55
Cars	6	1728	325.58	332.76	331.41	**0.02**	0.00	325.58	332.76	331.41	**0.02**	0.00
Lymphography	18	148	20.69	21.54	9.55	0.04	0.56	20.69	21.54	9.55	0.04	0.56
Monks-1-test	6	432	33.50	45.00	36.00	0.26	0.20	33.50	45.00	36.00	0.26	0.20
Monks-2-train	6	169	4.32	6.38	3.89	0.32	0.39	4.32	6.38	3.89	0.32	0.39
Nursery	8	12960	1483.58	1531.04	1508.20	0.03	0.01	1483.58	1531.04	1508.20	0.03	0.01
Shuttle-landingl	6	15	1.80	2.13	1.87	0.15	0.12	1.80	2.13	1.87	0.15	0.12
Soybean-small	35	47	12.53	12.53	8.89	**0.00**	0.29	12.53	12.53	8.89	**0.00**	0.29
Teeth	8	23	1.00	1.00	1.00	**0.00**	0.00	1.00	1.00	1.00	**0.00**	0.00
Average						0.14	0.21				0.14	0.21

Table 2 Average coverage of γ-decision rules for $\gamma \in \{G(T) \times 0.1, G(T) \times 0.2\}$

Decision table	Attr	Rows	$\gamma = G(T) \times 0.1$					$\gamma = G(T) \times 0.2$				
			Mod	dp	Greedy	Diff mod	Diff greedy	Mod	dp	Greedy	Diff mod	Diff greedy
Adult-stretch	4	16	6.25	7.00	6.25	0.11	0.11	6.25	7.00	6.25	0.11	0.11
Balance-scale	4	625	3.07	10.94	3.71	0.72	0.66	10.43	13.85	3.71	0.25	0.73
Breast-cancer	9	266	6.15	9.53	4.26	0.35	0.55	7.56	12.27	4.26	0.38	0.65
Cars	6	1728	325.58	332.76	331.41	**0.02**	0.00	325.58	332.82	331.41	**0.02**	0.00
Lymphography	18	148	20.69	24.38	9.55	0.15	0.61	35.84	36.84	9.55	0.03	0.74
Monks-1-test	6	432	33.50	45.00	36.00	0.26	0.20	33.50	45.00	36.00	0.26	0.20
Monks-2-train	6	169	4.32	6.38	3.89	0.32	0.39	4.87	7.30	3.89	0.33	0.47
Nursery	8	12960	1483.58	1602.99	1508.20	0.07	0.06	1485.89	1663.75	1508.20	0.11	0.09
Shuttle-landing	6	15	1.80	2.13	1.87	0.15	0.12	1.80	2.13	1.87	0.15	0.12
Soybean-small	35	47	12.53	12.83	8.89	**0.02**	0.31	12.53	12.83	8.89	**0.02**	0.31
Teeth	8	23	1.00	1.00	1.00	**0.00**	0.00	1.00	1.00	1.00	**0.00**	0.00
Average						0.20	0.27				0.15	0.31

Table 3 Size of the directed acyclic graph for $\gamma \in \{G(T) \times 0.001, G(T) \times 0.01\}$

Decision table	$G(T) \times 0.001$				$G(T) \times 0.01$			
	nd	edg	nd-dp	edg-dp	nd	edg	nd-dp	edg-dp
Adult-stretch	36	37	72	108	36	37	72	108
Balance-scale	654	808	1212	3420	654	808	1212	3420
Breast-cancer	2483	9218	6001	60387	2483	9218	6001	60387
Cars	799	1133	7007	19886	799	1133	7007	19886
Lymphography	26844	209196	40928	814815	26844	209196	40928	814815
Monks-1-test	568	694	2772	7878	568	694	2772	7878
Monks-2-train	632	1277	1515	6800	632	1277	1515	6800
Nursery	18620	27826	115200	434338	18620	27826	115200	434338
Shuttle-landing	78	257	85	513	78	257	85	513
Soybean-small	3023	38489	3592	103520	3023	38489	3592	103520
Teeth	118	446	135	1075	118	446	135	1075

denote that the relative difference regarding to average coverage of rules, for proposed algorithm and dynamic programming algorithm, is equal to zero or close to zero, and there exists a difference regarding to the size of the directed acycylic graph (see Table 5). The last row in Tables 1 and 2, presents the average value of the relative difference for considered decision tables. These values show, that on average, proposed approach allows to obtain values of coverage of constructed rules closer to optimal ones than greedy approach.

Tables 3 and 4 present a size of the directed acyclic graph, i.e., number of nodes (column *nd*) and number of edges (column *edg*) in the graph constructed by the

Table 4 Size of the directed acyclic graph for $\gamma \in \{G(T) \times 0.1, G(T) \times 0.2\}$

Decision table	$G(T) \times 0.1$				$G(T) \times 0.2$			
	nd	edg	nd-dp	edg-dp	nd	edg	nd-dp	edg-dp
Adult-stretch	36	37	72	108	36	37	72	108
Balance-scale	625	727	1204	3300	622	710	1200	3220
Breast-cancer	2483	9218	6001	60387	2480	9201	6001	60186
Cars	799	1133	7007	19886	794	1123	7002	19866
Lymphography	26832	208936	40925	813980	26362	202043	40779	791308
Monks-1-test	568	694	2772	7878	568	694	2772	7878
Monks-2-train	632	1277	1515	6800	631	1272	1515	6769
Nursery	18572	27558	115200	428129	18386	27226	115200	416387
Shuttle-landing	78	257	85	513	78	257	85	513
Soybean-small	3023	38413	3592	103351	3020	38039	3592	102523
Teeth	118	446	135	1075	118	446	135	1075

Table 5 Comparison of the size of the directed acyclic graph

Decision table	$G(T) \times 0.001$		$G(T) \times 0.01$		$G(T) \times 0.1$		$G(T) \times 0.2$	
	nd diff	edg diff	nd diff	edg diff	nd diff	edg diff	nd diff	edg diff
Adult-stretch	2.00	2.92	2.00	2.92	2.00	2.92	2.00	2.92
Balance-scale	1.85	4.23	1.85	4.23	1.93	4.54	1.93	4.54
Breast-cancer	2.42	6.55	2.42	6.55	2.42	6.55	2.42	6.54
Cars	8.77	17.55	8.77	17.55	8.77	17.55	8.82	17.69
Lymphography	1.52	3.89	1.52	3.89	1.53	3.90	1.55	3.92
Monks-1-test	4.88	11.35	4.88	11.35	4.88	11.35	4.88	11.35
Monks-2-train	2.40	5.32	2.40	5.32	2.40	5.32	2.40	5.32
Nursery	6.19	15.61	6.19	15.61	6.20	15.54	6.27	15.29
Shuttle-landing	1.09	2.00	1.09	2.00	1.09	2.00	1.09	2.00
Soybean-small	1.19	2.69	1.19	2.69	1.19	2.69	1.19	2.70
Teeth	1.14	2.41	1.14	2.41	1.14	2.41	1.14	2.41
Average	3.04	6.78	3.04	6.78	3.05	6.80	3.06	6.79

proposed algorithm and dynamic programming algorithm (columns *nd-dp* and *edg-dp* respectively).

Table 5 presents comparison of the number of nodes (column *nd diff*) and number of edges (column *edg diff*) of the directed acyclic graph. Values of these columns are equal to the number of nodes/edges in the directed acyclic graph constructed by the dynamic programming algorithm divided by the number of nodes/edges in the directed acyclic graph constructed by the proposed algorithm. Presented results show that the size of the directed acyclic graph constructed by the proposed algorithm is smaller than the size of the directed acyclic graph constructed by the dynamic programming algorithm. In particular, for the data sets "soybean-small", "teeth" and "cars", the results of the average coverage are almost the same (see Tables 1 and 2) but there exists a difference relative to the number of nodes (more than one time) and relative to the number of edges (more than two times).

Experiments were done using software system Dagger [1] which is implemented in C++ and uses Pthreads and MPI libraries for managing threads and processes respectively, on computer with i5-3230M processor and 8 GB of RAM.

7 Conclusions

A modification of the dynamic programming algorithm for optimization of approximate decision rules relative to the coverage was presented. Using relative difference on average coverage of γ-decision rules it was possible to see that on average, proposed approach allows to obtain values of coverage of constructed rules closer to optimal ones than greedy approach. The size of the directed acyclic graph constructed

by the proposed algorithm, for all data sets and $\gamma \in \Gamma(T)$, is smaller than the size of the directed acyclic graph constructed by the dynamic programming algorithm, and in the case of edges, the difference is at least two times. In the future works, accuracy of rule based classifiers using considered algorithms will be compared.

References

1. Alkhalid, A., Amin, T., Chikalov, I., Hussain, S., Moshkov, M., Zielosko, B.: Dagger: A tool for analysis and optimization of decision trees and rules. In: Computational Informatics. Social Factors and New Information Technologies: Hypermedia Perspectives and Avant-Garde Experiences in the Era of Communicability Expansion, pp. 29–39. Blue Herons, Bergamo (2011)
2. Amin, T., Chikalov, I., Moshkov, M., Zielosko, B.: Dynamic programming approach to optimization of approximate decision rules. Inf. Sci. **221**, 403–418 (2013)
3. Asuncion, A., Newman, D.J.: UCI Machine Learning Repository (2007). http://www.ics.uci.edu/~mlearn/
4. Błaszczyński, J., Słowiński, R.: Szeląg, M.: Sequential covering rule induction algorithm for variable consistency rough set approaches. Inf. Sci. **181**(5), 987–1002 (2011)
5. Dembczyński, K., Kotłowski, W., Słowiński, R.: ENDER: a statistical framework for boosting decision rules. Data Min. Knowl. Disc. **21**(1), 52–90 (2010)
6. Han, J., Kamber, M.: Data Mining: Concepts and Techniques. Morgan Kaufmann, San Francisco (2000)
7. Moshkov, M., Piliszczuk, M., Zielosko, B.: Partial Covers, Reducts and Decision Rules in Rough Sets - Theory and Applications, SCI, vol. 145. Springer-Verlag, Berlin Heidelberg, Germany (2008)
8. Moshkov, M., Zielosko, B.: Combinatorial Machine Learning - A Rough Set Approach, SCI, vol. 360. Springer, Berlin Heidelberg, Germany (2011)
9. Nguyen, H.S.: Approximate boolean reasoning: foundations and applications in data mining. In: Peters, J.F., Skowron, A. (eds.) Transactions on Rough Sets V, LNCS, vol. 4100, pp. 334–506. Springer, Berlin Heidelberg, Germany (2006)
10. Nowak, A., Zielosko, B.: Clustering of partial decision rules. In: Cyran, K.A., Kozielski, S., Peters, J.F., Stańczyk, U., Wakulicz-Deja, A. (eds.) Man-Machine Interactions, AISC, vol. 59, pp. 183–190. Springer, Berlin Heidelberg, Germany (2009)
11. Pawlak, Z., Skowron, A.: Rough sets and Boolean reasoning. Inf. Sci. **177**(1), 41–73 (2007)
12. Sikora, M., Wróbel, Ł.: Data-driven adaptive selection of rule quality measures for improving rule induction and filtration algorithms. Int J. Gen. Syst. **42**(6), 594–613 (2013)
13. Stańczyk, U.: Decision rule length as a basis for evaluation of attribute relevance. J. Intell. Fuzzy Syst. **24**(3), 429–445 (2013)
14. Stefanowski, J., Vanderpooten, D.: Induction of decision rules in classification and discovery-oriented perspectives. Int. J. Intell. Syst. **16**(1), 13–27 (2001)
15. Zielosko, B.: Optimization of approximate decision rules relative to coverage. In: Kozielski, S., Mrózek, D., Kasprowski, P., Małysiak-Mrózek, B., Kostrzewa, D. (eds.) Beyond Databases, Architectures, and Structures, CCIS, vol. 424, pp. 170–179. Springer, Switzerland (2014)
16. Zielosko, B.: Optimization of decision rules relative to coverage - comparative study. In: Kryszkiewicz, M., Cornelis, C., Ciucci, D., Medina-Moreno, J., Motoda, H., Raś, Z.W. (eds.) Rough Sets and Intelligent Systems Paradigms, LNCS, vol. 8537, pp. 237–247. Springer, Switzerland (2014)
17. Zielosko, B., Chikalov, I., Moshkov, M., Amin, T.: Optimization of decision rules based on dynamic programming approach. In: Faucher, C., Jain, L.C. (eds.) Innovations in Intelligent Machines-4–Recent Advances in Knowledge Engineering, SCI, vol. 514, pp. 369–392. Springer, Switzerland (2014)

Part XI
Computer Networks and Mobile Technologies

Characteristic of User Generated Load in Mobile Gaming Environment

Krzysztof Grochla, Wojciech Borczyk, Maciej Rostanski and Rafal Koffer

Abstract This paper describes an analysis of the variability of the load imposed by users of a mobile gaming platform. We measure the communication between the sample mobile gaming application and characterize the types of requests that are being transmitted to the cloud service. We characterize the variability of the load in time, finding weekly and daily patterns of load and differences between working days and weekends of approximately 20 %. We analyze the correlations between the processing of the different types of requests, and find that the processing time correlates with the total load observed on server, but is only partially related to the type of request being processed.

Keywords Mobile gaming · Performance evaluation · Sesion length · Load estimation

1 Introduction

The mobile devices like smartphones and tablets have become omnipresent in the current society. The mobile phones have been designed mainly for communication, but currently are being used also for entertainment, as a platform to play music, display video files or execute games. The market of games for mobile devices is experiencing very fast growth: according to the Newzoo Global Games Market Report [12] the games for smarthphones and tablets share approximately 20 % of the global gaming market, and its value is about to rise by 47 % for tablets and 18 % for smartphones

K. Grochla (✉)
Institute of Theoretical and Applied Informatics, PAS, Gliwice, Poland
e-mail: kgrochla@iitis.pl

W. Borczyk · R. Koffer
Incuvo, Katowice, Poland
e-mail: wborczyk@incuvo.com

M. Rostanski
University of Dabrowa Gornicza, Dąbrowa Górnicza, Poland
e-mail: mrostanski@wsb.edu.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_56

between 2012 and 2016. It shows that the gaming is moving from consoles to the mobile segment, as users interact with the mobile devices both in home and on the move.

The mobile gaming industry is becoming a significant part of the traffic in wireless mobile networks. Thanks to nearly constant availability of the internet connection the mobile games are not only executed locally, but also communicate with other players' devices and with the server infrastructure. It allows to enrich the gaming experience, creating multiplayer games and allows the users to share the user-generated content. Thus the user is becoming not only the consumer of the data, but also creates the content which s used by other users. The way the users interact with the game, how much of the data they want to share and in what times of the day they play influences the load imposed on both the network and the servers creating the infrastructure for the game.

The mobile games are facing many challenges related to the limited resources of the mobile devices in terms of e.g. battery life, storage and bandwidth [15]. The cloud based, distributed systems are popular as a server side system [3] or middleware for mobile applications [14]. Large number of mobile games assumes interactions between users and require constant connectivity with the server to synchronize what the users are doing in the virtual environment and load the elements of the virtual environment as the user is progressing within the game. The appropriate performance of the backend system is crucial for the perception of the game, the pauses caused by a delay in downloading are hardly acceptable by the users, thus the servers must not be overloaded [13]. On the other hand, the cost of virtual machines lease must be minimized, thus the appropriate scaling of the backend infrastructure is needed.

The patterns of mobile device usage during the day and in different days of the week change [8]. It is more likely that people will use mobile games in the free time or while they commute [17]. This behavior influences the load imposed on the server infrastructure. To correctly estimate the load in a mobile gaming cloud system the analysis how it changes during the day and during is needed. This paper presents the results of such measurements for a sample mobile gaming platform. The rest of the paper is organized as follows: in Sect. 2 we describe the gaming platform for which we have executed the measurements, next we describe the results and analyze the observed variability of the number of users, sessions, requests and other load metrics. We finish with a short conclusion.

2 Architecture of the Evaluated System

The evaluation of the gaming platform has been performed on the Createria [11], which is a quick and easy to use platform for creating, sharing and playing games that turns making games into a game. It allows the users to create games with just a few taps without any programming skills, share them with friends or simply play other people creations. The Createria is based on the multi-tier architecture, in which presentation, application processing, and data management functions are physically

separated. The game on the mobile device works as a client, sending HTTP requests to the cloud server when new data is needed (e.g. a user enters onto new level) or some data need to be stored (e.g. the user needs to save high score). This type of communication is typical for the mobile application, according to [10] 97 % of the traffic in modern mobile networks is transmitted through TCP and HTTP. The server side application has separated functions of data processing and storage. The cache is used on each of the layers to minimize the amount of communication needed and improve the game response time.

3 Characteristic of the Mobile Gaming User Behavior

The load of the mobile gaming platform depends on the number of users playing the game simultaneously and the number of requests coming from the games to the backend system. We concentrate on the evaluation of the number of requests, without direct analysis of the user mobility and behavior, as it is difficult to monitor the users' location and this problem has been analyzed in multiple other works—see e.g. [9]. To characterize how the load on the system changes in time, we have implemented a custom tool that monitors the amount of requests received by the backend cloud system in time. We have also measured the time required to process each request and the number of sessions established between the mobile application. The measurements have been executed over two weeks period, the representative subset of the data, which allowed to anonimize the results.

3.1 Session Variability

The Fig. 1 shows the experimental probability density function, calculated from the histogram of the number of concurrent session in the system. We can observe that it follows the normal distribution. The standard deviation is equal to the 30 % of the

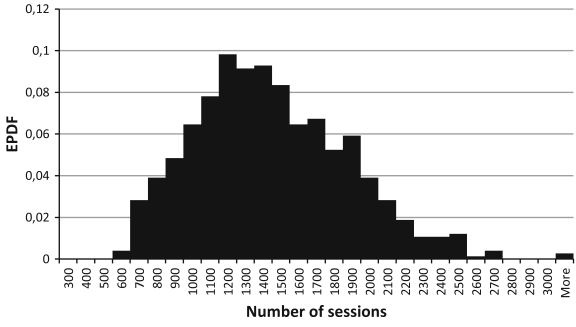


Fig. 1 Experimental probability distribution (EPDF) of number of concurrent sessions in the system

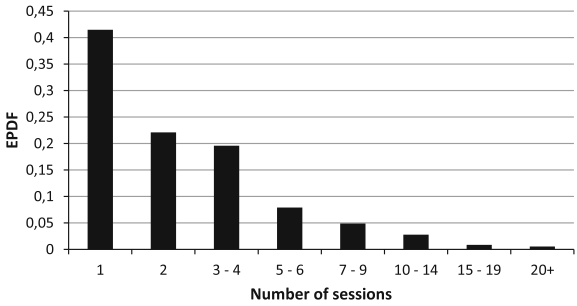


Fig. 2 EPDF of number of session per day for a single user

mean, what shows that the number of the users using the mobile gaming platform does not change significantly over the evaluated period.

Next we evaluated how many sessions a single user starts per day. The Fig. 2 shows that for the analyzed platform around 40 % of the users use the game only once per day. However most of the users return to the game more than once during a day, some do this 15 or even more times per day. The changes in the characteristic of the number of sessions per day are presented on Fig. 3, where daily histograms are shown. It can be seen that the distribution is stable and the amount of users using the application changes mainly among those who use the game only once or twice per day.

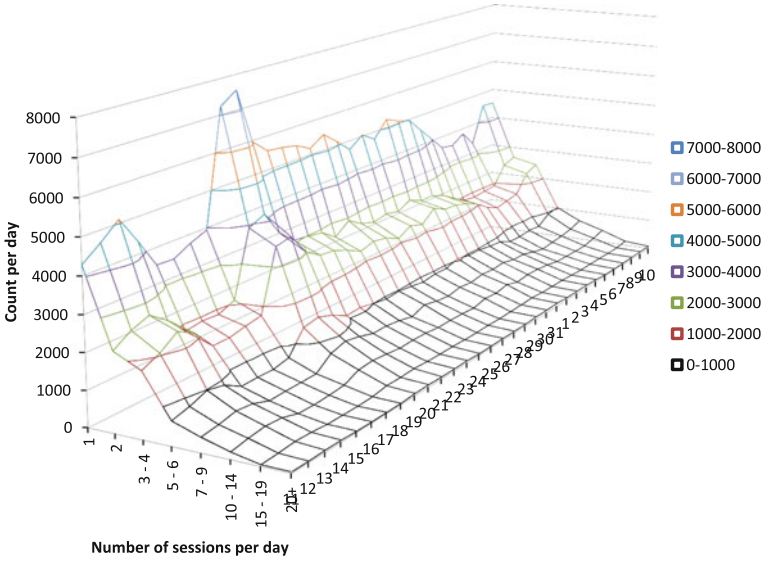


Fig. 3 Changes of number of session per day over a month period

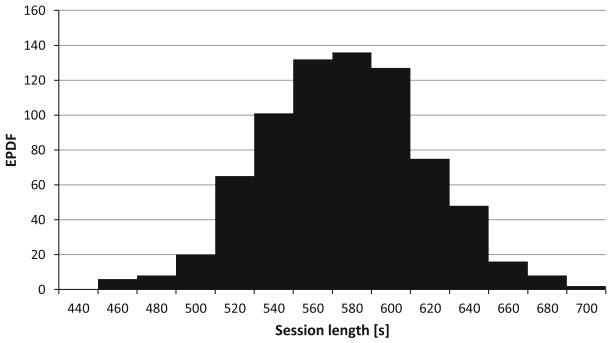


Fig. 4 Distribution of session length

The distribution of a hourly medians of session length is shown on the Fig. 4. The average session lasted almost 10 min, what is quite long for a mobile game—according to [5] the average session length for IOS games is lower than 7 min.

4 Weekly and Daily Patterns

To characterize the load variability in time we have measured the average number of requests transmitted to the servers depending on the day of the week. The measurement are presented on Fig. 5 and show an increase of load during the weekend of 10–20 %. There are no significant changes in load between work days, although on Friday the number of requests coming from the mobile application is slightly higher. We have also measured the load changes during the 15 consecutive days, starting

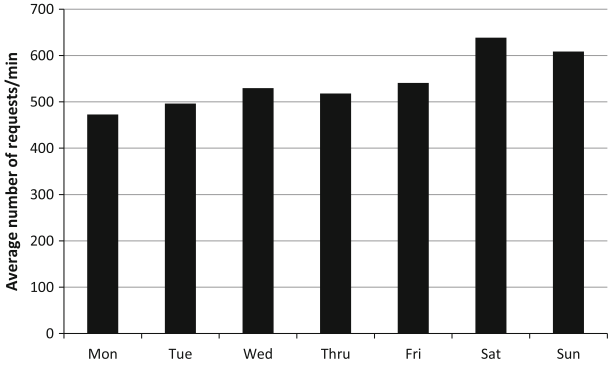


Fig. 5 Characteristic of load changes during a week

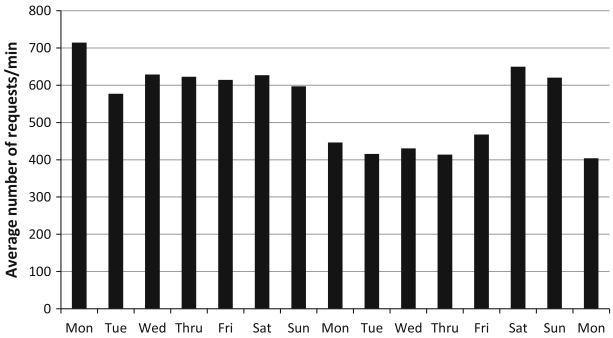


Fig. 6 Characteristic of load changes for different days during the analysed period

from Monday Dec 22th 2014 and end on Monday Jan 5th 2015. The load is considerably higher during the Christmas time, the difference is around 33 % comparing to the next week (Fig. 6).

The variability in hourly load averages is higher than between days of the week, as can be seen on Fig. 7. The evaluated gaming platform is available globally, but the majority of the players come from North America, thus the time on the plot is shown in PDT time zone. The load imposed during the afternoon peak of gaming activity is approximately 75 % higher than the minimum, observed in early morning hours. The peak in the load corresponds with the patterns of smartphone and tablet usage shown in [6], where the maximum of usage was also observed in the afternoon and the minimum between 4 and 5 am. However the difference between maximum and minimum is much lower (on the global data approximately 6 times more users are using the mobile devices in the afternoon than during the night), probably because the global availability of the analyzed platform.

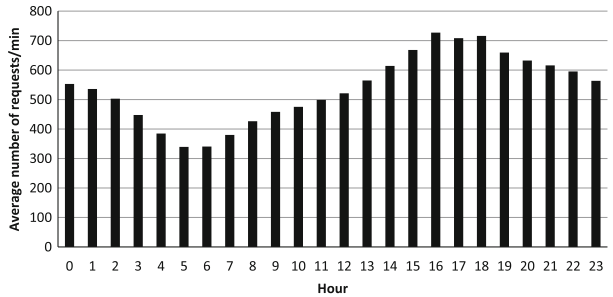


Fig. 7 Characteristic of hourly load changes

5 Load Characteristics

The communication protocol implemented in the analyzed mobile gaming platform is based on HTTP requests transmitted by the application to download more data needed during the game or post some information to the server (like e.g. high score). The use of HTTP based communication is common for mobile gaming, because of the availability of the libraries to support this protocol in both IOS and Android OSes and because this protocol is accepted by virtually every network operator and is rarely blocked by the firewalls. To characterize the network generated by the gaming activity we have identified 15 types of requests which are transmitted during the communication between the mobile application and the backend servers. We have measured the time required to process each of those 15 types of requests. The histograms of the time required to respond to a sample request type is shown on Fig. 8. It can be seen that it is a heavy tailed distribution, with most of the probability mass concentrated around 10–12 s, while there is a very low number of events when the request is processed for a longer time than 40 s.

To measure the correlation between the processing time for different types of requests we have monitored the 10 s average time required for each type of the request to finish. The correlation between two sample types of requests is presented on the scatterplot on Fig. 9. We can see that the processing time was heavily correlated—the correlation coefficient was equal 0.784 between those two applications and had very similar value when calculated between any of the 15 types of requests used by the analyzed mobile gaming platform. This is especially the case for the longest processing time observed, what corresponds to periods when the backend server is overloaded. The time required to respond to a request was in a very small part dependent on the type of the request and the processing time is dependent mainly on the temporal server load, not on the operation character. It also proves that when the server is overloaded every types of requests are degraded in very similar manner. However there were some periods in time where one of the types of requests required longer processing, while the second was quicker—this may be caused by the types of data requested from the database, which in some cases required longer access time.

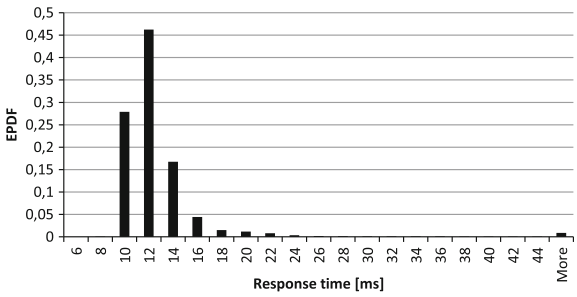


Fig. 8 Distribution of the processing time of requests sent from the mobile application

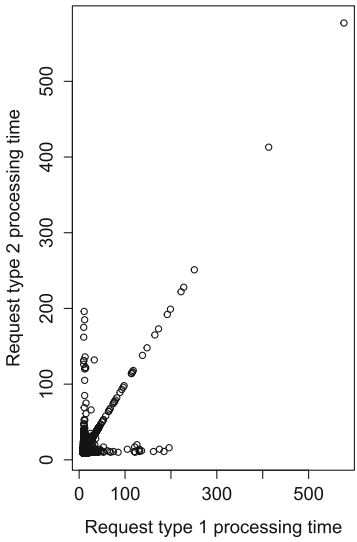


Fig. 9 Correlation between the processing time for two different types of requests

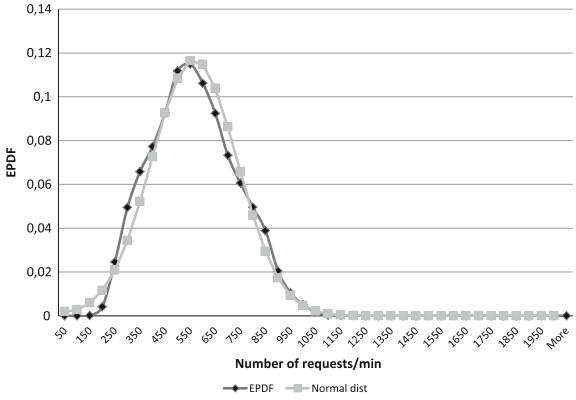


Fig. 10 Distribution of number of requests

We have also analyzed the distribution of the total amount of requests transmitted between application and the server. The Fig. 10 shows that it can be very well approximated by the the normal distribution.

6 Related Work

The patterns of mobile application usage have been studied in a few research papers. Böhmer et al. [2] present detailed application usage analysis from over 4100 users. They show similar daily usage patterns, with smallest number of users at 4–5 am and highest load in afternoon hours, but the difference between highest and lowest load is

much higher than in application platform evaluated by us. Similar daily load pattern is also observed in [1], where differences between addicted and regular users are analyzed. In [16] a prediction model for application usage patterns is developed, but it is dedicated to emulate switching between different applications, not estimation of the load in a mobile game. The load patterns in mobile gaming did not show the self-similar characteristics observed in the general internet traffic [4].

The session times for different types of applications have been evaluated in [7], where behavior of 21 participants was analyzed. The session time was changing heaving depending on the type of application used, with more than 40 % of application used for less than 15 s. In [18], also based on a small number of participants (25), 50 % of application session time is under 30 s and 90 % under 4 min. We have observed session times much longer than the average of almost 10 min, what is in line with the larger usage time reported for mobile games in previous works [5].

7 Conclusions

The measured characteristic of load between a mobile gaming application and the backend servers show strong correlation between the time of day and the load in mobile gaming platform. The load in different days of the week also changes noticeably, with load on weekend larger by more than 20 %. The distribution of changes in observed load shows that it follows normal distribution. The processing time has a long tail distribution and changes in similar way for all types of requests served by the same server.

Acknowledgments This work is in part supported by Polish National Centre for Research and Development under the grant No. INNOTECH-K3/HI3/20/228040/NCBR/14.

References

1. Ahn, H., Wijaya, M.E., Esmero, B.C.: A systemic smartphone usage pattern analysis: focusing on smartphone addiction issue. Int. J. Multimedia Ubiquit. Eng. **9**(6) (2014)
2. Böhmer, M., Hecht, B., Schöning, J., Krüger, A., Bauer, G.: Falling asleep with angry birds, facebook and kindle: a large scale study on mobile application usage. In: MobileHCI 2011, pp. 47–56. Stockholm (2011)
3. Dinh, H.T., Lee, C., Niyato, D., Wang, P.: A survey of mobile cloud computing: architecture, applications, and approaches. Wireless Commun. Mobile Comput. **13**(18), 1587–1611 (2013)
4. Domańska, J., Domańska, A., Czachórski, T.: A few investigations of long-range dependence in network traffic. In: Czachórski, T., Gelenbe, E., Lent, R. (eds.) Information Sciences and Systems 2014, pp. 137–144. Springer, Switzerland (2014)
5. Farago, P.: Flurry Presents Apps by the Numbers (2011). http://www.slideshare.net/marksilva/flurry-presents-apps-by-the-numbers
6. Farago, P.: The Truth About Cats and Dogs: Smartphone vs Tablet Usage Differences (2012). http://www.flurry.com/bid/90987/The-Truth-About-Cats-and-Dogs-Smartphone-vs-Tablet-Usage-Differences#.VQMI7eFldTs

7. Ferreira, D., Goncalves, J., Kostakos, V., Barkhuus, L., Dey, A.K.: Contextual experience sampling of mobile application micro-usage. In: MobileHCI 2014, pp. 91–100. Toronto, Canada (2014)
8. Gorawski, M., Grochla, K.: The real-life mobility model: RLMM. In: FGCT 2013, pp. 201–206. London (2013)
9. Gorawski, M., Grochla, K.: Review of mobility models for performance evaluation of wireless networks. In: Gruca, A., Czachorski, T., Kozielski, S. (eds.) Man-Machine Interactions 3, AISC, vol. 242, pp. 567–577. Springer, Switzerland (2014)
10. Huang, J., Qian, F., Gerber, A., Mao, Z.M., Sen, S., Spatscheck, O.: A close examination of performance and power characteristics of 4G LTE networks. In: MobiSys 2012, pp. 225–238. Low Wood Bay (2012)
11. Incuvo: Createrria, http://incuvo.com/createrria/
12. Newzoo: Newzoo Global Games Market Report (2013). http://venturebeat.com/2013/06/06/global-games-market-to-hit-86-1b-by-2016-as-mobile-charges-ahead/
13. Rostański, M., Buchwald, P., Arkadiusz, J.: Relative and non-relative databases performance with an android platform application. Theor. Appl. Inform. **25**(3–4), 224–238 (2013)
14. Rostanski, M., Grochla, K., Seman, A.: Evaluation of highly available and fault-tolerant middleware clustered architectures using RabbitMQ. In: FedCSIS 2014, pp. 879–884. Poland (2014)
15. Satyanarayanan, M.: Fundamental challenges in mobile computing. In: PODC 1996, pp. 1–7. Philadelphia (1996)
16. Shin, C., Hong, J.H., Dey, A.K.: Understanding and prediction of mobile application usage for smart phones. In: Ubicomp 2012, pp. 173–182. Pittsburgh (2012)
17. Skelley, T., Namoun, A., Mehandjiev, N.: The impact of a mobile information system on changing travel behaviour and improving travel experience. In: Mobile Web Information Systems, pp. 233–247. Springer (2013)
18. Yan, T., Chu, D., Ganesan, D., Kansal, A., Liu, J.: Fast app launching for mobile devices using predictive user context. In: MobiSys 2012, pp. 113–126. Low Wood Bay (2012)

Using Kalman Filters on GPS Tracks

Krzysztof Grochla and Konrad Połys

Abstract This paper describes the practical evaluation of the application of the Kalman filters to GPS tracks, gathered by mobile phones or GPS trackers. We try to answer the question whenever the filtering applied on higher layer of the mobile device software may improve the quality of the data provided by the GPS receiver. Two metrics are used for comparison: the average euclidean distance between the points on the GPS tracks and the actual location of the user and the area of the polygon created by intersection of the filtered and real track. We find that the Kalman filtering does not improve those two metrics and the direct use of the data provided by the GPS receiver provides track which is on average more near the real path than result of Kalman filtering. However we observe that this is caused by the errors introduced when the user change the direction and when we evaluate a parts of the path without rapid changes of direction (as e.g. crossing) the filters allow to generate the points which are more near the road taken by the user.

Keywords GPS tracking · Kalman filtering · Smartphone tracking · Mobility monitoring

1 Introduction

The growth in availability of mobile devices equipped with GPS (Global Positioning System) receivers facilitated the development and popularity of applications and services based on location of users. Applications on mobile devices can help us to navigate to any place, measure the traveled distance or show the way to nearest ATM. A large number of dedicated devices allowing to gather the GPS tracks are available on the market, of which some even have been patented [4]. The monitoring of the GPS location is important in multiple research projects, starting from monitoring

K. Grochla (✉) · K. Połys
Institute of Theoretical and Applied Informatics, PAS, Gliwice, Poland
e-mail: kgrochla@iitis.pl

K. Połys
e-mail: kpolys@iitis.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_57

of rare species (see e.g. [9] or [16]), through mobility modeling, to performance evaluation of wireless network [8].

The GPS tracking devices and applications periodically denote the location of the device, creating a set of records that contain three values: timestamp, latitude and longitude. The applications that use the location information on smartphones and other mobile devices typically rely on the information provided by the GPS receiver. The GPS receiver is a hardware device generating a stream of data compatible with the NMEA standard [2]. The NMEA stream may passed directly to an application to be parsed within it or may be processed by the device's operating system to provide the location information. The data provided by the GPS receiver are generated using a sophisticated signal processing algorithms to provide as precise location as possible. To the best authors knowledge, all GPS solutions available in smartphones and mobile tracking devices are proprietary, so detailed information on the filters applied within the hardware and firmware of such devices is not publicly available. But the location information provided by the GPS often fluctuates, even for a stationary device, and is subject to multiple sources of noise, such as e.g. the ionospheric irregularities [13].

In this paper we try to answer the question whenever the filtering applied on higher layer of the mobile device software (within the tracking application) may improve the quality of the data provided by the GPS receiver. We have selected Kalman filter as the most popular and widely used for the GPS tracks. We aim in comparing the quality of the data read directly from the GPS received with the tracks after filtering.

The rest of the paper is organized as follows: in the second section we present short literature review; next we describe the methods used to gather sample GPS tracks and the metrics used to compare them; in the following section we present the results and we finish with a short conclusion.

1.1 Review of the Literature

Kalman filtering, also known as linear quadratic estimation (LQE), have been proposed in 1960 in [10]. The filter uses a series of measurements observed over time, containing noise and other inaccuracies, and produces estimates of unknown variables that tend to be more precise than those based on a single measurement alone. The algorithm works in a two-step process. In the prediction step, the Kalman filter produces estimates of the current state variables, along with their uncertainties. When the next measurement (including some errors and random noise) are available, these estimates are updated using a weighted average, with more weight being given to estimates with higher certainty [11]. The Kalman filer may run in real time using only the present input measurements and the previously calculated state and its uncertainty matrix; no additional past information is required [15].

The problem of filtering the GPS data has been considered in multiple papers, starting from the 1990s. Mohamed and Schwartz use adaptive Kalman filtering for integrated inertial navigation system and GPS [12]. In [3] a method and system for accurately determining the position coordinates of a mobile GPS receiver by

resolving the double difference GPS carrier phase integer ambiguity has been proposed. The Kalman filters are often applied to vector-tracking GPS [6] or estimation of vehicle parameters [5].

2 Gathering and Filtering the GPS Tracks

We have developed a custom application for mobile phones with GPS receivers, called BX-Tracker [7]. This application with specially created algorithms is able to continuously, without noticeable battery drain, monitor the location of user. Raw data gathered from this application were analyzed and filtered for this research. For the filtering purpose were used the GPS tracks collected from different (brand and model) Android smartphones with BX-Tracker application. As an example were used two tracks for which gathered the most number of repetitions.

To evaluate the effect of tracks filtering we have developed application in Python. This application has implemented following features: removing positions with very low accuracy (we posited more than 400 m), duplicated entries (with the same position and time), removing entry when it's time is incorrect (eg. year 1970, in this case the position was wrong also), removing entries with impossible speed or acceleration (we assumed the top speed is 180 km/h and max acceleration 4 m/s in 2 s) and Kalman filter. We have used the implementation of the Kalman filters according to the book [5], using the Python code published in [14]. To measure how much the tracks are approximate to the reference track used another Python application which calculate average distance between tracks, area between them and draw graphic representation.

2.1 Comparison of the GPS Tracks

The definition of the metric describing the quality of the GPS track is crucial to compare two GPS tracks. The Fréchet distance is a commonly used measure of similarity between curves that takes into account the location and ordering of the points along the curves [1]. If we imagine a dog walking on one path and a person with a leash walking on the second, the Fréchet distance of two curves is the minimal length of any leash necessary for the dog and the person to move from the starting points of the two curves to their respective endpoints. The drawback of the Fréchet distance is that it measures only the distance in the most pessimistic location and does not represent the average error between points gathered by the GPS.

The area of the figure defined by the two curves is another method to estimate the average distance between them. To measure it we have linked the firsts and lasts points of the two tracks and calculated the area of the polygon created (or sum of area of all polygons created when the curves were crossing each other). This metric, known as area of overlap, is used in shape matching [17]. The smaller the area is the more near are two GPS tracks.

In case of GPS track we have a set of points which define the track and we are most interested in having those points as near to the real location as possible. For such problem we were able to calculate the distance between each of the points on the track and the nearest point of the ideal track, representing the real location of the user. Thus we were able to calculate the average euclidean distance between the tracks and we have used it as a second metric in our study.

3 Influence of Filtering on Sample GPS Tracks

3.1 Fitting of Kalman Filters

The selection of the parameters of Kalman filters can significantly influence the resulting quality of the achieved signal [18]. We have taken an iterative approach, trying to rerun the filtering and evaluate the quality of the generated path. We have used the two metrics described above: the average distance between the ideal path and the path generated by filtering and the area of overlap. To compare the results three similar tracks were needed—raw track, filtered one and the reference track (Fig. 1). The reference track was drawn on the map by precise representation of

Fig. 1 Tracks example—*green line* is the reference track, *blue line* is the raw track and the *red* is a filtered track, map ©OpenStreetMap contributors

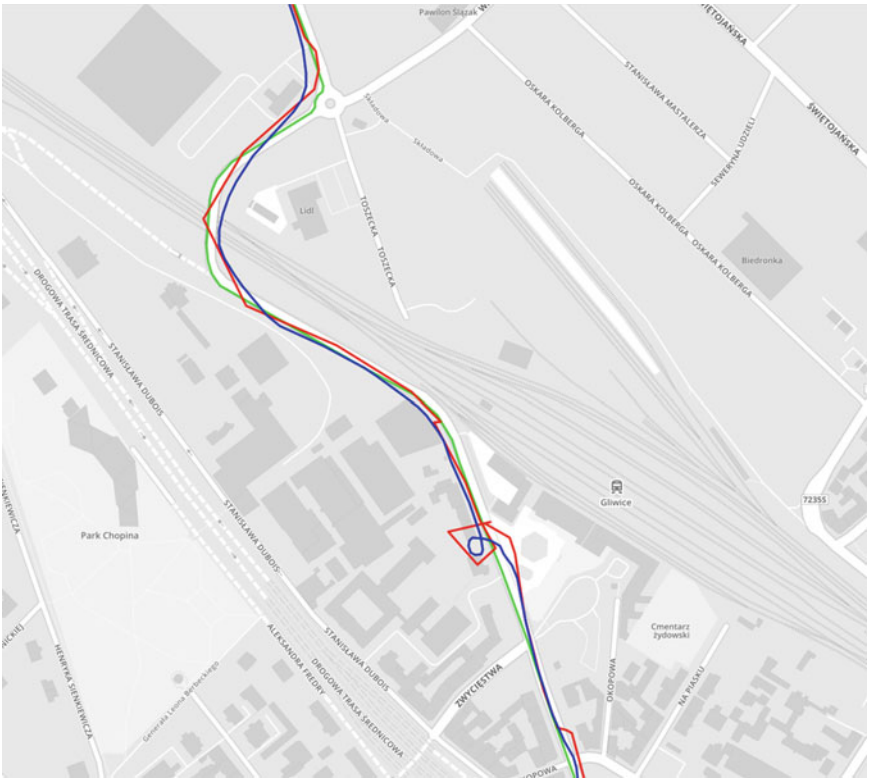


Fig. 2 Tracks example—*green line* is the reference track, *blue line* is the raw track and the *red* is a filtered track, map ©OpenStreetMap contributors

the traveled path on the map. Figure 2 depicts recorded raw track (blue line) with false loop near traffic lights, the red line is a filtered track and the green line is a reference track. The OpenStreetMap data were used as a reference background. Figure 3 depicts difference between raw track and reference track (on the left) and filtered and reference track (on the right).

The direct comparison of the measured metric showed that there were no such parameters of the Kalman filter for which the filtered path was more near the reference path than the path directly read from the GPS receiver. The results of the comparison based on two metrics: average distance between paths and the area of polygon created by the two paths, are presented in Table 1. For the parameters showing the best fit the average distance between the reference and the raw track was equal 5.31 m; for the filtered track it was equal 5.61 m. The areas of the polygon are respectively: 9190.99 and 9710.88 m^2. As the result, according to both metrics defined the filtered track was slightly less close to the reference track than the original (Fig. 4).

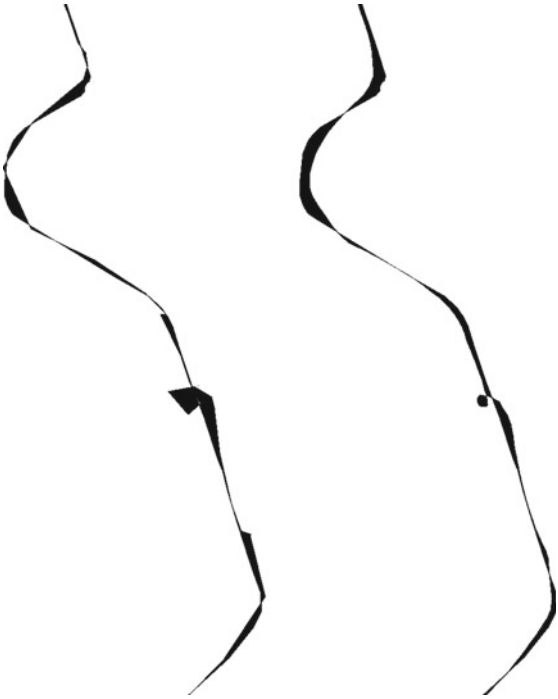


Fig. 3 Difference between: raw track and reference track—on the *left*, filtered and reference track—on the *right*

Table 1 Comparison of original and filtered path accuracy

	Average distance (m)	Area of polygon (m^2)
Original path	5.31	9190.99
Filtered path	5.61	9710.88

3.2 Comparison of the Filtered and Non-filtered Tracks

The increase of the average distance to the reference path and the area of the intersection figure can be explained by the tendency of the filtering to introduce errors when the user change the direction of movement. On Fig. 1 we can see a closeup of the location near a crossing, where a rapid change of direction took place. In such a place the filtered track is shown as a curve not reaching the location of the crossing, smoothed before the most far point of the track. A similar situation can be observed on another example near different crossing on Fig. 5. Another example of a situation in which the filtering introduces errors is a part of the path with multiple changes of direction in a short time, as it is presented in Fig. 4—in such a case the filters tend to straighten the path too much, partially removing some of the short turns.

The Kalman filtering, apart from increasing the average distance to the reference track, had a number of positive results. In a part of the path where the noise in GPS

Fig. 4 Comparison of filtered and unfiltered path with multiple changes of direction, map © OpenStreetMap contributors

signal generated some change on location when a device was not moving for some time (see the small loop in Fig. 2) the deviation from the original location is much smaller for the filtered path. The Kalman filters are also able to improve the situation when an error or noise causes a short time deviation of the signal from the real location, as it is presented on Fig. 6. In such a case the filtered path deviates less from the road taken by the user.

The tuning of parameters of the Kalman filters allows to select whenever the path stays more near the locations gathered by the GPS, or whenever it more noise prone and more smooth. The comparison of paths for different parameters is shown on Fig. 7 for the same location as on Fig. 1. The number of data points in time also allows to improve the quality of the GPS track, at the cost of the increased battery usage.

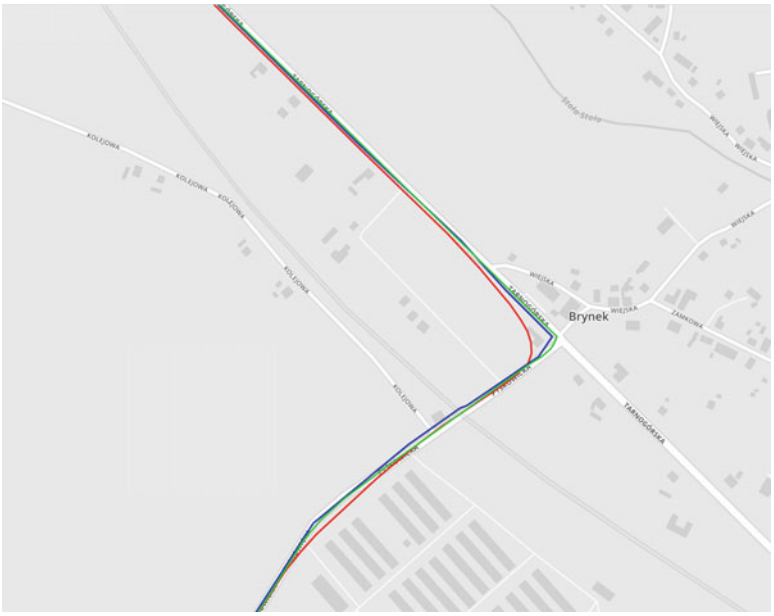


Fig. 5 Filtered and unfiltered path near a sample crossing, map ©OpenStreetMap contributors

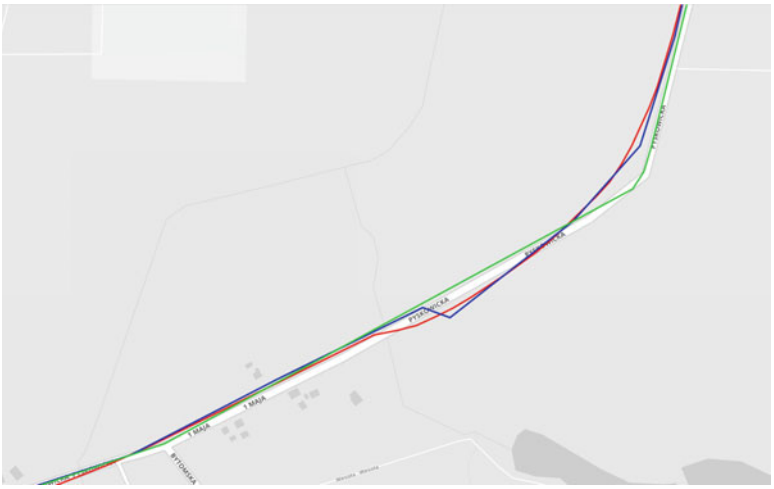


Fig. 6 Filtered and unfiltered path when error in GPS signal reception caused deviation from the real path, map ©OpenStreetMap contributors

Fig. 7 Filtered paths for different sets of parameters, map ©OpenStreetMap contributors

4 Conclusions

Kalman filtering is suitable for applications where input data are noisy. When it is used with GPS coordinates on the smartphone with small sample rate, the received track will be much smoother, less precise but more visual attractive than raw track. They allow to realize filtering functions such as: removing positions with low accuracy, duplicated entries, entries with incorrect time, impossible speed or acceleration. However the application of filtering increases the average distance between the monitored path and a reference path, thus introduce a small error in the location information.

References

1. Alt, H., Godau, M.: Computing the fréchet distance between two polygonal curves. Int. J. Comput. Geom. Appl **5**(1–2), 75–91 (1995)
2. Association, N.M.E., et al.: NMEA 0183–Standard for Interfacing Marine Electronic Devices. NMEA, Severna Park (2002)
3. Babu, B.: Method and system for resolving double difference gps carrier phase integer ambiguity utilizing decentralized Kalman filters, Patent US 5,451,964 (1995)
4. Baker, D.: Portable self-contained tracking unit and GPS tracking system. Patent US 6,339,397 (2002)
5. Burkul, S.R., Pawar, P.R., Jagtap, K.R.: Estimation of vehicle parameters using Kalman filter: review. Int. J. Curr. Eng. Technol. **4**(4), 2731–2735 (2014)

6. Chen, X., Wang, X., Xu, Y.: Performance enhancement for a gps vector-tracking loop utilizing an adaptive iterated extended Kalman filter. Sensors **14**(12), 23630–23649 (2014)
7. Foremski, P., Gorawski, M., Grochla, K.: Energy-efficient crowdsensing of human mobility and signal levels in cellular networks. Sensors (2015)
8. Gorawski, M., Grochla, K.: Review of mobility models for performance evaluation of wireless networks. In: Gruca, A., Czachórski, T., Kozielski, S. (eds.) Man-Machine Interactions 3, AISC, vol. 242, pp. 567–577. Springer, Switzerland (2014)
9. Grémillet, D., Dell'Omo, G., Ryan, P.G., Peters, G., Ropert-Coudert, Y., Weeks, S.J.: Offshore diplomacy, or how seabirds mitigate intra-specific competition: a case study based on GPS tracking of cape gannets from neighbouring colonies. Mar. Ecol. Prog. Ser. **268**, 265–279 (2004)
10. Kalman, R.E.: A new approach to linear filtering and prediction problems. J. Fluids Eng. **82**(1), 35–45 (1960)
11. Marsland, S.: Machine Learning: An Algorithmic Perspective. CRC Press, Boca Raton (2014)
12. Mohamed, A., Schwarz, K.: Adaptive Kalman filtering for INS/GPS. J. geodesy **73**(4), 193–203 (1999)
13. Pi, X., Mannucci, A., Lindqwister, U., Ho, C.: Monitoring of global ionospheric irregularities using the worldwide GPS network. Geophys. Res. Lett. **24**(18), 2283–2286 (1997)
14. Pilkington, N.: Kalman filter python implementation. https://github.com/nickponline/snippets/blob/master/kalman-filter.py
15. Rauh, A., Butt, S.S., Aschemann, H.: Nonlinear state observers and extended Kalman filters for battery systems. Int. J. Appl. Math. Comput. Sci. **23**(3), 539–556 (2013)
16. Schofield, G., Bishop, C.M., MacLean, G., Brown, P., Baker, M., Katselidis, K.A., Dimopoulos, P., Pantis, J.D., Hays, G.C.: Novel GPS tracking of sea turtles as a tool for conservation management. J. Exp. Mar. Biol. Ecol. **347**(1), 58–68 (2007)
17. Veltkamp, R.C., Hagedoorn, M.: State of the art in shape matching. In: Lew, M.S. (ed.) Principles of Visual Information Retrieval, pp. 87–119. APR, Springer, London (2001)
18. Yuen, K.V., Hoi, K.I., Mok, K.M.: Selection of noise parameters for Kalman filter. Earthq. Eng. Eng. Vib. **6**(1), 49–56 (2007)

Stability Analysis and Simulation of a State-Dependent Transmission Rate System

Evsey Morozov, Lyubov Potakhina and Alexander Rumyantsev

Abstract In this paper, we consider a model of communication system with state-dependent service rate. This mechanism allows to change service rate to increase the efficiency of the system. Motivation of such a system is discussed as well. Then we present the regenerative proof of the sufficient stability conditions of the system which is based on the negative drift of the workload process above a high threshold. Moreover, we describe a wireless communication system in which transmission rate is being both Markov-modulated and also queue-dependent. Simulation results are presented, which illustrate the behavior of the queue size depending on the used threshold-based service rate switching mechanism.

Keywords Queue-dependent service rate · Stability conditions · Regeneration · Markov-modulated transmission rates · Simulation

1 Introduction

In this paper, we consider a class of state-dependent service rate models. Our analysis, being mainly theoretical, focuses both on the stability of a wide class of state-dependent systems and on the validation of the stability condition by simulation of a particular model. In part, a motivation of this research comes from the analysis of a wireless communication system with Markov-modulated transmission rates. In this system, the so-called *best rate (BR) users* (having the highest transmission

E. Morozov (✉) · L. Potakhina · A. Rumyantsev
Institute of Applied Mathematical Research, Karelian Research Centre, Petrozavodsk, Russia
e-mail: emorozov@karelia.ru

L. Potakhina
e-mail: lpotahina@gmail.com

A. Rumyantsev
e-mail: ar0@karelia.ru

E. Morozov · L. Potakhina · A. Rumyantsev
Petrozavodsk State University, Petrozavodsk, Russia

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_58

rate) have an absolute priority, and the queue-based control is *indirect*: the bigger queue size is, the bigger the probability to have at least one BR user in the system becomes [13]. In this work we generalize this system allowing the service rates to be also dependent on the current queue size. This generalization opens a new promising opportunity to control state-dependent communication and computer systems and is closely connected with the so-called *green computing*. Indeed, the constantly high energy consumption of contemporary data centers dramatically contrasts with under-utilization of the majority of them. This motivates the hardware vendors to develop different technologies allowing the data center owner to reduce his power budget. In particular, these technologies allow switching between the full capacity utilization [12] and the low-power states, either on the CPU level [5, 10, 11], or on the level of the machine and it's components [9]. However, operating these tools may cause a performance degradation and, in particular, a temporary instability (for instance, unacceptable increase of the queue size). Thus, it is crucially important, keeping the system within the stability region, to find an energy saving scenario while operating in non-full capacity utilization regime. It is especially important for energy-consuming high performance computing clusters (HPC). On the other hand, a HPC is controlled by a queue manager system which, as a rule, has information on the current queue size. Thus, depending on the current queue size, the manager may set the CPU frequency to minimize the energy consumption. (In this regard see [1, 7, 8].)

In summary, the main contribution of this work is the new regenerative proof of the stability condition of the queue-dependent service rate system. Another contribution is description of a generalized wireless system with Markov-modulated and queue-dependent transmission rates.

The rest of the paper is organized as follows. In Sect. 2, we give the regenerative proof of the sufficient stability conditions of the queue-dependent system. In Sect. 3, we describe a multi-class wireless system with Markov-modulated transmission rates. Then we generalize this system to capture a dependence between transmission rates and current queue size. In Sect. 4, simulation of the queue-size process in a particular state-dependent system is presented.

2 Stability Analysis of Queue-Dependent System

In this section, we present the regenerative proof of the sufficient stability conditions of the queue-dependent system. We consider an initially empty FIFO $GI/G/1$-type system with the renewal input with arrival instants $\{t_n,\ n \geq 0\}$, the queue size $\nu(t)$ (the number of customers in the system at instant t) and M thresholds,

$$0 = x_0 < x_1 < \cdots < x_M < x_{M+1} := \infty, \tag{1}$$

such that, if the queue size $\nu_n := \nu(t_n^-) \in [x_i,\ x_{i+1})$, then the service time S_n of customer n is selected from an i.i.d. sequence $\{S_n^{(i)},\ n \geq 0\}$ with generic element $S^{(i)}$ and expectation $0 < \mathsf{E}S^{(i)} < \infty$, $i = 0, \ldots, M$. Also we denote by W_n the waiting time of customer n in the queue. The waiting time sequence $\{W_n\}$ satisfies Lindley's recursion

$$W_{n+1} = \left(W_n + S_n - \tau_n\right)^+, \quad n \geq 0, \tag{2}$$

where $(x)^+ := \max(x,\ 0)$. The regenerations of the described system are defined as

$$\theta_{n+1} = \min\{t_k > \theta_n : W_k = \nu_k = 0\}, \quad n \geq 0 \quad (\theta_0 := 0). \tag{3}$$

Denote generic regeneration period by θ. If $\mathsf{E}\theta < \infty$, then the queueing system is called *positive recurrent*, and it is a crucial fact to establish stability [16]. It is worth mentioning that in general the sequence $\{W_n\}$ defined by (2) *is not a Markov chain*. It is because service time S_n depends on the queue size ν_n (while a dependence S_n on W_n keeps Markov property). It makes stability analysis more challenging and motivates the presence of *solidarity* Theorem 1 below.

Now we formulate and prove stability conditions of the above described queue-dependent system. The analysis in [17, 18] allows to simplify the proofs below. Define the remaining regeneration time at instant t_n as

$$\theta(n) := \min_k(\theta_k - t_n : \theta_k - t_n > 0).$$

The proof is based on the following result [6]:

$$\text{if } \theta(n) \not\Rightarrow \infty \quad \text{(in probability) as } n \to \infty, \quad \text{then } \mathsf{E}\theta < \infty. \tag{4}$$

The following statement expresses a *solidarity property*.

Theorem 1 *In the system, $\nu_n \Rightarrow \infty$ (in probability) if and only if $W_n \Rightarrow \infty$.*

Proof Let S_n be the service time of customer n and $S(t)$ be the remaining service time at instant t^-. Then for any $n \geq 0$, $x \geq 0$ and $k \geq 1$:

$$\begin{aligned}
\mathsf{P}(W_n > x) &= \mathsf{P}\left(\sum_{j=0}^{\nu_n - 1} S_j + S(t_n) > x\right) \\
&\geq \mathsf{P}\left(\sum_{j=0}^{k-1} S_j + S(t_n) > x,\ \nu_n \geq k\right) \\
&\geq \mathsf{P}\left(\sum_{j=0}^{k-1} S_j + S(t_n) > x\right) - \mathsf{P}(\nu_n < k).
\end{aligned} \tag{5}$$

Assume $\nu_n \Rightarrow \infty$. Since $\min_i \mathsf{E}S^{(i)} > 0$, then $\mathsf{E}(\min_i S^{(i)}) > 0$, and by Strong Law of Large Numbers,

$$\frac{\sum_{j=0}^{k-1} S_j}{k} \geq_{st} \frac{\sum_{j=0}^{k-1} \min_{0\leq i\leq M} S_j^{(i)}}{k} \to \mathsf{E}(\min_i S^{(i)}) > 0, \ \ k \to \infty,$$

where $\geq_{st}$ means *stochastic inequality*. Then $\sum_{j=0}^{k-1} S_j \to \infty$ as $k \to \infty$, and k_0 exists such that the probability $\mathsf{P}(\sum_{j=0}^{k_0-1} S_j > x)$ becomes arbitrary close to 1 (for x fixed), while the probability $\mathsf{P}(\nu_n < k_0)$ can be done arbitrary small for n large enough. Hence, it follows from (5) that $W_n \Rightarrow \infty$. Conversely, assume $W_n \Rightarrow \infty$. Since for each x, n, k,

$$\mathsf{P}(W_n > x) \leq \mathsf{P}\left(\sum_{j=0}^{\nu_n-1} S_j + S(t_n) > x, \ \nu_n \geq k\right) + \mathsf{P}\left(\sum_{j=0}^{k-1} S_j + S(t_n) > x\right),$$

then

$$\begin{aligned}\mathsf{P}(\nu_n \geq k) &\geq \mathsf{P}\left(\sum_{i=0}^{\nu_n-1} S_j + S(t_n) > x, \ \nu_n \geq k\right) \\ &\geq \mathsf{P}(W_n > x) - \mathsf{P}\left(\sum_{i=0}^{k-1} S_j + S(t_n) > x\right). \end{aligned} \tag{6}$$

Note that $\max_i \mathsf{E}S^{(i)} < \infty$ and that the sequence of the remaining service times $\{S(t_n), \ n \geq 0\}$ is tight [18], that is, for each $\varepsilon > 0$, a constant D exists such that $\sup_n \mathsf{P}(S(t_n) > D) \leq \varepsilon$. Then, for fixed k, the second term (subtrahend) in (6) can be done arbitrary small for a large enough $x := x_0$. Thus, because $W_n \Rightarrow \infty$, $\mathsf{P}(W_n > x_0)$ becomes arbitrary close to 1 for all n large enough, implying $\mathsf{P}(\nu_n \geq k) \to 1$ as $n \to \infty$, and it completes the proof.

The following statement gives sufficient stability conditions of the system.

Theorem 2 *Assume that* $\mathsf{E}S^{(M)} < \mathsf{E}\tau$ *and that*

$$\min_{0\leq i\leq M} \mathsf{P}(\tau > S^{(i)}) > 0. \tag{7}$$

Then $\mathsf{E}\,\theta < \infty$.

Proof Recall that the waiting times $\{W_n\}$ satisfy Lindley's recursion (2), and note that the increments of the waiting times are upper bounded as

$$\Delta_n := W_{n+1} - W_n \leq \max_{0\leq i\leq M} S_n^{(i)} \leq \sum_{i=0}^{M} S_n^{(i)}.$$

Since the conditional expectation is

$$\mathsf{E}(\Delta_n \mid W_n = x) = \mathsf{E}\Big((x + S_n - \tau_n)^+ - x\Big), \tag{8}$$

then, for an arbitrary $x \geq 0$, we obtain the following upper bound,

$$\begin{aligned}\mathsf{E}\Delta_n &= \mathsf{E}(\Delta_n,\ W_n \leq x) + \mathsf{E}(\Delta_n,\ W > x)\\ &\leq \mathsf{P}(W_n \leq x)\sum_i \mathsf{E}S^{(i)} + \max_{y \geq x} \mathsf{E}\Big((y + S_n - \tau_n)^+ - y\Big)\mathsf{P}(W_n > x). \end{aligned} \tag{9}$$

Assume that $W_n \Rightarrow \infty$, then, by Theorem 1, $\nu_n \Rightarrow \infty$, and it is easy to show that

$$\Big((x + S_n - \tau_n)^+ - x\Big) \Rightarrow S^{(M)} - \tau \ \text{ (in distribution) as } n \to \infty. \tag{10}$$

Because $S_n - \tau_n \leq_{st} \max_{0 \leq i \leq M} S_n^{(i)}$ and

$$\mathsf{E} \max_{0 \leq i \leq M} S^{(i)} \leq \sum_{i=0}^{M} \mathsf{E}S^{(i)} < \infty,$$

then the sequence $\{S_n - \tau_n,\ n \geq 0\}$ is uniformly integrable. Then it follows from (10) that the convergence in average holds,

$$\mathsf{E}\Big((x + S_n - \tau_n)^+ - x\Big) \to \mathsf{E}(S^{(M)} - \tau) < 0,\ \ n \to \infty.$$

As a result, (9) implies $\limsup_{n\to\infty} \mathsf{E}\Delta_n < 0$, and it contradicts the assumption $W_n \Rightarrow \infty$. As a result,

$$\inf_i \mathsf{P}(W_{n_i} \leq D_0) \geq \delta_0 \tag{11}$$

for some constants $\delta_0 > 0,\ D_0 < \infty$ and a deterministic sequence $n_i \to \infty\ (i \to \infty)$. Then, using condition (7) and inequality (11), we show (as in [18] or [17]) that $\mathsf{E}\theta < \infty$.

3 A Wireless System with Markov-Modulated and Queue-Dependent Transmission Rates

The available transmission rates of jobs in wireless networks vary in time due to various reasons (for instance, fading, user mobility, etc.). A base station can estimate the achievable transmission rates with a high precision, and it is desirable to identify good channel-aware schedulers (see Max-Rate scheduler [14], the Proportional

Fair scheduler [3], [15], Relatively Best scheduler [4]). It has been proved that, to maximize the stability region, the base station's resources should be allocated to the jobs with the highest (BR) transmission rate [13]. These works motivate the analysis of the following time-slotted single-server wireless system with Markov-modulated transmission rates as an adequate model of some wireless systems. The number of jobs $A(t)$ arriving at each instant $t = 0, 1, \dots$, form an i.i.d. input sequence with generic element A and the mean $\lambda = \mathsf{E}A \in (0, \infty)$. The job size (service requirement) b of a job is measured in *bits* and has a general distribution with mean $\mathsf{E}b < \infty$. For each job, the channel condition at each slot are governed by a finite irreducible aperiodic Markov chain with an $N \times N$-transition matrix $\boldsymbol{Q}$. When a job is in channel condition k, it receives data at transmission rate r_k. We assume that

$$0 \le r_1 < \cdots < r_N,$$

and denote the ordered vector $r = (r_1, \dots, r_N)$. Then $B := \lceil b/r_N \rceil$ is the number of slots in the BR channel condition needed to complete transmission of the job. Because, in the BR state, the channel uses its full capacity, then the departed work (during any slot $(t,\ t+1)$) equals 1. Define the traffic intensity (in the BR state) as $\rho := \lambda \mathsf{E}B$.

In previous work [13] (also see [19]), stability analysis of the described *original* system (with many servers and K classes of jobs) has been developed under the assumption that the jobs in the BR states have *absolute priority* over the non-BR jobs, while the tie-breaking rule for the non-BR jobs was not specified. It is because that, for the stability analysis, only behavior of the saturated system is important [17, 18]. It follows from [13] that the system with Markov-modulated transmission rates is positive recurrent if $\rho < 1$. (However, in general, it is not enough for stability of the multi-class multi-server system considered in [13].)

Now, to apply previous stability analysis and extend a feedback mechanism, we generalize the original system allowing the transmission rates, being Markov-modulated, also depend on the queue size. Moreover, we focus on the specific state-dependent mechanism *inside the stability region*. In the generalized system, we consider the regions $\Delta(k) := [x_{k-1},\ x_k)$, $k = 1, \dots, M+1$, see (1). Then we realize the state-dependent control by means of the transmission vectors $r^{(1)}, \cdots, r^{(M)}$, such that if, at the service beginning instant of a job, queue size belongs to $\Delta(k)$, then the transmission rates vector becomes (in an evident notation) $r^{(k)} = (r_1^{(k)}, \dots, r_N^{(k)})$. This feedback allows to vary transmission rates to optimize capacity utilization. On the other hand, if we put $B = S^{(M)}$, then condition $\mathsf{E}S^{(M)} < \mathsf{E}\tau$ in Theorem 2 is equivalent to condition $\rho < 1$ in [13], *provided* that the queue size in the generalized system is above the highest threshold x_M. By this reason, in the next section we use condition $\rho < 1$ to stabilize the generalized system in heavy-loaded regime. Note that regenerations of the generalized system are defined by (3). The flexibility of the state-dependent mechanism inside the stability region is the key point for simulation results presented below.

4 Simulation Results

We simulate the system with channel condition states governed by the following $N = 3$-state (aperiodic, irreducible) matrix

$$\boldsymbol{Q} = \begin{pmatrix} 1/2 & 1/2 & 0 \\ 4/9 & 4/9 & 1/9 \\ 1/3 & 1/3 & 1/3 \end{pmatrix},$$

corresponding to a Markov chain. We assume that each newly arriving job is initially in condition 1, and that conditions are changed (accordingly with matrix $\boldsymbol{Q}$) at the end of each slot. The number of jobs arriving in each slot follows a Poisson distribution with parameter λ. The following job size distributions are considered: (i) Pareto distribution $F(x) = 1 - (x_m/x)^{\alpha}$, $x \geq x_m > 0$, $\alpha > 0$, $F(x) = 0$, $x \leq x_m$; and (ii) Weibull distribution $F(x) = 1 - e^{-(x/s)^i}$ with parameters $s > 0$, $i > 0$ and $x \geq 0$. We note that the *heavy-tailed* Pareto distribution is widely used to model traffic of the modern communication systems, while the Weibull distribution covers a wide class of the heavy-tailed and *light-tailed* distributions including exponential. It makes both distributions very useful to model empirical distributions appearing in practice. In all experiments simulation time is 10 000 slots. (Simulation has been carried by means of the system R [20].) Also we consider two thresholds, $x_1 = 100$, $x_2 = 200$ (that is $M = 2$), implying

$$\Delta(1) = [0,\ 100),\ \Delta(2) = [100,\ 200),\ \Delta(3) = [200,\ \infty).$$

Thus $r^{(k)} = (r_1^{(k)}, r_2^{(k)}, r_3^{(k)})$ is the transmission rate vector which is applied just before beginning of a service, if the queue size belongs to $\Delta(k)$, $k = 1, 2, 3$. Because the BR jobs have absolute priority, we select a BR job (if exists) randomly. If no BR jobs exist, then we choose (randomly) a non-BR job. A chosen job is then served with the corresponding transmission rate: if the channel condition for this job is j, and the queue size belongs to $\Delta(k)$, then its transmission rate is $r_j^{(k)}$. Figures illustrate dynamics of the queue size in the generalized system (denoted by (1)), and also in the original system with transmission rate vector $r^{(3)}$ (denoted by (2)).

Figures 1, 2 show dynamics of the queue-size process for the Weibull job size distribution with parameters $s = 7$ and $i = 1.1$, and for two sets of the transmission rates: (i) (2, 3, 4), (6, 5, 7), (6, 8, 10) (Fig. 1), and (ii) (6, 5, 7), (2, 3, 4), (6, 8, 10) (Fig. 2). The input rate is $\lambda = 0.7$, implying $\rho = 0.84$. (Recall that the traffic intensity is calculated for the highest rate r_3 when the queue size is above the largest threshold $x_2 = 200$.) By the selected transmission rates, in scenario (i), queue size grows rapidly in region $\Delta(1)$ and slowly in $\Delta(2)$, while the opposite situation is observed in case (ii). However, when the process exceeds the threshold x_2, the negative drift prevents it further increase in both scenarios.

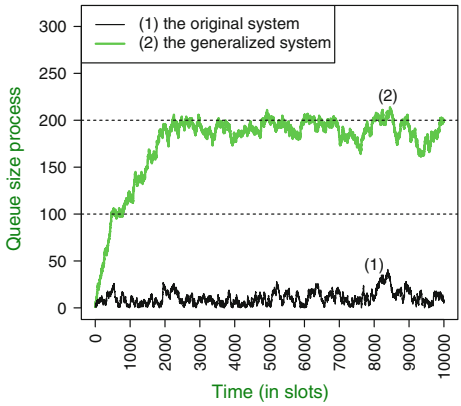


Fig. 1 Queue size for Weibull job size

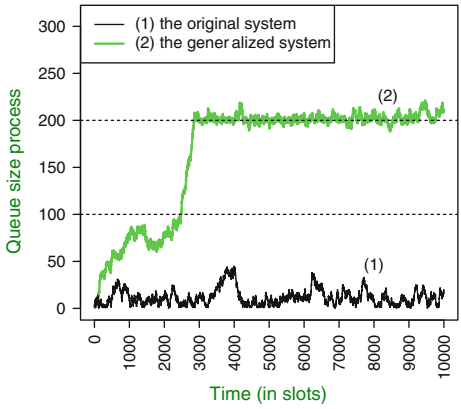


Fig. 2 Queue size for Weibull job size

Similar queue size dynamics is observed for the Pareto job size distribution with parameters $x_m = 4$, $\alpha = 5$, input rate $\lambda = 0.96$ and transmission rates (2, 3, 4), (5, 4, 6), (6, 8, 10) (Fig. 3), and (5, 4, 6), (2, 3, 4), (6, 8, 10) (Fig. 4). As a result, we obtain $\rho = 0.97$.

Thus in all cases, provided the basic process in the state-dependent system is below the threshold $x_2 = 200$, the server uses a part of capacity allowing the queue size grows extremely fast. However, if the queue size exceeds the threshold 200, the system uses full its capacity to stabilize the queue size. As a result, this control can be used to optimize the system resources utilization. At the same time, the original system uses the highest rates regardless of the queue size. It allows to stabilize the system but not by the most effective way.

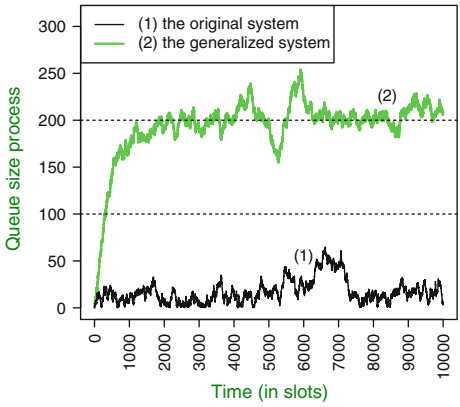


Fig. 3 Queue size for Pareto job size

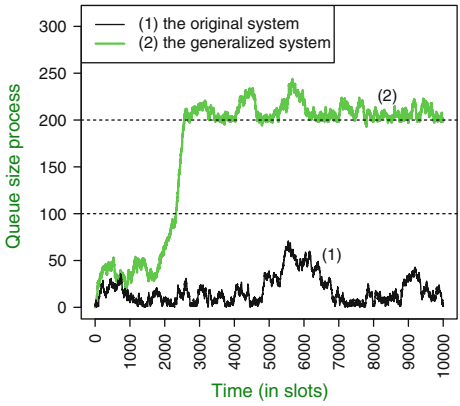


Fig. 4 Queue size for Pareto job size

5 Concluding Remarks

In this work, we consider a wide class of state-dependent systems which can be used to model the dynamics of some communication systems, including wireless systems, to optimize capacity utilization. Using regenerative approach, we formulate and prove sufficient stability conditions of the system with queue-dependent service rate (Theorem 2). The proof uses a limiting characterization of the remaining regeneration time and is based on the negative drift of the queue-size process above a threshold. The novelty of the analysis is that, unlike convenient workload-dependent models, the workload process in the considered systems is not a Markov chain. It makes stability analysis more difficult and requires a solidarity property (formulated and proved in Theorem 1). The theoretical analysis is supported by simulation of a particular case of a wireless network model with queue-dependent and Markov-modulated transmission rate. In the experiments we use both Pareto and Weibull job size distributions which are now well-recognized models for the empirical distributions arising in traffic analysis. The considered systems are also closely connected

with the systems with *asymptotically work-conserving* (AWC) discipline [16, 19]. Under the AWC discipline server uses full its capacity only when a basic process exceeds a (large) threshold x, and in many practical situations this mechanism can be realized by means of change of service rate. (In this regard we refer to the work [2] for various aspects of the systems with *workload-dependent* service rate.) Finally, as it is mentioned in the Introduction, the proposed models may be used in green computing to optimize energy consumption, however we leave a detailed analysis for a future research.

Acknowledgments This research is supported by the Program of Strategic development of Petrozavodsk State University. Research of E. Morozov is supported by Russian Foundation for Basic Research, projects 15-07-02341, 15-07-02354, 15-07-02360. Research of L. Potakhina is supported by Russian Foundation for Basic Research, projects 15-07-02341, 15-07-02354. Research of A. Rumyantsev is supported by Russian Foundation for Basic Research, projects 13-07-00008, 14-07-31007, 15-07-02341, 15-07-02354.

References

1. Alonso, M., Coll, S., Martínez, J.M., Santonja, V., López, P., Duato, J.: Power saving in regular interconnection networks. Parallel Comput. **36**(12), 696–712 (2010)
2. Bekker, R., Borst, S.C., Boxma, O.J., Kella, O.: Queues with workload-dependent arrival and service rates. Queue. Syst. **46**, 537–556 (2004)
3. Bender, P., Black, P., Grob, M., Padovani, R., Sindhushayana, N., Viterbi, A.: CDMA/HDR: a bandwidth-efficient high-speed wireless data service for nomadic users. IEEE Commun. Mag. **38**(7), 70–77 (2000)
4. Borst, S.: User-level performance of channel-aware scheduling algorithms in wireless data networks. IEEE/ACM Trans. Netw. **13**(3), 636–647 (2005)
5. Brodowski, D., Golde, N.: Linux CPUFreq: https://www.kernel.org/doc/Documentation/cpufreq/governors.txt
6. Feller, W.: An Introduction to Probability Theory and Its Applications, vol. II. Wiley, New York (1971)
7. Gandhi, A., Harchol-Balter, M., Das, R., Kephart, J.O., Lefurgy, C.: Power capping via forced idleness. In: WEED 2009, pp. 1–6. Austin, USA (2009)
8. Gandhi, A., Harchol-Balter, M., Das, R., Lefurgy, C.: Optimal power allocation in server farms. ACM SIGMETRICS Perform. Eval. Rev. **37**(1), 157–168 (2009)
9. Hewlett-Packard, Intel, Microsoft, Phoenix, Toshiba: Advanced Configuration & Power Interface: http://www.acpi.info/spec.htm
10. IBM: IBM EnergyScale for POWER7 Processor-Based Systems: http://www-03.ibm.com/systems/power/hardware/whitepapers/energyscale7.html
11. Intel: Enhanced Intel SpeedStep Technology: http://www.intel.com/cd/channel/reseller/ASMO-NA/ENG/203838.html
12. Intel: Intel TurboBoost Technology: http://www.intel.ru/content/www/ru/ru/architecture-and-technology/turbo-boost/turbo-boost-technology.html
13. Jacko, P., Morozov, E., Potakhina, L., Verloop, I.: Maximal flow-level stability of best-rate schedulers in heterogeneous wireless systems. Trans. Emerg. Telecommun. Technol. **26**(8) (2015)
14. Knopp, R., Humblet, P.: Information capacity and power control in single-cell multiuser communications. In: IEEE ICC 1995, pp. 331–335. Seattle, USA (1995)
15. Kushner, H., Whiting, P.: Convergence of proportional-fair sharing algorithms under general conditions. IEEE Trans. Wirel. Commun. **3**, 1250–1259 (2004)

16. Morozov, E.: A multiserver retrial queue: regenerative stability analysis. Queue. Syst. **56**, 157–168 (2007)
17. Morozov, E.: Stability analysis of a general state-dependent multiserver queue. J. Math. Sci. **200**(4), 462–472 (2014)
18. Morozov, E., Delgado, R.: Stability analysis of regenerative queueing systems. Automat. Remote control **70**(12), 1977–1991 (2009)
19. Morozov, E., Potakhina, L.: Asymptotically work-conserving disciplines in communication systems. In: Gaj, P., Kwiecień, A., Stera, P. (eds.) Computer Networks, CCIS, vol. 522, pp. 326–335. Springer, Switzerland (2015)
20. R Development Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna (2011): http://www.R-project.org/

Part XII
Data Management Systems

Independent Data Partitioning in Oracle Databases for LOB Structures

Lukasz Wycislik

Abstract The paper presents an implementation of data partitioning in Oracle databases dedicated for LOB (i.e. Large OBjects). These LOB structures could encapsulate any of binary data including multimedia or electronic documents in various formats. The solution, unlike most of the other authors' proposals, is narrow enough that it allows for defining an API, a data schema and an implementation that need not change regardless of usage scenarios. However, it is flexible enough that one instance of proposed subsystem allows for the provision of partitioning services for many other systems what may be useful, for example, in systems deployed in accordance with the multi-tenant architecture.

Keywords Oracle · Databases · Partitioning · LOB · Secure files · ASM

1 Introduction

Contemporary trends in the construction of information systems move towards distributed computing, where even one service can be deployed on multiple nodes, resulting in both an increase in the availability and scalability of the solution. But a system, consisting of a number of services, should see the persisted data as a logical whole. This data can grow in a 'vertical' way, for example due to the need to collect and store historical data for a given entity, which is often required by law or in a 'horizontal' manner, in the case of the collection of data from multiple entities. The last case is gaining importance recently due to the increasingly popular cloud computing, where one system can support multiple independent entities what is often called multitenancy.

The collection of such large volumes of information in databases brings a lot of challenges, among others we should mention at least queries performance, system availability and partial data archiving. One way to deal with the complexity of the problem are varied data partitioning techniques implemented by commercial database

L. Wycislik (✉)
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: lukasz.wycislik@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_59

vendors. Unfortunately, they are most often in addition to the basic version of the platform and one has to pay extra for them substantial amounts of money.

However, in some applications, a variety of options offered by commercial partitioning mechanisms would not be fully utilized, and thus its purchase would not be justified economically. A good example is the partitioning of unstructured data that in database technology is called LOB (Large OBject). The paper presents solution of partitioning such type of data in Oracle database, that require neither partitioning option nor enterprise edition one.

This proposal is ideal for applications such as storage of xml files, documents in various formats and other multimedia data.

2 Related Work

The advantages offered by the partitioning for handling large data seem obvious. No wonder that since the introduction of this concept, it has become a field of interest in scientific circles.There are a number of monographs and articles on the advanced techniques of data storage and processing in which much of the content is dedicated to technique of partitioning. Most common of them are limited to the presentation concept already implemented in commercial solutions [3]. Some other deal with improvement possibilities of the mechanisms, which have already been implemented by the supplier, in several fields of application e.g. data warehouses [1], XML databases [4, 9], RDF databases [8].

There are also papers describing partitioning implementation from scratch that for Oracle database can be found even in [2]. Unfortunately, the proposals that attempt to cover all commercial functions are complicated to maintain and sometimes prone to failures. This is undoubtedly related to the complexity of these solutions that attempt to move the partitioning logic from application to database layer and must be implemented by techniques such as stored procedures and triggers. The risk of failure increases in the case of frequent changes in the system particularly those that require changes to the database schema.

This article proposes a solution dedicated to the partitioning of LOB objects. This specialization can be widely used in applications that operate on XML files, documents in various formats or multimedia files. It is however narrow enough that it allows defining an API, a data schema and an implementation that need not change regardless of usage scenarios.

3 Commercial Partitioning Option

Oracle is one of the largest companies in the world today. It is also the name of a well-known and highly regarded database platform, which is widely used in all areas of datastore technology. First partitioning functionality appeared in version 8, published

in 1997. Developed throughout the life of the product is today a very powerful tool to deal with the processing and storage of large volumes of data. It allows to partition (and subpartition) tables, index-organized tables and indexes, where the partitioning algorithms can be based on:

- range,
- list,
- hash,
- reference,
- interval.

More information about native data migration mechanisms can be found in [10]. And more information about partitioning concepts can be found in [3] or directly in Oracle White Paper documentation [7].

But this functionality is available only in Enterprise Edition, which is $30000 more expensive (per one processor) than Standard Edition and costs $11500 extra (per one processor).

4 Proposed Implementation

The proposed implementation is dedicated for storing and processing LOB objects in a partitioned by interval way. There are several strategies concerning storing LOB objects in databases. The most generic involve storage of large objects outside a database server, directly on a file system. In some database servers, this approach seems both to facilitate the management of a database (archiving of such large data is then performed directly from the operating system) and to keeping performance at an appropriate level (due to elimination of overhead resulting from storing data in database files that often have specific format). In Oracle database this stategy could be implemented (and often was, due to known limitations of LOB types) using *directory* objects in order to direct access to file system with *UTL_FILE* or *DBMS_FILE_TRANSFER* PL/SQL packages. This approach required data binding between the world of database and the world of file system what could be implemented, eg. by storing for each lob object the directory and file name in the file system.

But this was an issue up to version 10 g. Since 11 g version Oracle introduced SecureFiles mechanism and claims that storing LOB objects inside database is not only comparatively efficient but even exceeds the generic strategy both in read and write performance [6]. SecureFiles feature offers also such advanced options as deduplication, compression and encryption.

Whats's more, using Oracle ASM technology [5] (that comes with Standard Edition completely free) we get additional elasticity and possibility of using transparent load balancing and failover mechanisms.

4.1 Domain Model

The domain model was visualized in the Fig. 1 using physical data diagram.

Each system that wants to use partitioning mechanism has to be registered (*systems* entity). The whole system may consists of several modules (*modules* entity) each of which may need to store LOB objects of several classes (*blobclasses*), e.g. multimedia records of different format or size. Within each class the partitioning mechanism may be configured independently. Every LOB record (*records* entity) is stored in one tablespace that corresponds to one partition (*partitions* entity).

In Fig. 2 an example of folders organizing concept was shown. The partitioning system require to define *DiskGroups* (*diskgroups* entity) that reflect volumes (e.g. disks, disk arrays) available for the system. The effective path of the concrete partition

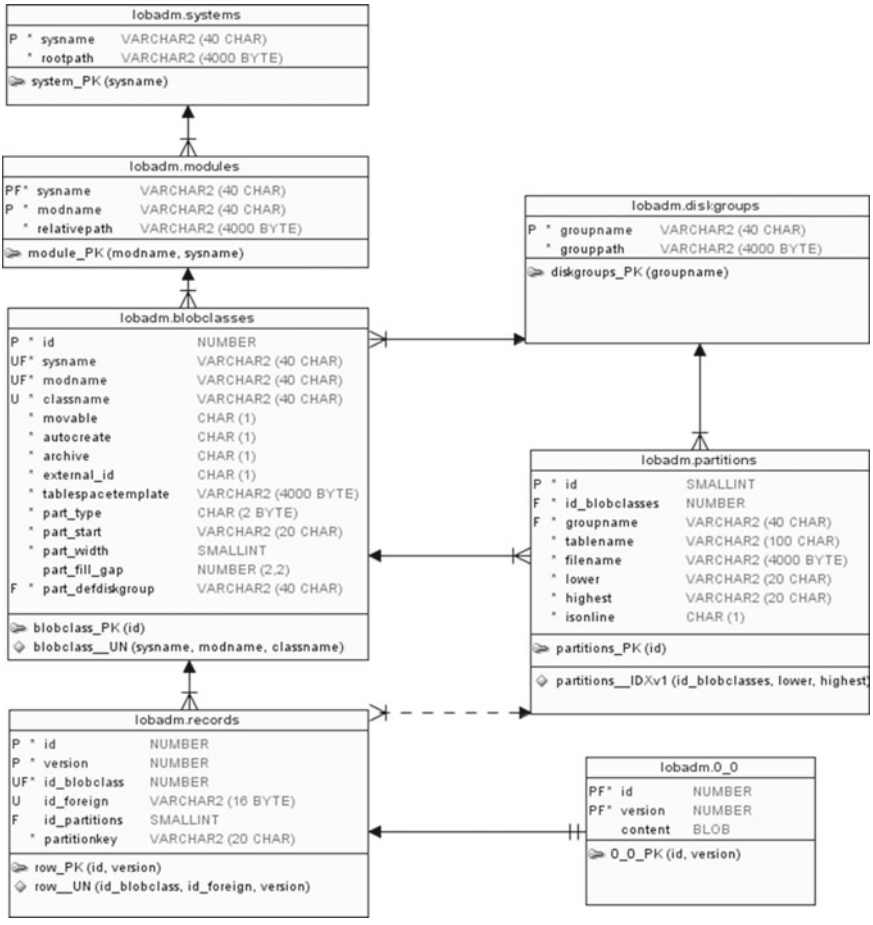


Fig. 1 Data model of the LOB partitioning system

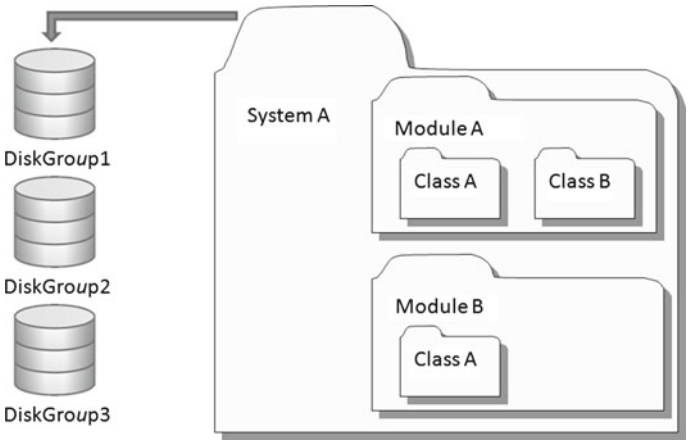


Fig. 2 An example of folders organizing concept

(located in the concrete tablespace) is defined by concatenation of diskgroup path and relative paths of system, module and class. As may be seen in data model (Fig. 1) a diskgroup is related to a class and not to a system what allows for scattering data of the same class between different diskgroups. This is usefull particularly when some partitions with older data (thus less frequently accessed) could by stored on slower but cheaper devices.

Depending on the needs, the partitioning system for a given class can be configured variously:

- record can be identified externally (when suitable host system provides records identification by it's own) or internally (when partitioning system provides records identification),
- the partitioning system can store all versions of the given record or the latest one only,
- partition can be created automatically (according to time intervals defined by user) or must be manually created before suitable record can be stored,
- records, in case of partitioning key change, can or cannot be moved between different partitions.

One of configuration aspects is the size and location of partitions to be created automatically. There are three parameters involved – *par_start* defining the 'datum point', *part_width* defining default width of partitions and *part_fill_gap* which is explained later.

But even in automatic mode it is possible to create a partition manually by specifying time boundaries explicitly (see *pLower* and *pHighest* parameters or partitioning key *pPartitionKey* in *PartitionCreateAndCommit* procedure in listing 58.1). Example of case when partition is being created automatically but formerly another one was created manually is shown in Fig. 3. The 'datum point' is fixed at 2015-01-01,

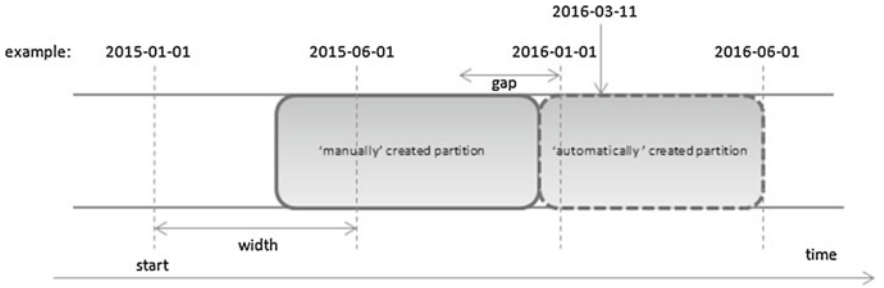


Fig. 3 Partition creating concept

the width is a half a year and the gap is one half of width. Dotted vertical lines shows default partitioning grid derived from a 'datum point' and a width. But one partition has been created manually (let us say that boundaries are 2015-04-01 and 2015-11-01). Then, where is the need of partition creation for the record with partitioning key 2016-03-11, the system try to conform to default grid, but this could lead to one month gap to already existing partition. Because the *fill_gap* parameter is set to three months the system will not allow the formation of a gap and extend partition to be created by fixing boundaries to 2015-11-01 and 2016-06-01 respectively.

4.2 Implementation

All required implementation was encapsulated into two PL/SQL packages. One for administrative activities such as defining systems, modules, classes, diskgroups etc. shown in listing 1. This functionality can also be accessed from user graphical interface for end users (administrators).

Listing 1 PL/SQL package for administration

```
function GetPartitions( pSysname in systems.sysname%type ,
    pModName in modules.modname%type ,
    pClassName in blobclasses.classname%type )
    return PartitionsTableType pipelined ;
function GetDiskGroups return DiskGroupsTableType pipelined ;
function GetBlobClasses return BlobClassesCurType pipelined ;
procedure PartitionCreateAndCommit( pResult out integer ,
    pSystem blobclasses.sysname%type ,
    pModname blobclasses.modname%type ,
    pClassname blobclasses.classname%type ,
    pPartitionKey records.partitionkey%type ,
    pLower in out partitions.lower%type ,
    pHighest in out partitions.highest%type ,
    pGroupName in partitions.groupname%type ) ;
```

```
procedure LobClassModify( pResult out integer ,
    pSystem blobclasses.sysname%type ,
    pModname blobclasses.modname%type ,
    pClassname blobclasses.classname%type ,
    pTSTemplate in blobclasses.tablespacetemplate%type ,
    pPartStart in blobclasses.part_start%type ,
    pPartWidth in blobclasses.part_width%type ,
    pPartFillGap in blobclasses.part_fill_gap%type ,
    pDefDiskGroup in blobclasses.part_defdiskgroup%type ) ;
procedure DiskGroupSet( pResult out integer ,
    pName in diskgroups.groupname%type ,
    pPath in diskgroups.grouppath%type ) ;
procedure DiskGroupUnset( pResult out integer ,
    pName in diskgroups.groupname%type ) ;
```

It is important to note that the mechanism of partitions creation must execute DDL (Data Definition Language) operations that can be only executed within PL/SQL procedure body by means of *execute immediate* command. For one partition one tablespace and table must be created. In the Fig. 1 the "0_0" table was show as an example of concrete table that implements concrete partition. It consists of three obvious columns (identifiers and LOB content) but what is important referential integrity is also defined what ensures the consistency of records metadata and partitioned content of records.

The physical properties of tablespaces for tables implementing partitions may be defined with templating mechanism (*tablespacetemplate* attribute of *blobclasses* entity). For instance for a given class of LOB the template clause could be—*size 10M autoextend on next 100M maxsize 1012M extent management local autoallocate SEGMENT SPACE MANAGEMENT AUTO logging*. This phrase will be concatenated with the whole expression that creates given tablespace. A part of source code of *PartitionCreateAndCommit* procedure that is responsible for creating tablespaces is shown in listing 58.2.

Listing 2 The templating mechanism for defining physical propertis of a tablespace

```
clause := 'create bigfile tablespace ' || iname ||
          '_S datafile ''' || ffname || ''' ' ||
          bc.tablespacetemplate ;
execute immediate clause ;
```

The second package covers services provided directly for host applications (listing 58.3). At the beginning the host system must introduce itself and declare to what module and class it want to access (*OpenLobClass*). The system then get an handler which must use in subsequent calls of other procedures.

Remaining procedures, together with their parameters are self-describing and implement CRUD (Create/Read/Update/Delete) operations on LOB objects.

Listing 3 PL/SQL package for LOB operations

```
procedure OpenLobClass( pLobClass out blobclasses.id%type ,
    pSystemName in blobclasses.sysname%type ,
    pModuleName in blobclasses.modname%type ,
    pClassName in blobclasses.classname%type ) ;
procedure Get( pResult out Integer ,
    pLobClass in blobclasses.id%type ,
    pPartitionKey out records.partitionkey%type ,
    pContent out blob ,
    pId in records.id%type ,
    pversion in out records.version%type ,
    pExtId in records.id_foreign%type default null ) ;
procedure Put( pResult out Integer ,
    pLobClass in blobclasses.id%type ,
    pPartitionKey in records.partitionkey%type ,
    pContent in blob ,
    pId out records.id%type ,
    pExtId in records.id_foreign%type default null ) ;
procedure Modify( pResult out Integer ,
    pLobClass in blobclasses.id%type ,
    pPartitionKey in records.partitionkey%type ,
    pContent in blob, pId in records.id%type ,
    pversion out records.version%type ,
    pExtId in records.id_foreign%type default null ) ;
procedure Remove( pResult out Integer ,
    pLobClass in blobclasses.id%type ,
    pId in records.id%type ,
    pversion out records.version%type ,
    pExtId in records.id_foreign%type default null ) ;
```

Transaction management is left to the application calling functionality encapsulated in the PL/SQL packages, what means that no *commit* commands have been included in procedure bodies. This decision is due to the fact that host systems may manage domain metadata by its own so it is more safe (regarding data consistency) to commit metadata transaction and LOB transaction once. The only exception is procedure *PartitionCreateAndCommit* which calls DDLs phrases that do commit implicitly. That is why the *Put* procedure in case of absence of proper partition does not create it implicitly but returns adequate error code leaving the host application an explicit call of *PartitionCreateAndCommit*.

5 Conclusions and Summary

The paper, on the example of PL/SQL implementation, presents the approach of interval partitioning implementation dedicated for LOB objects processing and storing. The specificity of interval processing is implemented in a separate PL/SQL package so extending functionality to different type of partitioning is a minimally invasive.

The concept appears to be complete and possible to implement in other popular database servers. Thanks to the fact that this solution of partitioning LOB data in Oracle database does not require neither partitionig option nor enterprise edition, it allows one to save significant amounts of money.

The proposed solution has already been successfully applied to the clinical documents system, where it stores and manages source formes of electronic documents formatted as XML files.

The system is also multitenant-ready—one instance suports partitioning services for many host systems, keeping their data separately (thanks to dedicated tablespaces) enables archiving or transfer their data to other data centers.

Further research will rely on the proper selection of the physical parameters of the tablespace taking into account the specifics of the LOB files that are stored (considering both the occupation of space and query performance). The second direction of development would be to extend the functionality to allow management of partitions availability (i.e. making partitions online/offline) and changing partitions location (i.e. partitions moving).

Acknowledgments This work was supported by NCBiR of Poland (No INNOTECH-K3/IN3/46/229379/NCBR/14).

References

1. Bellatreche, L., Boukhalfa, K.: An evolutionary approach to schema partitioning selection in a data warehouse. In: Tjoa, A.M., Trujillo, J. (eds.) Data Warehousing and Knowledge Discovery. Lecture Notes in Computer Science, vol. 3589, pp. 115–125. Springer, Berlin Heidelberg (2005)
2. Kaleta, M., Chwastowski, J., Czajkowski, K.: Niezależne partycjonowanie danych w bazach Oracle. Studia Informatica **34**(2B), 139–158 (2013)
3. Kuhn, D.: Pro Oracle Database 12C Administration, 2nd edn. Apress, Berkely (2013)
4. Luoma, O.: Efficient queries on XML data through partitioning. In: Filipe, J., Cordeiro, J. (eds.) Web Information Systems and Technologies. Lecture Notes in Business Information Processing, vol. 8, pp. 98–108. Springer, Berlin Heidelberg (2008)
5. Oracle: Oracle Automatic Storage Management. http://docs.oracle.com/cd/E11882_01/server.112/e18951.pdf
6. Oracle: Oracle Database 11g: SecureFiles. http://www.oracle.com/technetwork/database/options/compression/overview/securefiles-131281.pdf
7. Oracle: Oracle Partitioning with Oracle Database 12c. http://www.oracle.com/technetwork/database/options/partitioning/partitioning-wp-12c-1896137.pdf
8. Vasani, S., Pandey, M., Bhise, M., Padiya, T.: Faster query execution for partitioned RDF data. In: Hota, C., Srimani, P. (eds.) Distributed Computing and Internet Technology. Lecture Notes in Computer Science, vol. 7753, pp. 547–560. Springer, Berlin Heidelberg (2013)

9. Wycislik, L.: Performance issues in data extraction methods of ETL process for XML format in oracle 11g. In: Kozielski, S., Mrozek, D., Kasprowski, P., Małysiak-Mrozek, B., Kostrzewa, D. (eds.) Beyond Databases, Architectures, and Structures, Communications in Computer and Information Science, vol. 424, pp. 581–589. Springer International Publishing (2014)
10. Wyciślik, L., Augustyn, D., Mrozek, D., Pluciennik, E., Zghidi, H., Brzeski, R.: E–LT Concept in a light of new features of oracle data integrator 12c based on data migration within a hospital information system. In: Kozielski, S., Mrozek, D., Kasprowski, P., Małysiak-Mrozek, B., Kostrzewa, D. (eds.) Beyond Databases, Architectures and Structures, Communications in Computer and Information Science, vol. 521, pp. 190–199. Springer International Publishing (2015)

Hybrid Column/Row-Oriented DBMS

Małgorzata Bach and Aleksandra Werner

Abstract The rapid growth of data volumes which store the increasing amount of information makes the necessity of searching for the effective methods of data storing and processing. Some researches on this field recommend changing the row data organization that is classical for DBMS to the columnar one and/or the in-memory approach usage. The article presents chosen hybrid solutions which simultaneously enable storing data both in a row-based way and column-based one, as well as process these data in the in-memory technology.

Keywords Row store · Column store · Clustered columnstore index · OLAP · Hybrid column/row oriented systems

1 Introduction

The dynamic development currently observed in a business world closely depends on the quality and the speed of decision-making, which are the result of the multi aspect analysis of massive amount of data collected by the companies. The number of questions that must be answered by analysts continues to grow as well as the amount and complexity of data that must be processed to provide the answers to their questions. The question is whether in the changing reality the traditional database solutions are able to meet the expectations.

The significant increase in the number of solutions that use new alternative techniques and methods of working with data can be noticed at present. Among them the idea of the columnar format data storage seems to be the most promising and prospective. This idea is not actually a new one as it originates from the early 70s of the last century. In the last years the renaissance of this conception has been observed. Therefore, the analysis of hybrid solutions offering a choice between the row and the

M. Bach (✉) · A. Werner
Institute of Informatics, Silesian University of Technology, Gliwice, Poland
e-mail: malgorzata.bach@polsl.pl

A. Werner
e-mail: aleksandra.werner@polsl.pl

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3_60

columnar data storage methods was done. The most important findings that appeared as a result of the performed analysis are included in this article.

Presented study is a continuation of the previous research made in the field of searching the new and more efficient ways of implementing the analytical processing. It was described in [3–5].

2 Problem Description and Related Works

The specificity of the OLAP (**O**n-**L**ine **A**nalytical **P**rocessing) is definitely different from the OLTP (**O**n-**L**ine **T**ransaction **P**rocessing) transactional one. The purpose of the transactional systems is to support the current activities of the company, e.g. bank systems that support customer accounts, financial and accounting systems of super- and hypermarkets cooperating with the cash registers. These systems are optimized for a maximum transactional performance, concurrency and availability.

In contrast to the transactional systems, the analytical OLAP systems are designed for managers, analysts and administrators. Their main task is to enable the implementation of complex, multi aspect statements (reports), operating on large quantities of data that facilitate analysis of the company and allow to take the proper business decisions. The data used in the analytic systems is generally not modified. While in the OLTP systems search operations (SELECT), addition (INSERT), modification (UPDATE) or deleting (DELETE) occur equally often, the OLAP systems generally focus on reading data. These differences in the tasks set to the both types of a processing make it difficult to find a universal data model that could guarantee the desired performance in both cases.

For more than 40 years the relational model has been dominating the database market. Is this model suitable enough for both transactional and analytical processing?

The world of the relational databases is two-dimensional. Data is stored in the tabular structures where the rows describe the entities from a specific reality which the database concerns and the columns contain the attributes characterizing them. Such two-dimensional perception of data is used on the conceptual level. On the physical-storage level the database tables must be mapped into one dimension—row-by-row or column-by-column [11].

In the first approach the full information about an object is stored together. For example, in the first place the first *record*—embracing full information about the first customer, as his name, date of birth, address, etc., is stored. Then full information regarding the second customer is written on the disk, and so on. In contrast to this approach, the 'column-by-column' one keeps all attribute information together. For example in the first place all customers' names are stored, next—all their dates of birth, etc.

There are a lot of studies concerning the row and column data organization. The paper [6] presents basic differences between column-oriented and traditional row-oriented databases. It describes why column oriented DBMS's are better for the

needs of analytical processing than traditional ones. Many authors claim that one size does not fit to all, so it enforces different types of database systems for OLTP and OLAP [18]. The article [8] argues that column stores may be efficiently emulated as well on row stores. Because of the fact that IT systems are used more and more intensive, there is a need to develop a large scale data processing [14, 15]. These studies examine chosen database technologies—e.g. Vertica or IQ to confirm the advantages of using columnar data processing.

Existing studies show that column-organized approach offers many benefits to analytic workloads but in the case of transactional systems, where UPDATE or INSERT operations data are relatively frequent, the column organization can be less efficient. This discrepancy means that different systems for different, analytical or transactional, needs must be used.

The resolution can be the usage of a novel hybrid solution combining both technologies in one product that has appeared in recent times.

In tests described in [3] the column-oriented systems: Actian Vectorwise, Sybase IQ and Infobright were compared and the obtained results were related to the results gained for the SQL Server 2012 system using the nonclustered columnstore index. The time of execution and implementation plans for the queries with analytical extensions[1] were analysed at that time. That study showed that Vectorwise and Infobright solutions came off very well although they were new systems, in contrast to the worse Sybase IQ which is considered to be a pioneer of a commercial columnar database.

Taking into account the average result of all made tests, the SQL Server 2012 with the column index turned out to be the best[2] (Fig. 1). The study paid attention to certain limitations which entailed the usage of these types of indexes. Among them the fact that the nonclustered columnstore index was an additional structure that in a significant way could increase the size of the database, was emphasized. The research in this area was presented in [2]. Besides data from a table with a column index could only be read and it was not possible to add new values to them or update existing ones. In order to execute INSERT, UPDATE or DELETE operations the index should be firstly deleted and after the modification of data re-created. Obviously it entailed the additional time consumption. Despite the high efficiency of the search tasks these restrictions could cause described solution not to be used in certain cases. Shortly after completion the tests presented in [3] a new version of the SQL Server system, SQL Server 2014, was made available for the users, which introduced another restriction-free type of the columnstore index. It gave the impulse to continue the research in the field of systems which offer both row- and column-organized tables.

[1] Queries included operators: ROLLUP, CUBE, GROUPING SETS, aggregate functions: COUNT, AVG, SUM, ranking functions: RANK, DENSE_RANK and PARTITION BY clause.

[2] For a better visualization the differences in speed of query execution in Actian Vectorwise, Infobright, Sybase IQ and MS SQL Server 2012 database systems, the percentage scale on Y axis was used.

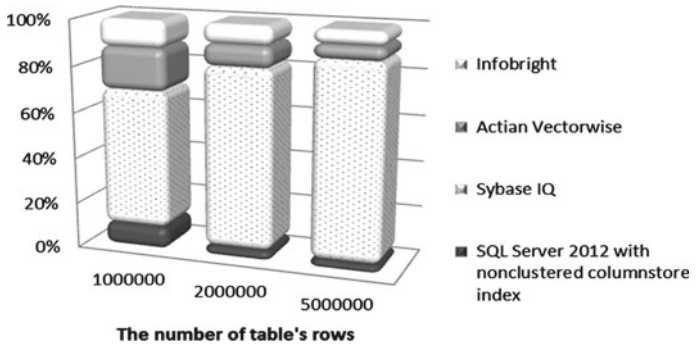


Fig. 1 The cumulative execution time of sample queries for Actian Vectorwise, Infobright, Sybase IQ and MS SQL Server 2012

3 Hybrid System Architecture

D. Abadi in [1] presents 3 different approaches to building hybrid systems, namely: *PAX*, *Fractured Mirrors* and *Fine-grained hybrids*.

The PAX idea is to store as many rows of data as can be fitted into a block, but within the block the data is stored in the columns. The benefits of compression and cache-efficiency in column stores are preserved and additionally the whole rows are brought back in a single step. *PAX* is different from a 'pure' columnar storage where each column is stored in the entirely separate disk blocks. Assuming that there is a table with 10 attributes, in a 'pure' column store data from each original tuple is spread across 10 different disk blocks, whereas in PAX all data for each tuple can be found in a single disk block. That is the key difference between both methods. It appears that Vectorwise which was tested and described in [3] use the first approach—*PAX*.

In the *Fractured Mirrors* approach all data is replicated. Each replica has the different storage layouts, row-store and column-store. If the query is scan-oriented, e.g. an aggregation or summarization query, it is sent to the column-store replica. In other situations it is sent to the row-store layout.

In the third approach, the *Fine-grained hybrids*, the individual tables can be divided into both: the row and the column-oriented stores. If some columns are often accessed together they can also be stored together (in rows). In this case the remaining columns are stored separately. This can be done within a disk block, a table or even at a slightly larger grain across the tables. Nowadays more and more vendors create hybrid systems using one or more of these approaches.

3.1 Description of Chosen Implementation of Hybrid DBMS

MS SQL Server In contrast to the SQL Server 2012 which has only nonclustered columnstore index, a new kind of columnstore index—the clustered one, is introduced in SQL Server 2014. Both indexes use the same in-memory technology, but they differ in the features they support. Data is stored in a columnar format with each column of data separated into its own segment. The segments are organized into a row group which can contain over one million rows. If the number of rows in a table exceeds this value, the new row groups are created and the column segments are broken across the row groups. SQL Server uses new query execution mechanism, called batch-mode, which is closely integrated with, and optimized around, the columnstore storage format [10, 16, 17].

One of the main benefits of a columnstore index is the high compression of data. SQL Server compresses the data at the segment level in which data is the most homogenous. It causes the achievement of higher compression rates in comparison with the traditional ones.

The nonclustered columnstore indexes cannot be updated. It contains a copy of a subset of all the columns in the table. To update the data with this kind of index, it must be dropped and rebuilt. It can result in worsening in a query performance.

A clustered columnstore index is the physical storage for the entire table and is the only index for the table. The clustered columnstore index is updateable. It permits data modifications and bulk load operations. To reduce the fragmentation of the column segments and to improve the performance for loading and other DML operations, the columnstore index can save some data temporarily into a rowstore table, called a deltastore. To return the correct query results, the clustered columnstore index combines query results from both the columnstore and the deltastore tables [2].

Clustered columnstore indexes use the deltastore in order to prevent fragmentation of column segments in the columnstore. Rows are accumulated in the deltastore until the number of rows reaches the maximum allowed for the row group. If the deltastore contains the maximum number of rows per row group, SQL Server marks the row group as CLOSED and moves them into the columnstore.

Oracle Oracle 12.1.0.2 database has a new interesting option that enables tables to be simultaneously represented in the memory using a traditional row format and a new in-memory hybrid column format. The columnar format is an additional transaction-consistent copy of the object, so it does not replace any of the already existed Oracle database structures—e.g. buffer cache [9]. Thanks to the existence of this special dual-format database architecture, the analytic queries are automatically routed to the column format and the transactional queries to the row format. Due to the fact, specific columns usually have many repeated values, Oracle Database In-Memory compresses these repeated values to save the memory. Besides it optimizes processing by executing query predicates only once for each unique column value [12].

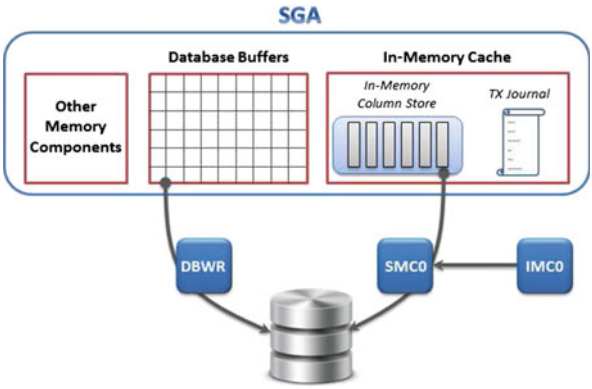


Fig. 2 The in-memory structures in the Oracle instance

After enabling the In Memory (IM) Cache,[3] the in-memory column option can be defined at one of the following levels: NONE, LOW, MEDIUM, HIGH, CRITICAL.[4] Additionally the various compression can be specified—i.e. MEMCOMPRESS FOR DML, MEMCOMPRESS FOR QUERY or MEMCOMPRESS FOR CAPACITY.[5] The particular levels allow to impose an order of the objects to be loaded into the in-memory pool [9]. By default, Oracle Database spreads the database object's data in the IM column store while the first accessing of the table (option NONE) [13]. After the In Memory Option activation, two extra memory areas are allocated in the System Global Area (SGA): In Memory Column Store (IMCS) where the data of interest are stored in the columnar structure format and TX Journal that serves to keep IMCS consistent—i.e. aligned to changes in the data [1]. Besides, new background processes, IMCO, SMCO and Wnnn, manage to load and maintain the IMCS (Fig. 2).

The In-Memory area contains two subpools: IMCU pool that stores In Memory Compression Units (IMCUs) and SMU one that stores Snapshot Metadata Units (SMUs). IMCUs contain column formatted data while SMUs contain metadata and transactional information [9].

DB2 BLU Acceleration is a dynamic in-memory columnar technology available with the DB2 10.5 release. Each column is physically stored in a separate set of data pages. Each column is compressed with its own compression dictionaries. DB2 uses a form of Huffman encoding with the traditional adaptive mechanisms. Data is compressed on the basis of the frequency of values in a column. It means that the value that appears many times should be compressed more than other values that do not appear so often. In addition to column-level compression, BLU Acceleration

[3] The INMEMORY_SIZE initialization parameter specifies the amount of memory reserved for use by the IM column store. The larger the in-memory area, the greater the number of database objects that can utilize it.

[4] CRITICAL > HIGH > MEDIUM > LOW.

[5] The data stored in the in-memory column format is automatically compressed with a set of compression techniques that improve the memory capacity, the query performance or both elements.

also uses page-level compression when appropriate. This helps further compression of data based on the local clustering of values on individual data pages.

As data is loaded into column-organized tables, BLU Acceleration tracks the minimum and maximum values on ranges of rows in metadata objects that are called the synopsis tables. When the query is run, BLU Acceleration looks up the synopsis tables for ranges of data that contain the value that matches the query. It effectively avoids the blocks of data values that do not satisfy the query. Additionally BLU Acceleration skips straight to the portions of data that matches the query [7].

4 Tests

The test environment was configured with the usage of the 2.5 GHz computer with Intel(R) Core(TM) i5-2520M CPU, 8 GB of RAM and 2.67 GHz computer with Intel(R) Xeon CPU, 16 GB of RAM. All tests were performed against the employment agency database consisting of the following tables: Person (75 columns and more than 400 000 rows), employment Periods (15 columns and 150 000 rows) and Events (8 columns, 2 800 000 rows). In order to verify the usefulness of hybrid solutions in data processing, the set of variety of queries was executed. Queries with/without tables joining, with/without aggregate functions, with GROUP BY and HAVING clauses and with analytic functions were analysed during the tests. Analytic functions sometimes called OLAP offer the ability to perform the complex data analysis within a single SQL statement. These functions, coupled with the GROUP BY extensions, CUBE or ROLLUP, provide an efficient mechanism to compute analyses—for example ranking, which returns the rank of a row relative to the other rows in a partition.

For the research purposes not only different types of queries were performed but also the number of table's rows was varied up to 20 000 000 in the biggest table concerned. In the first test scenario data access efficiency was estimated, so the several SQL queries of a different categories were executed.

The execution time of three sample queries, performed on an SQL Server database, is presented in the Fig. 3. The first query included ROLLUP operator, the second—aggregate functions: COUNT, AVG and SUM, and the third—SUM and RANK functions. In these queries up to the three tables were used. The biggest one contained nearly 17 500 000 records.

It can be easily noticed that the execution time of the queries performed for the tables using the traditional index significantly exceeds the time measured for the tables with the clustered columnstore index. The second query shows that the time was even 64 times longer for tables with a traditional index than for the tables with a clustered columnstore one.

It can also be noticed that a columnstore index copes much better with the scalability. It can be seen that the time needed to execute the query which uses tables with regular/traditional index sharply increased with the increase of tables' rows. In the case of columnstore index—the increase was relatively small (Fig. 4).

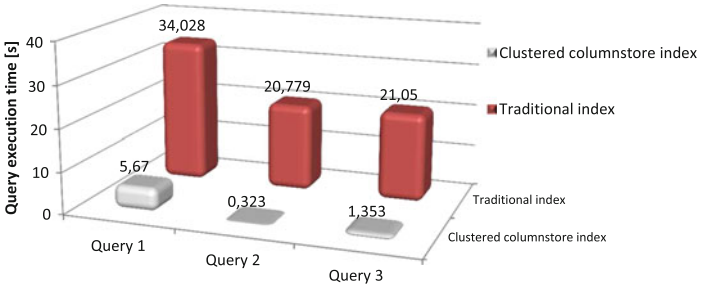
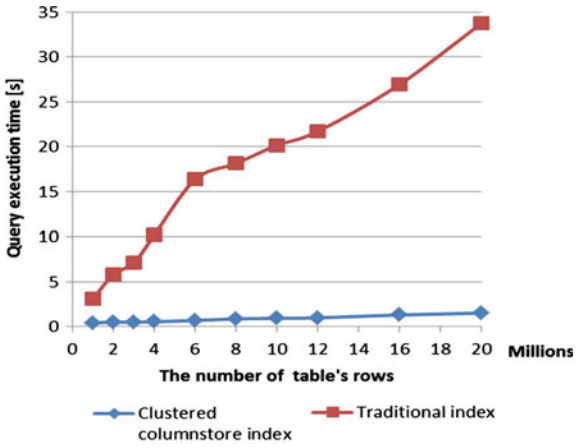


Fig. 3 The execution time of three sample queries—MS SQL Server

Fig. 4 The dependency of query execution time on the number of processed data—MS SQL Server



As it was previously mentioned a columnstore index is a part of the Microsoft's in-memory technology. The operation of a columnstore index creation takes place in the memory, but after the operation finishes, the compressed table is stored on a disk. Thus, the influence research of a memory usage seemed to be reasonable. The storage parameters such as index space and data space for tables containing from 1 up to 20 million rows were analysed in this step. It was noticed that the size of data in the table with clustered columnstore index was several but sometimes even over a dozen times smaller in comparison with the size of the table without index or with a traditional one. For example, for a table with 1 000 000 rows the clustered columnstore index required the 7,68 MB[6] data space and till 89,461 MB without index. For the table with the 16 000 000 rows it was 90,391 and 1344,125 MB respectively.

In subsequent tests the attempts were made to examine the effect of the applying data compression option on the query processing performance. These analyses were performed in Oracle 12.1.0.2 database. For this purpose the database tables were

[6] All row groups were compressed.

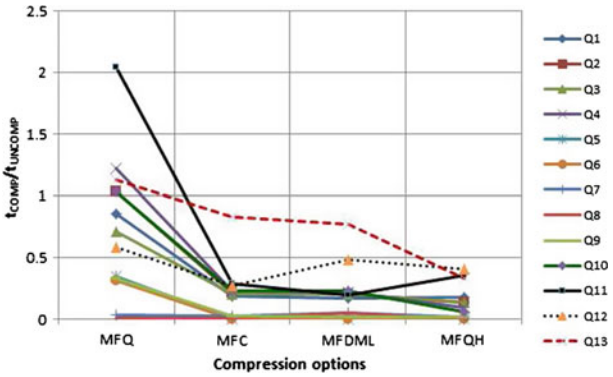


Fig. 5 Time ratio for queries performed on data organized by rows and columns—Oracle system

duplicated and in every table's copy another, different, compression option[7] was enabled. Next, after executing the set of queries, the query execution time measured for a tables with the enabled compression was divided by the query execution time measured for the tables with the NO INMEMORY option.

The result obtained illustrated how many times slower/faster the query was performed on compressed data in comparison with the query response time resulting after execution on the uncompressed data (Fig. 5). Special time ratio was calculated by dividing the execution time that was got for compressed, column-oriented, data (t_{COMP}) by the execution time that was got for uncompressed, row-oriented, data (t_{UNCOMP}).

In the case of all queries Q1–Q13 the graph lines show the same trends: the questions were always done faster for tables with INMEMORY MFC, MFDML and MFQH options than for tables with the row organization of data—i.e. with the NO INMEMORY option.

In the case of query Q11 that belonged to the group of analytical queries (ROLLUP option), the time measured for the column organization of data compressed especially for query performance (MFQ) was over twice longer than the execution time for the row organization of data. For other levels of compression, outcomes were satisfactory—i.e. the time was significantly shorter.

It is worth noting that according to the documentation the MEMCOMPRESS FOR QUERY HIGH option provides the greatest savings in both: memory and the fastest possible execution of the SELECT statements. And it was proved to be the most efficient type of the compression. The average value of the ratio presented in the described figure equals 0,14 for MFQH, while for MFDML/MFC: 0,20 and 0,75 for MFQ compression.

[7]INMEMORY MEMCOMPRESS FOR QUERY (MFQ), MEMCOMPRESS FOR CAPACITY (MFC), MEMCOMPRESS FOR DML (MFDML) and MEMCOMPRESS FOR QUERY HIGH (MFQH).

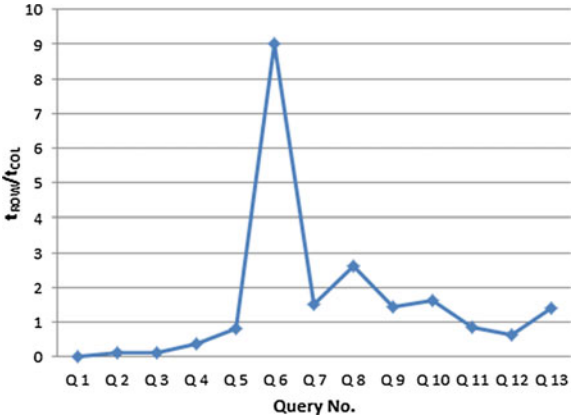


Fig. 6 The quotient of query execution time for data with the row/column storage organization—DB2 system

The outcomes achieved during the tests carried out on DB2 database are illustrated in Fig. 6. To better visualize the differences in speed of query execution in case of using row- and column-organization, the quotient of execution time for row table (t_{ROW}) and execution time for column table (t_{COL}) is presented. It can be seen that query No. 6 was executed 26 times longer for row-organized tables in comparison with the column-organized tables. In the case of other queries the differences were not so big. For queries 7–10 with aggregates functions in SELECT clause the column organization proved to be better. The similar improvement was observed in the analytical query 13 using the RANK function. For queries 1–5 that had transactional nature and returned all columns of processed tables the row storage data organization was more beneficial than columnar storage.

5 Summary

In the world of a permanent data growth, the adequate mechanisms are necessary to guarantee the required performance of data storage and processing. There are many ways to achieve this goal. One of them is the columnar data organization and in-memory technology.

The main purpose of the research presented in the paper was to analyse and to assess the possibility of usage of a new type of index offered in the MS SQL Server 2014 for storage and processing the big data volumes. Besides, other solutions enabling both row and column data organization, for improving OLAP as well OLTP, were examined. Tests carried out as described in Sect. 4, proved the usefulness of examined hybrid approaches. The columnstore index visibly improved data access time as well as the scalability of the database system. In the Figs. 3 and 4 the detailed results for chosen queries executed for tables of a given size were presented. Similarly, Figs. 5 and 6 visualize comparisons of Oracle and DB2 systems

performance. The findings indicate that columnar data storage significantly speeds up the warehouse processing. Performed study confirmed hybrid architecture is really novel and interesting approach so it should be developed and implemented by others vendors too.

References

1. Abadi, D.: A tour through hybrid column/row-oriented DBMS schemes (2014). http://dbmsmusings.blogspot.com/2009/09/tour-through-hybrid-columnrow-oriented.html
2. Abadi, D., Madden, S., Hachem, N.: Columnstores versus rowstores: how different are they really? In: ACM SIGMOD/PODS. Vancouver (2008)
3. Bach, M., Werner, A.: Analiza zasadności stosowania kolumnowej organizacji danych dla celów przetwarzania analitycznego. In: Internet in the Information Society, pp. 5–18 (2014)
4. Bach, M., Werner, A., Duszeńko, A.: Dobór struktur danych pod kątem optymalizacji przetwarzania analitycznego. Studia Informatica **33**, 145–156 (2012)
5. Bach, M., Werner, A., Duszeńko, A.: Ocena efektywności stosowania indeksów kolumnowych w bazach danych. Studia Informatica **33**, 129–144 (2012)
6. Bajaj, P., Dhindsa, S.K.: A comparative study of database systems. Int. J. Eng. Innov. Technol. **1**, 267–269 (2012)
7. Chen, W.J., Bläser, B., Bonezzi, M., Lau, P., Pacanaro, C., Schlegel, M., Zaka, A., Zietlow, A.: Architecting and deploying DB2 with BLU acceleration. In: IBM International Technical Support Organization (2014)
8. Jindal, A.: The mimicking octopus: towards a one-size-fits-all database architecture. In: VLDB Ph.D. Workshop, pp. 78–83. Singapore (2010)
9. Marx, D.: Oracle database in-memory option. In: Sangam14. Bangalore (2014). http://www.aioug.org/sangam14/images/Sangam14/Presentations/DB-In-Mem-Workshop.pdf
10. Microsoft: Columnstore Indexes Described. http://msdn.microsoft.com/en-us/library/gg492088.aspx
11. Microsoft: Using Clustered Columnstore Indexes. http://msdn.microsoft.com/en-us/library/dn589807.aspx
12. Oracle: Oracle Database In-Memory, Powering the Real-Time Enterprise. http://www.oracle.com/technetwork/database/options/database-in-memory-ds-2210927.pdf
13. Oracle: Oracle Help Center: Database Administrator's Guide. https://docs.oracle.com/database/121/ADMIN/memory.htm#ADMIN14239
14. Plattner, H.: A common database approach for OLTP and OLAP using an in-memory column database. In: ACM SIGMOD/PODS, pp. 1–2. Providence (2009)
15. Plattner, H.: Column-Oriented Databases, an Alternative for Analytical Environment (2010)
16. Potasiński, P.: SQL Server 2014–STATISTICS IO a klastrowane indeksy kolumnowe. http://blog.sqlgeek.pl/category/sql-server-2014
17. Sheldon, R.: SQL Server 2014 columnstore index: the good, the bad and the clustered (2014). http://searchsqlserver.techtarget.com/feature/SQL-Server-2014-columnstore-index-the-good-the-bad-and-the-clustered
18. Stonebraker, M., Cetintemel, U.: "One size fits all": an idea whose time has come and gone. In: ICDE, pp. 2–11. Tokyo (2005)

Author Index

A
Arcile, Johan, 3
Argyros, Antonis, 19, 353

B
Babiarz, Artur, 151
Bach, Małgorzata, 697
Bal, Artur, 329
Banasiak, Dariusz, 595
Bassara, Maciej, 269
Beierle, Christoph, 449
Bensz, Wojciech, 205
Berka, Petr, 391
Binias, Bartosz, 281
Borczyk, Wojciech, 653
Brachman, Agnieszka, 605
Budniak, Karol, 487

C
Capinski, Michal, 195
Chmiel, Wojciech, 615
Czabanski, Robert, 563
Czachórski, Tadeusz, 3
Czornik, Adam, 401

D
Dajda, Jacek, 461
DeAndrés-Galiana, Enrique J., 30
Derezińska, Anna, 93
Devillers, Raymond, 3
Didier, Jean-Yves, 3
Douvantzis, Petros, 19

F
Fernández-Martínez, Juan L., 30
Foszner, Pawel, 411
Frąckiewicz, Mariusz, 329

G
Garus, Jan, 341
Gąciarz, Tomasz, 341
Gdawiec, Krzysztof, 499
Grabska, Ewa, 551
Grochla, Krzysztof, 653, 663
Gruca, Aleksandra, 291
Grzechca, Damian, 487

H
Harezlak, Katarzyna, 291

I
Idzik, Michał, 461

J
Jaksik, Roman, 205
Jaskot, Krzysztof, 281
Jezewski, Michal, 563
Jonak, Katarzyna, 225
Jurgaś, Piotr, 401

K
Kaminski, Marek, 103
Karmeshu, 45
Kasprowski, Pawel, 291
Klaudel, Hanna, 3
Koffer, Rafal, 653
Komosinski, Maciej, 216, 269
Kopniak, Piotr, 103
Kordecki, Andrzej, 329

A. Gruca et al. (eds.), *Man–Machine Interactions 4*, Advances in Intelligent Systems and Computing 391, DOI 10.1007/978-3-319-23437-3

Kotas, Marian, 305, 315, 507
Kumar, Sanjeev, 45
Kurpas, Monika, 225
Kuusik, Rein, 421
Kyriazis, Nikolaos, 19

L
Lancucki, Rafal, 251
Landowska, Agnieszka, 115
Leski, Jacek M., 305, 315, 507, 563
Leśniak, Łukasz, 461
Lind, Grete, 421

Ł
Łabaj, Wojciech , 237
Łachwa, Andrzej, 551

M
Makris, Alexandros, 19
Małyszko, Dariusz, 517
Mensfelt, Agnieszka, 216
Michalak, Marcin, 437
Michel, Damien, 19
Moroń, Tomasz, 305, 315, 507
Morozov, Evsey, 673
Möhle, Sibylle, 449
Myszor, Dariusz, 259

N
Nakanishi, Jun, 61
Niezabitowski, Michał, 401
Nowakowski, Arkadiusz, 627

O
Oikonomidisski, Iason, 19

P
Paliouras, Konstantinos, 353
Palus, Henryk, 281, 329
Panteleris, Paschalis, 19
Pęszor, Damian, 377
Piętak, Kamil, 127, 461
Pojda, Dariusz, 525
Polanski, Andrzej, 195, 237, 251, 377, 411
Połys, Konrad, 663
Potakhina, Lyubov, 673
Przemysław Pruszowski, 377
Puszynski, Krzysztof, 225

Q
Qammaz, Ammar, 19

R
Radulescu, Andreea, 61
Rataj, Artur, 3
Redosz, Karol, 93
Roditakis, Konstantinos, 19
Rostanski, Maciej, 653
Rumyantsev, Alexander, 673

S
Sadowska, Aleksandra, 127
Schmidt, Adam, 161, 169
Sharma, Manoj, 77
Sharma, Shachi, 45
Sikora, Beata, 437
Siminski, Krzysztof, 573, 583
Simiński, Roman, 473
Skabek, Krzysztof, 525
Skurowski, Przemysław, 365
Słupik, Janusz, 377
Smiatacz, Michal, 115
Smieja, Jaroslaw, 205
Sobczyk, Jurand, 437
Socała, Jolanta, 365
Sonis, Stephen J., 30
Stańczyk, Urszula, 535
Szczęsna, Agnieszka, 377
Szkodny, Tadeusz, 179
Szwed, Piotr, 615

Ś
Ślusarczyk, Grażyna, 551

T
Tokarz, Krzysztof, 487
Topa, Paweł, 216, 269
Tyszka, Jarosław, 216, 269
Tzevanidis, Konstantinos, 19

V
Vijayakumar, Sethu, 61

W
Werner, Aleksandra, 697
Wieczorek, Wojciech, 627
Winiarczyk, Ryszard, 525

Wojcicki, Robert, 605
Wojciechowski, Konrad, 365
Wycislik, Lukasz, 687
Wysokiński , Michał, 461

Z

Zemblys, Raimondas, 139

Zielosko, Beata, 639