Khalid Saeed
Rituparna Chaki
Agostino Cortesi
Sławomir Wierzchoń (Eds.)

LNCS 8104

Computer Information Systems and Industrial Management

12th IFIP TC8 International Conference, CISIM 2013
Krakow, Poland, September 2013
Proceedings

ifip

Springer

Lecture Notes in Computer Science 8104

Khalid Saeed Rituparna Chaki
Agostino Cortesi Sławomir Wierzchoń (Eds.)

Computer Information Systems and Industrial Management

12th IFIP TC8 International Conference, CISIM 2013
Krakow, Poland, September 25-27, 2013
Proceedings

Springer

Volume Editors

Khalid Saeed
AGH University of Science and Technology
Krakow, Poland
E-mail: saeed@agh.edu.pl

Rituparna Chaki
West Bengal University of Technology
Kolkata, India
E-mail: rituchaki@gmail.com

Agostino Cortesi
Università Ca' Foscari Venezia
Venice, Italy
E-mail: cortesi@unive.it

Sławomir Wierzchoń
Polish Academy of Sciences
Warsaw, Poland
E-mail: s.wierzchon@ipipan.waw.pl

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-40924-0 e-ISBN 978-3-642-40925-7
DOI 10.1007/978-3-642-40925-7
Springer Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013947711

CR Subject Classification (1998): H.4, C.2, H.3, I.4-5, D.2, C.2.4, K.6.5

LNCS Sublibrary: SL 3 – Information Systems and Application, incl. Internet/Web and HCI

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

Printed on acid-free paper

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

Preface

CISIM 2013 was the 12th of a series of conferences dedicated to computer information systems and industrial management applications. The conference was supported by IFIP TC8 Information Systems. This year it was held during September 25–27, 2013, in Krakow.

Over 60 papers were submitted to CISIM by researchers and scientists from universities around the world. Each paper was assigned to 3 reviewers initially, and in case of conflicting decisions, another expert's review had to be sought for a number of papers. In total, about 200 reviews were collected from the reviewers for the submitted papers. Because of the strict restrictions of Springer's Lecture Notes in Computer Science series the number of accepted papers was limited. Furthermore a number of electronic discussions were held between the PC chairs to decide about papers with confusing reviews and to reach a consensus. After the discussions, the PC chairs decided to accept about 70% of the total submitted papers. The decision of selecting this percentage was indeed very hard as almost all papers were highly relevant and interesting, with good presentation and contents. We therefore followed the standard way of acceptance based on the score obtained from the referees' evaluation.

The main topics covered by the chapters in this book are biometrics, security systems, multimedia, classification, and industrial management. Besides these, the reader will find interesting papers on computer information systems as applied to wireless networks, computer graphics, and intelligent systems.

We are grateful to the three esteemed speakers for their keynote addresses. The authors of the keynote talks were Profs. Krzysztof Cios, Mieczysław Alojzy Kłopotek, and Ryszard Tadeusiewicz and Michał Woźniak. We sincerely believe that the technical papers are well complemented by these keynote lectures covering state-of-the-art research challenges and the solutions.

We would like to extend our gratitude to all the PC members for making the effort to maintain the standard of the conference. We are highly indebted to all the reviewers for their excellent high-quality reviews, which helped to retain the scientific level of the conference. We are also grateful to Andrei Voronkov whose EasyChair system eased the submission and selection process and greatly supported the compilation of the proceedings. We also thank the authors for sharing their latest achievements through the great contributions presented in the book chapters.

We hope that the reader's expectations will be met and that the participants enjoyed their stay in the beautiful historic city of Krakow.

July 2013

Khalid Saeed
Rituparna Chaki
Agostino Cortesi
Sławomir T. Wierzchoń

Organization

Program Committee

Waleed Abdulla	The University of Auckland, New Zealand
Raid Al-Tahir	The Univ. of the West Indies, Trinidad and Tobago
Adrian Atanasiu	Bucharest University, Romania
Aditya Bagchi	Indian Statistical Institute, India
Sukriti Bhattacharya	Ca' Foscari University of Venice, Italy
Rahma Boucetta	National Engineering School of Gabes, Tunisia
Silvana Castano	University of Milan, Italy
Nabendu Chaki	Calcutta University, India
Rituparna Chaki	West Bengal University of Technology, India
Young Im Cho	University of Suwon, South Korea
Ryszard Choraś	University of Technology and Life Sciences, Poland
Sankhayan Choudhury	University of Calcutta, India
Agostino Cortesi	Ca' Foscari University of Venice, Italy
Dipankar Dasgupta	University of Memphis, USA
Pierpaolo Degano	University of Pisa, Italy
David Feng	University of Sydney, Australia
Pietro Ferrara	ETH Zürich, Switzerland
Riccardo Focardi	Ca' Foscari University of Venice, Italy
Aditya K. Ghose	University of Wollongong, Australia
Raju Halder	Ca' Foscari University of Venice, Italy
Kaoru Hirota	Tokyo Institute of Technology, Japan
Władysław Homenda	Technical University of Warsaw, Poland
Sushil Jajodia	George Mason University, USA
Khalide Jbilou	Université du Littoral Côte d'Opale, France
Dong Hwa Kim	Hanbat National University, South Korea
Ryszard Kozera	The University of Western Australia, Australia
Flaminia Luccio	Ca' Foscari University of Venice, Italy
Romuald Mosdorf	Technical University of Białystok, Poland
Debajyoti Mukhopadhyay	Maharashtra Institute of Technology, India
Yuko Murayama	Iwate University, Japan
Nishiuchi Nobuyuki	Tokyo Metropolitan University, Japan
Andrzej Pacut	Technical University of Warsaw, Poland
Isabelle Perseil	Télécom Paris Tech, France
Marco Pistoia	IBM Watson Research Center, USA

Igor Podolak	Jagiellonian University, Krakow, Poland
Piotr Porwik	University of Silesia, Poland
Khalid Saeed	AGH University of Science and Technology, Krakow, Poland
Anirban Sarkar	National Institute of Technology, Durgapore, India
Kwee-Bo Sim	Chung-Ang University, South Korea
Władysław Skarbek	Warsaw University of Technology, Poland
Vaclav Snasel	VSB-Technical Univ. of Ostrava, Czech Republic
Bernhard Steffen	Technische Universität Dortmund, Germany
Giancarlo Succi	Free University of Bozen, Italy
Jacek Tabor	Jagiellonian University, Krakow, Poland
Ryszard Tadeusiewicz	AGH University of Science and Technology, Krakow, Poland
Andrea Torsello	Ca' Foscari University of Venice, Italy
Nitin Upadhyay	BITS Pilani, India
Heinrich Voss	Technische Universität Hamburg, Germany
Slawomir T. Wierzchon	Polish Academy of Sciences, Warsaw, Poland
Krzysztof Ślot	Lodz University of Technology, Poland

Additional Reviewers

Adamski, Marcin
Ferrara, Pietro
Kubanek, Mariusz
Misztal, Krzysztof
Pejaś, Jerzy
Rybnik, Mariusz
Shaikh, Soharab Hossain
Szczepański, Adam
Tabędzki, Marek

Abstracts of Keynotes

Building Data Models with Rule Learners: Classical, Multiple-Instance, and One-Class Learning Algorithms

Krzysztof Cios

Computer Science Department, Virginia Commonwealth University,
1111 W Broad St,
VA 23220 Richmond, U.S.A.
kcios@vcu.edu

Abstract. First, we shall talk about supervised inductive machine learning algorithms that generate rules and explain why rule learners are a preferred choice for model building in domains where understanding of a model is important, such as in medicine. Then we will introduce a classical rule learner that is scalable to big data. Note that classical rule learners require knowledge about class memberships of all instances. Next, we will introduce challenging multiple-instance learning (MIL) and one-class learning problems. The MIL is concerned with classifying bags of instances instead of single instances. A bag is labeled as positive if at least one of its instances is positive, and as negative if all of its instances are negative. In a one-class scenario only a single (target) class of instances is available; this type of learning is also known as an outlier, or novelty, detection problem. Since most inductive machine learning algorithms require discretization as a pre-processing step we will briefly describe an information-theoretic algorithm that uses class information to automatically generate a number of intervals for a given attribute. Second, we shall present MIL and one-class algorithms and introduce a general framework for converting classical algorithms into such algorithms.

What Is the Value of Information – Search Engine's Point of View

Mieczysław A. Kłopotek

Institute of Computer Science of the Polish Academy of Sciences
ul. Jana Kazimierza 5, 01-248 Warszawa Poland

Abstract. Within the domain of Information Retrieval, and in particular in the area of Web Search Engines, it has become obvious long time ago that there is a deep discrepancy between how the information is understood within computer science and by the man-in-the-street.

We want to make an overview of ways how the apparent gap can be closed using tools that are technologically available nowadays.

The key to a success probably lies in approximating (by means of artificial intelligence) the way people judge the value of information.

Man-Machine Interactions Improvement by Means of Automatic Human Personality Identification

Ryszard Tadeusiewicz and Adrian Horzyk

AGH University of Science and Technology
Al. A. Mickiewicza 30
Cracow 30-059, Poland
rtad@agh.edu.pl, horzyk@agh.edu.pl

Abstract. During the man-machine interactions planning and forming we must frequently concentrate on the semantic aspects of communication. For example, striving to more acceptable (for users) forms of communication with numerous computer applications we put big effort in the increasing of machine intelligence, developing more advances methods of automatic reasoning and enriching quantity and quality of knowledge built-in into computer resources. Meanwhile, emotions play an equally essential role as rational reasoning in the judge intelligence of a partner. Therefore, a computer could be accepted as intelligent (or even liked) partner in cooperation with the man if it considers human needs, especially the emotional ones. Such needs must be first recognized. Such recognition must be performed during the natural interactions between man and machine because nobody likes to be tested or examined before they can start merit communication with the selected computer application. Moreover, nobody can honestly and objectively classify their own personality. Hence, in this aspect, we cannot obtain necessary information asking a person about his or her features of personality. The keynote will present a new method for automatic human needs recognition. The personality and needs of the partner can be recognized watching the following behaviors:

- verbal, the way of talking, using vocabulary, phrases, inflection, sentence constructions, ...
- non-verbal (body language), facial and body expressions, the way of movement, dressing-up, driving cars, bicycles, ... concerning environment, family, etc.

During the typical man-machine communication we can perform automatic passive classification of man personality by means of psycholinguistic analysis. The details of how this personality can be discovered during natural language man-machine communication will be presented during the lecture.

Application of Combined Classifiers to Data Stream Classification

Michał Woźniak

Department of Systems and Computer Networks
Wroclaw University of Technology
Wyb. Wyspianskiego 27, 50-370 Wroclaw, Poland.
michal.wozniak@pwr.wroc.pl

Abstract. The progress of computer science caused that many institutions collected huge amount of data, which analysis is impossible by human beings. Nowadays simple methods of data analysis are not sufficient for efficient management of an average enterprize, since for smart decisions the knowledge hidden in data is highly required, as which multiple classifier systems are recently the focus of intense research. Unfortunately the great disadvantage of traditional classification methods is that they "assume" that statistical properties of the discovered concept (which model is predicted) are being unchanged. In real situation we could observe so-called concept drift, which could be caused by changes in the probabilities of classes or/and conditional probability distributions of classes. The potential for considering new training data is an important feature of machine learning methods used in security applications or marketing departments. Unfortunately, the occurrence of this phenomena dramatically decreases classification accuracy.

Table of Contents

Full Keynote Papers

Invited Paper

Biometrics and Biometrics Applications

Pattern Recognition and Image Processing

Various Aspects of Computer Security

Networking

Algorithms

Industrial Applications

What is the Value of Information - Search Engine's Point of View

Mieczysław A. Kłopotek

Institute of Computer Science of the Polish Academy of Sciences
ul. Jana Kazimierza 5, 01-248 Warszawa Poland

Abstract. Within the domain of Information Retrieval, and in particular in the area of Web Search Engines, it has become obvious long time ago that there is a deep discrepancy between how the information is understood within computer science and by the man-in-the-street.

We want to make an overview of ways how the apparent gap can be closed using tools that are technologically available nowadays.

The key to a success probably lies in approximating (by means of artificial intelligence) the way people judge the value of information.

Keywords: information, semantic search.

1 Introduction

Information Science, especially Information Retrieval, and in particular Search Engines needs to measure information value for at least a couple of reasons:

- document ranking in search engines - for typical queries, consisting of a few words, hundreds or even thousands matching documents are found, so to satisfy the user, one may use additional criteria for ranking, like the value of contained information
- filtering information according to one's interests (such as fresh information, information without advertisement, formulated in a simple language, etc.),
- classification of information,
- selection of a representative of a group of documents,
- selection of information to store in a database search engine - necessary because of the enormity of the Internet as compared to the capacity of even the largest search engine databases

From user's point of view, the value of information should be measured at least along the following dimensions:

- *quality* - agreement with the state of the real world (positive for true, negative for false information)
- *utility* - usefulness for the purposes of the user (positive if useful, negative, when harmful)

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 1–12, 2013.

- *actuality* - about the current state of the world or from the past
- *intelligibility* - whether or not the receiver can capture the content of the message
- *accessibility* - whether or not the message can reach the user for whatever reasons, (e.g. whether or not it has been sent)

Now the big question is if we are capable to capture the value of information along these dimensions. The answer is: at least partially yes. The subsequent sections will be devoted to show how with current technology these measurements may be performed.

Befolre proceeding let us however point at difficulties when matching these user expectations. Measuring information has a long tradition in Computer Science. As early as in 1948 Shannon [20] laid foundations for development of information theory quantifying the information as the entropy of the source of signal which defined the minimal bandwidth through which a given information may pass. Shannon's entropy measure was also for years considered as an upper limit for compression that only the best compression algorithms could approach. In the recent time however a new perspective is opening. In the domain of lossless compression a new branch of visually lossless compression emerged [17], bringing dozens of times higher compression rate than predicted by Shannon's information theory. With the advent of HTML and the separation of text and images also degrees of information compression far beyond original Shannon's concepts were achieved. Recently, the concept of excess entropy [7] has been coined to point at the fact that going beyond the pure statistical evaluation of the source of information (sender) by taking into account the structure (syntax), semantics, pragmatics and apobetics[1] we can achieve higher information levels than at the statistical level.

These sample developments point at two important points missed by the original proposal of Shannon: the significance of information receiver (like in the visually lossless compression) and the structure of information (like in HTML example). Hence the definition of information itself has been broadened beyond Shannon's concept and is understood as the content of a message sent from an information source (sender) to an information sink (receiver) increasing the state of knowledge of the latter. This imples that any information stems from an intelligent sender. And this is exactly the reason why the information value can be measured along the aforementioned dimensions in spote iof the fact that our computer technology is far from a through understanding of the real world - to measure information stemming from an intelligent sender we can apply artificial intelligence tools and methods as we will subsequently see.

[1] The *pragmatic level* of information means the activity of the receiver intended/achieved by the sender via the message. The issue of *apobetics* has been probably first raised by Gitt [10]. and it means the goal of the sender pursued when sending the message.

2 Measuring Information Quality

As mentioned, the quality of the information is the degree of compliance with the state of the real world, which a Web page describes (can be positive when the line and negative if it is inconsistent).

Can we really say how the information is consistent with the state of the real world? Not at this stage of development, but we can now at least point to situations where non-compliance is suspected :

- other users were not interested in reading it
- creator is not understanding the topic
- creator does not care about the form of the page,
- author prepared a propagandist text
- the text is in fact an advertisement

Whether or not the other users were interested in reading it, we can tell by measuring the time spent on the site and extent of returning to this page.

To detect, whether the developer worked carelessly, without much attention to content, we can measure linguistic correctness (care, the spelling, the analysis of syntax and elements of semantic analysis), as well as calculating the so-called badrank [16] to see of the author cares about the relevance of links. It may be useful to distinguish the spelling mistakes of the so-called "Typos" (e.g., by measuring the so-called edit distance properly loaded on the keyboard). Spelling errors may indicate a low level of education, and thus knowledge, and "typos" - little attention to detail paid by the creator (perhaps not read their texts). The measure should take into account not only the absolute number of errors, but the length of the text.

Web statistics can allow to distinguish among these pages with probably the highest value in terms of care and representativeness. Word documents can be assessed using methods based on tfidf (term frequency, inverse document frequency). Term frequency is the number of occurrences of words in the document, while the inverse document frequency is the inverse of the number of documents in which the term occurs (or respective logs of them). The value of the text would represent the sum of the individual words tfidf in document. Instead tfidf also the variance in the number of instances of documents and other measures of diversity of the words between documents can be used.

The above-mentioned concept of PageRank can be used to assess the value of web pages in various ways. The quality assessment is of course the basic PageRank itself. But also its variant called badrank. Badrank [16] is a measure of "spamming" by the creator as part of the so-called link farm. It is assumed that the WWWW pages pointing at the spamming page are also involved in the spamming action. Therefore, the mechanism is constructed similar to PageRank [19] except that we reverse the directions of the arrows in the Web graph. The so-called personalized random walk is assumed that is the random walker makes his out-of-boring jump to the group U of web pages which are manually qualified as spamming pages. The probability of hitting on the page is a measure of the spam. Beside spam farms [24] there are also many other techniques of spamming and corresponding methods of spam detection, based on reverting the spamming technologies. Let us mention some, described e.g. in [18]

- inserting multiple keywords to the content - antispamming by a comparison of histogram of occurrence of words in queries and other categories of words (e.g. participation in the common words [stop-words]) in general and the distribution of popular etc. words in the document
- inserting multiple keywords to the title - antispamming as above
- inserting words that are keywords pasted together (e.g. freedownload) - antispamming by observation over the average length of words
- inserting a large part of the text as anchor text - antispamming by proper statistics
- a side effect of effort put into spamming is usually neglecting the Web page appearance - antispamming by checking for a small number of HTML tags compared to the length of text
- duplicating content - antispamming by checking for high compressibility gzip - more than four times the typical compression rate
- spam generated by a random mechanism - antispamming by computing the conditional independence of the occurrence of words in n-grams on a page

Besides spam there are also other types of harmful content on the Web, like pornography [1] (which is relatively easy to recognize), illegal trade (fake brand products), criminal, violent racist and other harassment pages, as well as advertisement pages for harmful products aiming at distorting the people and the society. In this case one relies also on a set of predefined vocabulary, focused crawling, group of editors, creating directories of harmful and / or safe pages, with support of methods of event extraction and analysis. One can assess goodness of other pages by personalized PageRank in normal Web graph and badness by personalized PageRank in the reverse graph, and then compare goodness and badness of each page.

Next one can look into such as inserted by the creator of the links are actually in line with the theme of the document. You can highlight the following techniques:

- clustering (cluster analysis), links within the cluster is considered to be topical
- Links from / to spam - to be non-topical
- matching topic subject links to the document - similarity of the text near the anchor to the vector of 20 words with the highest tfidf in the target document [5]
- as previously, but with separate treatment of links within the Web domain and going outside, classifying ranks relative to these groups (e.g., above the median)
- topical triangles: if A is related to B, A points B, and C points to A and to B, then the links from C to A and B may be considered topical.
- link between the pages A and B is topical if both pages are often found high ranking in many queries

- links of type hubs / Authorities in HITS and SALSA technology
- comparative analysis of the structure of the web pages according to the recommendations of various guidelines of web page construction (e.g. elimination of navigational links by identifying their permanent place)

In most of techniques for detecting spam or other harmful content some numerical values are computed. These values can be the ratios of frequency of occurrences as compared with the "gold standard" or results of some statistical tests of significance aiming to reject the hypothesis that the website does not represent the spam.

Beside looking for signs of page misconduct, also efforts were made to define positive tests of page quality. So in [9] it is proposed to consider a page a good one if there are no errors of any kind, if the creator of the web page can be clearly identified (with contact details), verifying that the site belongs to a serious organization having experts in the field presented by the Web page, and the inclusion of references (links) to relevant pages on the topic. Here, of course, it is necessary to use a shallow analysis of natural language, identifying named entities and the use of appropriate semantic resources (lists of individuals or organizations or types of organizations that are trustworthy). Also one needs methods for appropriate classification of the content of the page [3] to match it against the list of experts. [8] proposes a number of methods for assessing the quality of Web pages edited by communities, in which the method of time series analysis of changes and of the list of readers / writers is exploited. Unfortunately, not all suggestions of [8] are plausible, just to mention the requirement "Neutral point of view". Though today technology is close to properly assessing the attitude of the author (via called a sensual analysis) and "neutrality" can in principle be measured automatically, this measure has nothing to do with the quality measurement as our primary goal is to ensure that the contents are true (reflect actual state of the world) and hence it cannot be not neutral with respect to the truth.

3 Measuring Utility

Utility is the degree to how much information brings the viewer to learn about the world to the extent necessary for the running existence / immediate action (can be positive if the beneficial and negative when it is harmful).

The difference to the quality lies in the fact that we now care for receiver's needs. So high quality information is not enough, it may prove useless for the current goals, therefore reading it may be waste of time and hence harmful.

Hence, the role of search engines is to provide the users with what they currently need. Thus, the utility can be seen in the context of the user's query.

Utility may be again judged by viewing reaction of other receivers. For example, if page advertised product and the customer bought it, it means that the information provided was useful.

Previously there were efforts to profile the user claiming that the content returned by the search engine should best fit user's profile. Nowadays such an approach is deemed to be a failure because we cannot predict from the past user's information need in the future (the user can look now for something else).

Therefore a different pathway is steered at. Rather then trying to find user's profile, a search engine reply with diverse topics is aimed at. One presents best fitting the query but differing from the preceding ones on the list, not only by content but also by topic (in fact similarity to a query is weighted against the difference to the predecessors). So the value of a page is not only judged by its similarity to the query but also its new content compared to what is provided by better pages.

In general, measuring utility requires climbing up to at least semantic level of the message. Our group is engaged in developing a semantic search engine covering the whole Polish Internet. To ensure appropriate quality of responses, we seek a well-founded method of matching user's query to the document contents. Let us briefly explain the notion of semantic search engine and its impact on ranking method requirements.

As already stressed, semantics of information expresses the meaning of this information. In linguistics research on the semantics tries among others to relate symbols (like words, phrases, characters) to (real) beings which they mean (so-called denotations) therefore related areas like morphological and syntactic analysis is engaged. Understanding of semantics may prove useful in comprehending pragmatics (the expected acting of the recipient upon obtaining the information) and apobetics (the goal of the sender when sending the information).

Identification of the meaning of an information has been subject of intense research. So-called "semantic search" is deemed to be a method of improvement of search engine response by means of understanding of user intent as well as of the search terms in the context of the document space. If we take into account advances in natural language research, we easily guess that there is virtually no chance to realize the goal of semantic search, formulated in this way, in near future. Computers have no chance to understand semantics of textual messages as they have no "experience" with the reality surrounding us humans. Access to semantics of real world appears to be a remote goal.

Therefore in our research project NEKST we reformulated in a significant way the task of semantic analysis of Internet documents by understanding the task in an operational way. Instead of trying to pretend that the machine understands the meaning of the text, we use the fact that both the information sender and the recipient are human beings. Hence not the search engine but the man has to understand the text, and the search engine only supports him in this understanding. This support has the form of so-called semantic transformations on the text which on the one hand enrich the text with new features extending search characteristics and on the other hand may move the text to other space than the document space that is into the space of objects the documents are about.

So the semantic transformation means such a transformation of the document and/or query content that allows for traditional document search via a semantically related query [4,14].

Within the system NEKST the following types of semantic transformations have been implemented:

- user suggestions [22],
- substitution with synonyms, hypernyms, hyponyms and other related concepts,
- concept disambiguation [3],
- document categorization [3],
- personalized PageRank [15],
- cluster analysis and assignment of cluster keywords to documents [2],
- explicit separation of document cluster and document search,
- extraction of named entities and relations between them [23],
- diversification of responses to queries,
- dynamic summarizing [13], and
- identification and classification of harmful contents.

Of course English language there exists a significant body of research on the above-mentioned topic, but with our system we demonstrated that the concepts are also implementable for non-English and in particular for highly inflectional languages with flexible grammar like Polish.

If you take the semantic transformation view then it is obvious that you need all the traditional mechanisms of a search engine also under semantic search, including the ranking mechanisms.

Note that traditional PageRank itself is a career of a (limited) amount of semantic information. One usually assumes that a link is added to a page with some semantic relation to the pointed page in mind. But various variants of PageRank are considered in conjunction with semantic search because the links are not perfect semantic relation indicators. So one variant, implemented also by us, is the TrustRank [11] where you may give more weight to one outgoing link and less to another (e.g. based on textual similarity between the pointing and the pointed page). Still another approach is to create a kind of "topic sensitive" PageRank [12]. You may split the collection into a set of rough categories and for each of these categories one can compute a separate PageRank. If one recognizes that a query belongs to a particular topic, the personalized PageRank may be used. It may, however, happen that one query is related to two or more categories. An efficient way of computing personalized PageRank for a mixture of categories is proposed in [15].

4 Actuality Measurement

As mentioned the actuality tells us whether the Web page contains up-to-date information or was accurate in the past only.

Though we cannot confront the web content with the state of the real world, we can nonetheless check whether or not the creator of the Web page cares for its content.

Actuality can be measured by tracking the activity of web page authors. One can use the measures of deadrank or refresh rate, etc.

In the old days, to verify the actuality of a web page, one could query the date of the last update, but today many servers update artificially the creation date or the pages are dynamically generated. However, local servers are not able to overcome the dynamic nature of the Internet. Due to the frequent deletion of web pages, one can discover not maintained pages by looking at their dead links (the author did not notice disappearance of pages).

This fact gave rise to the DeadRank. Deadrank is based on the PageRank on the reverse Web graph, though it is slightly different from badrank. One thinks about the DeadRank as the probability of visiting the page by a random walker starting his journey on non-existent web pages and jumping back to them upon being bored. Not maintained pages will have high Deadrank. Of course, the problem here is to obtain information about the pages that do not exist. Usually, we can learn about the non-existence from a suitable server error, but the server can also cheat, redirecting to another page, or creating one on the fly. One can compensate for this effect by investigating what answer is given by the server if you specify an address that you are sure does not exist. Comparison of responses will give us an indication of whether the page exists or it is an artifact of the server.

Refresh rates of Web pages are also subject of manipulation by servers nowadays, but there were technologies developed for search engines guiding the visit frequency of spiders depending on some measures of refresh rate. The article [6] proposed the so-called temporal freshness, consisting of a mixture of (1) content freshness (PF), expressing updates of the page as such and (2) in-link freshness (INF), reflecting updating of the links pointing to the page. At the time i, these quantities are:

$$PFt_i(p) = \sum_{j<i} e^{(j-1)*a*\Delta t} \sum_k w_k * C(j,k,p)$$

$$InFt_i(p) = \sum_{j<i} e^{(j-1)*a*\Delta t} \sum_l w_l * C(j,l,p)$$

where w_k and w_l are weights of activities of type k on the Web page and activities of type l on in-links of the Web page resp. $C(j,k,p)$ and $C(j,k,p)$ tell if the respective activity appeared or not on the web page at time point j. Based on these data the quantity TFC (temporal correlation freshness) is defined, which is the ratio of a temporal correlation between the measurements of PF and INF. Intuition for this measurement is as follows: the site is "fresh" if its changes are seen by the links. These measurements assume that we have a historical copies of pages available (for example, InternetArchive).

5 Measurement of Intelligibility

The concept of intelligibility specifies whether or not the information can be decoded by the receiver / mirrors what the sender wanted to send.

Intelligibility can be evaluated, e.g. by examining whether you can build profile of customers visiting the given page from the click stream and if the prevailed clicking behavior on the page corresponds to the obtained profile.

Other methods rely on an assessment of "the literary" quality. One examines the length of the words used on the page in terms of the number of letters and syllables, sentence length counted in words, syllables and/or letters. One also examines the variation of the words (the ratio of the number of words used to words in general, the cumulative distribution function of the words used etc.). Out of these statistics some complexity measures are derived.

One of the best known methods in this area is called Flesch Reading Ease index

$$FRi = 206.876 - 1.015ASL - 84.6ASW$$

where ASL is the average number of syllables per word, and ASW - the average number of words in a sentence[2].

More sophisticated methods can utilize the available semantic resources, including the classification of the conceptual difficulties of individual words. Usually lists of difficult words are difficult to obtain, but lists of easy words are easier to get.

Although measures related directly or indirectly to the length of the sentences are fairly good indicator of clarity, in case of hypertext documents on the internet we have to handle the problems of punctuation - it is neglected and thus the easiness may be underestimated.

But it turns out that a significant part of the easy pages point generally to the easy (easy to understand) ones, while the difficult pages point to difficult ones. Therefore, for example, one can use so-called TrustRank method [11] (an analogue of PageRank in social networks) to assess the page easiness based on prior assessment by humans of a set of seed pages.

6 Measuring Availability

Availability means whether or not the information was sent by the sender.

The availability of information on the Internet can be tested by determining whether the page is a search result for search for meaningful question asked. But for the modern Internet technology the detection of availability can be much more sophisticated. First, we can ask what is the probability that a random wanderer will go to the respective information (website / web page). This concept is close to the so-called PageRank. Second, we consider the wanderer applying

[2] See `http://www.editcentral.com/gwt1/EditCentral.html` for alternative measures like Automated readability index, Flesch-Kincaid grade level, Coleman-Liau index, Gunning fog, SMOG index.

the backspace key, which leads to a measure of RBS. Thirdly, the wanderer can look for certain keywords, characterizing the respective information then we will have to deal with so-called Query-Dependent PageRank. Finally, instead of asking about the probability we can think about the ranking for the mentioned measurement in general or for a given query, consisting of one, two or three words (the minimum distance from the top ranking after these words / phrases).

We also have to take into account the technical possibilities of the presentation and the ability of human perception. From a technical point of view, it is essential to tell the format of information (standard HTML, PDF, or one of over 300 different formats of text documents, which are used to store information) and match it against the palette of formats supported by user's web browser. The issue is further extended by the problems of active sites, using JavaScript, Basic or applets and information presented in the form of files. Here we can calculate the probability of format compatibility between the source and the mouth of information.

The so-called Web Accessibility Guidelines[3], developed by W3C, indicate further need for considering the latest standards of presentation (e.g. CSS in conjunction with HTML for ornamental purposes), the usage of explanatory text, the usage of non-textual elements (graphics, sound, Flash), as well as alternatives to or features of HTML, which may be not supported by one's browser (frames, iframe, embed, etc.). Based on the use / non-utilization of such features, software has been developed such as Valet[4], which counts the number of violations of these formalist readings in the form of a linear model that presents a measure of the availability of information (see [21]).

Moreover, nowadays there is a tendency to restrict access to the Internet. Under the new legislative proposals such as the U.S. availability of information on the Internet regulation[5] is the future depend on the wealth of internet users. ISPs can freely censor information available. Therefore, the measurement will be available not just answer "yes / no", but the function of the funds available to the surfer, as well as the selection of service providers by issuing and seeking the document.

7 Concluding Remarks

Over six decades of rapid development of computer science have passed since the seminal paper of Shannon, who dared to ask how much information we can pass through an information channel. By asking this, he implied that information is an objective reality that needs to be quantified somehow. It has to be regretted that over the years one has forgotten that information is not a product of computer science and cannot be arbitrarily defined. With the advent of Web technologies it became again an acute problem get an in-depth understanding of nature and properties of information (as an immaterial dimension of the real world)

[3] http://www.w3.org/WAI/WCAG20/quickref/

[4] http://valet.webthing.com/access/url.html

[5] http://www.fcc.gov/openinternet

and how they should be measured in order to provide search engine users with most valuable output. To have a valuable output we must establish the value of each piece of information. This is somehow contrary to the current philosophical trends that hamper the actual reflection on the nature of information.

We have tried to demonstrate in this presentation why it is necessary to go beyond Shannon model, why it is important to go beyond statistical level to appropriately capture the value of the information. We have pointed to the five dimensions of information evaluation: quality, utility, actuality, intelligibility and availability. Last not least we have demonstrated that with current tools or ones available in near future it is possible to measure the information value according to these dimensions.

Surely, we do not have a proper understanding of all aspects of information and the technical realization of some concepts is far from satisfactory. But we can nonetheless hope that in near future the search engines will be able to tell us far more about the web pages we are looking for just accelerating our search for helpful information. The search engine designers have to keep in mind that the people do not live to search but rather they search to live.

References

1. Horng, W.B., Lee, C.P., Chen, C.W.: Classification of age groups based on facial features. Tamkang Journal of Science and Engineering 4(3), 183–191 (2001)
2. Chojnacki, S., Kłopotek, M.A.: Grupowanie stron w polskim internecie. In: Proceedings of Artificial Intelligence Studies, pp. 1–8 (2012)
3. Ciesielski, K., Borkowski, P., Kłopotek, M.A., Trojanowski, K., Wysocki, K.: Wikipedia-based document categorization. In: Bouvry, P., Kłopotek, M.A., Leprévost, F., Marciniak, M., Mykowiecka, A., Rybiński, H. (eds.) SIIS 2011. LNCS, vol. 7053, pp. 265–278. Springer, Heidelberg (2012)
4. Ciesielski, K., Czerski, D., Dramiński, M., Kłopotek, M.A., Wierzchoń, S.T.: Semantic information within the BEATCA framework. Control and Cybernetics 39(2), 377–400 (2010)
5. Li, C., Li, K.-Q.: Hyperlink classifcation: A new approach to improve pagerank. In: DEXA 2007, pp. 274–277. IEEE Computer Society, Washington, DC (2007)
6. Dai, N., Davison, B.: Capturing page freshness for web search. In: Sigir 2010, Geneva, Switzerland, july 19-23, pp. 19–23 (2010)
7. Dêbowski, Ł.: Excess entropy in natural language: Present state and perspectives. Chaos 21, 037105 (2011)
8. Dondio, P., Barrett, S.: Computational trust in web content quality: A comparative evalutation on the wikipedia project (2007)
9. Fogg, B.J., Soohoo, C., Danielson, D.R., Marable, L., Stanford, J., Tauber, E.R.: How do users evaluate the credibility of web sites?: a study with over 2,500 participants. In: Proceedings of the 2003 Conference on Designing for User Experiences, DUX 2003, pp. 1–15. ACM, New York (2003)
10. Gitt, W.: In the Beginning was Information (Am Anfang war die Information). New Leaf Publishing Group (2006 edition) (1997)
11. Gyöngyi, Z., Garcia-Molina, H., Pedersen, J.: Combating web spam with trustrank. In: Proceedings of the Thirtieth International Conference on Very Large Data Bases, VLDB 2004, vol. 30, pp. 576–587. VLDB Endowment (2004)

12. Haveliwala, T.H.: Topic-sensitive PageRank: A context-sensitive ranking algorithm for web search. IEEE Trans. Knowl. Data Eng. 15(4), 784–796 (2003)
13. Kłopotek, M., Wierzchoń, S., Ciesielski, K., Dramiński, M., Czerski, D.: Conceptual Maps of Document Collections in Internet and Intranet. Coping with the Technological Challenge, 139 pages. IPI PAN Publishing House, Warszawa (2007)
14. Kłopotek, M.A., Wierzchoń, S.T., Ciesielski, K., Czerski, D., Dramiński, M.: Towards the notion of typical documents in large collections of documents. In: Zhang, Q., Segall, R.S., Cao, M. (eds.) Visual Analytics and Interactive Technologies: Data, Text and Web Mining Applications, pp. 1–18. IGI-Global (2011)
15. Kłopotek, M.A., Wierzchoń, S.T., Czerski, D., Ciesielski, K., Dramiński, M.: A Calculus for Personalized PageRank. In: Kłopotek, M.A., Koronacki, J., Marciniak, M., Mykowiecka, A., Wierzchoń, S.T. (eds.) IIS 2013. LNCS, vol. 7912, pp. 212–219. Springer, Heidelberg (2013)
16. Kolda, T.G., Procopio, M.J.: Generalized badrank with graduated trust. Technical Report SAND2009-6670, Sandia National Laboratories, Albuquerque, NM and Livermore, CA (October 2009).
17. Li, X., Lei, S.: Block-based segmentation and adaptive coding for visually lossless compression of scanned documents. In: Proc. ICIP, vol. III, pp. 450–453 (2001)
18. Ntoulas, A., Najork, M., Manasse, M., Fetterly, D.: Detecting spam web pages through content analysis. In: Proceedings of the 15th International Conference on World Wide Web, WWW 2006, pp. 83–92. ACM, New York (2006)
19. Page, L., Brin, S., Motwani, R.: The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, Previous number = SIDL-WP-1999-0120 (November 1999).
20. Shannon, C.E.: A mathematical theory of communication. The Bell System Technical Journal 27, 379–423, 623–656 (1948).
21. Simarro, F.M., González, P., Lozano, M.D., Vanderdonckt, J.: Quality models for automated evaluation of web sites usability and accessibility. In: International COS T294 Workshop on User Interface Quality Model (2005)
22. Sydow, M., Ciesielski, K., Wajda, J.: Introducing diversity to log-based query suggestions to deal with underspecified user queries. In: Bouvry, P., Kłopotek, M.A., Leprévost, F., Marciniak, M., Mykowiecka, A., Rybiński, H. (eds.) SIIS 2011. LNCS, vol. 7053, pp. 251–264. Springer, Heidelberg (2012)
23. Wróblewska, A., Sydow, M.: Dependency-based extraction of entity-relationship triples from polish open-domain texts, vol. 7(30), pp. 61–70. Publ. House of University of Natural Sciences and Humanities, Siedlce (2012)
24. Wu, B., Davison, B.D.: Identifying link farm spam pages. In: Proceedings of the 14th International World Wide Web Conference, pp. 820–829. ACM Press (2005)

Application of Combined Classifiers to Data Stream Classification

Michał Woźniak

Department of Systems and Computer Networks
Wroclaw University of Technology
Wyb. Wyspianskiego 27, 50-370 Wroclaw, Poland
michal.wozniak@pwr.wroc.pl

Abstract. The progress of computer science caused that many institutions collected huge amount of data, which analysis is impossible by human beings. Nowadays simple methods of data analysis are not sufficient for efficient management of an average enterprize, since for smart decisions the knowledge hidden in data is highly required, as which multiple classifier systems are recently the focus of intense research. Unfortunately the great disadvantage of traditional classification methods is that they "assume" that statistical properties of the discovered concept (which model is predicted) are being unchanged. In real situation we could observe so-called concept drift, which could be caused by changes in the probabilities of classes or/and conditional probability distributions of classes. The potential for considering new training data is an important feature of machine learning methods used in security applications or marketing departments. Unfortunately, the occurrence of this phenomena dramatically decreases classification accuracy.

Keywords: machine learning, classifier ensemble, data stream, concept drift, incremental learning, forgetting.

1 Introduction

Designing such solutions we should take into consideration that in the modern world the most of the data arrive continuously and it causes that smart analytic tools should respect this nature and be able to interpret so-called data streams. They should take into consideration that:

- the statistical dependencies among the observations of given objects and their classifications could change,
- data can come flooding in the analyzer what causes that it is impossible to label all records.

This phenomena is called *concept drift* [1] and it comes in many forms, depending on the type of change. In general, the following approaches can be considered to deal with the mentioned above problem

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 13–23, 2013.

- Rebuilding a classification model if new data becomes available, which is very expensive and impossible from a practical point of view, especially if the concept drift occurs rapidly.
- Detecting concept changes in new data and if these changes are sufficiently "significant", then rebuilding the classifier.
- Adopting an incremental learning algorithm for the classification model.

We will concentrate on the last proposition. Adapting the learner is a part of an incremental learning [2]. The model is either updated (e.g., neural networks) or needs to be partially or completely rebuilt (as CVFDT algorithm [3]). Usually we assume that the data stream is given in a form of data chunks. When dealing with the sliding window the main question is how to adjust the window size. On the one hand, a shorter window allows focusing on the emerging context, though data may not be representative for a longer lasting context. On the other hand, a wider window may result in mixing the instances representing different contexts. Therefore, certain advanced algorithms adjust the window size dynamically depending on the detected state (e.g., FLORA2 [1]) or algorithms can use multiple windows [4]. One of the important group of algorithms dedicated to stream classification exploits strength of ensemble systems, which work pretty well in static environments [5], because according to "no free lunch theorem" [6] there is not a single classifier that is suitable for all the tasks, since each of them has its own domain of competence. A strategy for generating the classifier ensemble [7] should guarantee its diversity improvement and consequently accuracy increasing. Let us enumerate the main propositions how to get a desirable set of individual classifiers:

- The individual classifiers could be train on different datasets, because we hope that classifiers trained on different inputs would be complementary.
- The individual classifiers can use the selected features only.
- Usually it could be easy to decompose the classification problem into simpler ones solved by the individual classifier. The key problem of such approach is how to recover the whole set of possible classes.
- The last and intuitive method is to use individual classifiers trained on different models or different versions of models.

2 Combined Classifier

According to Wolpert's *no free lunch* theorem there is not a single pattern recognition algorithm that is appropriate for all the tasks we are faced with, since each classifier has its own domain of competence [6]. Usually we have a pool of different classifiers at our disposal to solve a given problem. Therefore, methods that can exploit the strengths of individual classifiers are the focus of intense research [8]. There are many relevant works formulated conclusions regarding combined classification quality, such as [9] where a neural network ensemble was considered or [10] where authors dealt with majority voting applied to handwriting recognition. Turner [11] showed that averaging outputs of an infinite number of unbiased and independent classifiers can lead to the same response as the optimal Bayes classifier. Ho [12] underlined that a decision combination function must receive useful representation of each classifier's decision. Specifically, they considered several method based on decision ranks, such as Borda count. Finally, the landmark works devoted introducing bagging [13] and boosting [14,15] which are able to produce strong classifiers [16], in the (*Probably Approximately Correct*) theory [17] sense, on the basis of the weak ones.

It is important to notice that the classifier ensemble design does not differ from that of a classical pattern recognition [18] application, in which we select the most valuable features and choose the best classification method from the set of available ones. The design of a classifier ensemble aims to create a set of complementary/diverse classifiers and assign an appropriate combination method of individual classifier outputs as well as possible. We have to mention Ho's work [19] who distinguished two main approaches:

- Coverage optimization focus on the generation of a set of mutually complementary classifiers which can be combined to achieve optimal accuracy using a fixed decision combination function.
- Decision optimization concentrates on designing and training an appropriate decision combination function while a set of individual classifier is given in advance [20].

We can distinguish the important issues that must be taken into consideration when building classifier ensemble grouping them into the following problems:

- Proposing the topology i.e., interconnections among individual classifiers in the ensemble.
- Selecting a pool of diverse and complementary individual classifiers for the ensemble.
- Designing a combination rule, aimed at creating a mechanism that can exploit the strengths of the selected classifiers.

Let's focus on the problem of forming a valuable pool of classifiers called ensemble pruning. Selecting members of classifier ensemble with different kinds of components is a key feature of considered system design, because we have to notice that apart from increasing the computational complexity, combining similar classifiers should not contribute much to the combined classifier under construction. An ideal ensemble includes mutually complementary individual classifiers which are characterized by the high diversity and accuracy [21], because we expect that combined classifier accuracy increases according to the diversity increasing of individual classifier pool [22]. First, classifiers must be selected to obtain positive results from their fusion. In [23] Sharkley et al. proposed four level of diversity based on the majority voting rule answer, coincident error, and possibility of at least one correct answer of ensemble members. Brown et al. [24] reflected that it is not appropriate for the case where diversity of an ensemble is differ in differ subspace of the feature space.

3 Classifier Ensembles for Data Stream

Among the most popular ensemble approaches, the following are worth noting: the Streaming Ensemble Algorithm (SEA) [25] or the Accuracy Weighted Ensemble (AWE)[26]. Both algorithms keep a fixed-size set of classifiers. Incoming data are collected in data chunks, which are used to train new classifiers. If there is a free space in the ensemble, a new classifier joins the committee. Otherwise, all the classifiers are evaluated based on their accuracy and the worst one in the committee is replaced by a new one if the latter has higher accuracy. The SEA uses a majority voting strategy, whereas the AWE uses the more advanced weighted voting strategy. A similar formula for decision making is implemented in the Dynamic Weighted Majority (DWM) algorithm [27]. Nevertheless, unlike the former algorithms, the DWM modifies the weights and updates the ensemble in a more flexible manner. The weight of the classifier is reduced when the classifier makes an incorrect decision. Eventually the classifier is removed from the ensemble when its weight falls below a given threshold. Independently, a new classifier is added to the ensemble when the committee makes a wrong decision. The final group consists of algorithms that address the question of when drift occurs. Not all classification algorithms dealing with concept drift, require drift detection. Some evolving systems continuously adjust the model to incoming data [28]. This technique is called implicit drift detection [29] as opposed to explicit drift detection methods that raise a signal to indicate change. The detector can be based on changes in the probability distribution of the instances [30,31,32] or classification accuracy [33,34]. Among the machine learning methods dealing with concept drift, a new class has recently emerged [35], comprising algorithms that process data streams featuring a recurring context. Two additional requirements are imposed on algorithms in this class:

- the system should maintain knowledge of previously emerged contexts, and
- it should be effective in recognizing contexts and switching to a valid one.

Both issues can be effectively solved by an ensemble system. For example, in [35] the authors propose collecting context oriented learners in a ”global set” of classifiers along with their selection procedure. The ensemble consists of classifiers that achieve arguably better results than a random classifier. To address the second issue, certain algorithms use a conceptual representation of contexts. This idea originates from Turney’s definition of context-sensitive features [36] and the early work by Widmer [37] on meta-learning algorithms. It is based on mapping attributes into contextual clues representing some characteristic features of the respective context. In [38], the authors exploit an additional similarity measure between context representations for weighting a classifier’s contribution to the ensemble. The method presented in [39] incorporates a stream clustering algorithm, which aims to group incoming data batches automatically based on their conceptual representation. An incremental classifier is created (and updated if needed) for each cluster/context and stored in the pool. A similar solution can be found in [40], but here classifiers in the pool are adaptively weighted according to their performance on a recent data batch.

4 Exemplary Combined Algorithm for Data Stream Classification

We assume that the classified data stream is given in a form of data chunks denotes as $\mathcal{DS}_k$, where k is the chunk index. The concept drift could appear in the incoming data chunks. We do not detect it, but we try to construct self-adapting classifier ensemble. Therefore on the basis of the each chunk one individual is trained and we check if it could form valuable ensemble with the previously trained models. In our algorithm we propose to use the Generalized Diversity (denoted as $\mathcal{GD}$) proposed by Partridge and Krzanowski [41] to assess all possible ensembles and to choose the best one. $\mathcal{GD}$ returns the maximum values in the case of failure of one classifier is accompanied by correct classification by the other one and minimum diversity occurs when failure of one classifier is accompanied by failure of the other.

$$\mathcal{GD}(\Pi) = 1 - \frac{\sum_{i=1}^{L} \frac{i(i-1)p_i}{L(L-1)}}{\sum_{i=1}^{L} \frac{ip_i}{L}} \tag{1}$$

where L is the cardinality of the classifier pool (number of individual classifiers) and p_i stands for the probability that i randomly chosen classifiers from Π will fail on randomly chosen example.
We can also use modification of $\mathcal{GD}$ called Coincident Failure Diversity ($\mathcal{CFD}$) is the modification of Generalized Diversity proposed by Partridge and Krzanowski as well [41]

$$\mathcal{CFD}(\Pi) = \begin{cases} 0 & p_0 = 1 \\ \frac{1}{1-p_0} \sum\limits_{l=1}^{n} \frac{n-l}{n-1} p_l & p_0 < 1 \end{cases} \tag{2}$$

It returns 0 if the pool Π is not diverse and 1 if each classifier errs on different example.

Lets $P_a(\Psi_i)$ denotes frequency of correct classification of classifier Ψ_i and $itter(\Psi_i)$ stands for number of iterations which Ψ_i has been spent in the ensemble. We propose to establish the classifier's weight $w(\Psi_i)$ according to the following formulae

$$w(\Psi_i) = \frac{P_a(\Psi_i)}{\sqrt{itter(\Psi_i)}} \tag{3}$$

This proposition of classifier aging has its root in object weighting algorithms where an instance weight is usually inversely proportional to the time that has passed since the instance was read [42] and Accuracy Weighted Ensemble (AWE)[26], but the proposed method called Weighted Aging Ensemble (WAE) incudes two important modifications:

1. classifier weights depend on the individual classifier accuracies and time they have been spending in the ensemble,
2. individual classifier are chosen to the ensemble on the basis on the non-pairwise diversity measure.

The WAE pseudocode is presented in Alg.1 [43].

Algorithm 1. Weighted Aging Ensemble (WAE)

```
Require: input data stream, data chunk size, classifier training procedure, ensemble size $L$
$i := 1$
repeat
    collect new data chunk $DS_i$
    train classifier $\Psi_i$ on the basis of $DS_i$
    add $\Psi_i$ to the classifier ensemble $\Pi$
    if $i > L$ then
        $\Psi_{k+1} = \Psi_i$
        $\Pi_t = \emptyset$
        $GD_t = 0$
        for $j = 1$ to $L + 1$ do
            if $\mathcal{GD}(\Pi \backslash \Psi_i)$ (calculated according to (1)) $> GD_t$ then
                $\Pi_t = \Pi \backslash \Psi_i$
            end if
        end for
        $\Pi = \Pi_t$
    end if
    $w := 0$
    for $j = 1$ to $L$ do
        calculate $w(\Psi_i)$ according to (3)
        $w := w + w(\Psi_i)$
    end for
    for $j = 1$ to $L$ do
        $w(\Psi_i) := \frac{w(\Psi_i)}{w}$
    end for
    $i := i + 1$
until end of the input data stream
```

5 Experimental Investigations

In this section we will present an illustrative example of WAE for a data stream classification problem. The aims of the experiment were to assess if the proposed method of weighting and aging individual classifiers in the ensemble is valuable proposition compared with the methods which do not include aging or weighting techniques.

5.1 Set-up

All experiments were carried out on the SEA dataset describes in [25]. For each of the experiments we decided to form homogenous ensemble i.e., ensemble which consists of the Naive Bayes classifiers.

During each of the experiment we tried to evaluate dependency between data chunk sizes (which were fixed on 50, 100, 150, 200) and overall classifier quality (accuracy and standard deviation) for the following ensembles:

1. $w0a0$ - an ensemble using majority voting without aging.
2. $w1a0$ - an ensemble using weighted voting without aging, where weight assigned to a given classifier is inversely proportional to its accuracy.
3. $w1a1$ - an ensemble using weighted voting with aging, where weight assigned to a given classifier is calculated according to (3).

Method of ensemble pruning was the same for each ensemble and presented in Alg.1. The only difference was line 19 of the pseudocode what was previously described. All experiments were carried out in the Java environment using Weka classifiers [44].

5.2 Results

The results of experiment are presented in Fig.1-2. The first figure shows the accuracies of the tested ensembles for a chosen experiment. Unfortunately, because of the space limit we are not able to presents all extensive results, but they are

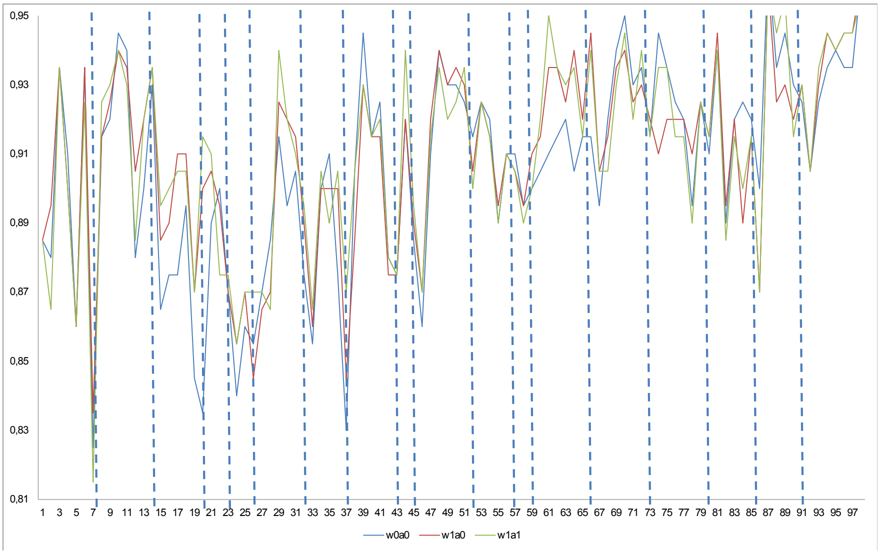


Fig. 1. Classification accuracy of the ensembles consist of Naive Bayes classifiers for the chunk size = 200. Vertical dotted lines indicate concept drift appearances.

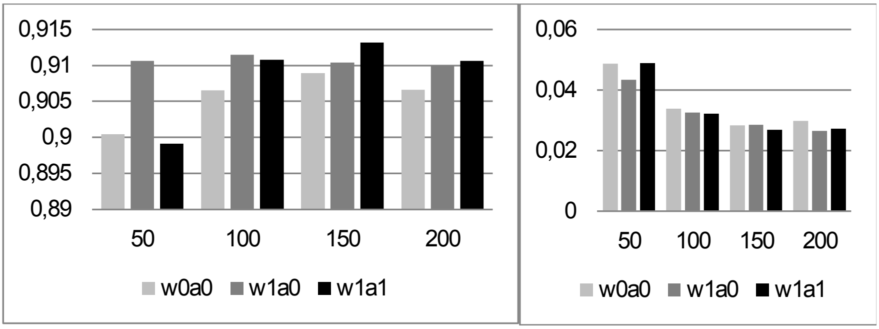


Fig. 2. Classification accuracy (left) and standard deviation (right) of Naive Bayes classifier for different data chunk sizes

available on demand. Fig.2 presents overall accuracy and standard deviation for the tested methods and how they depend on data chunk size.

On the basis of presented results we can formulate several observations. It does not surprise us that quality improvements for all tested method according to increasing data chunk size. Usually the WAE outperformed others, but the differences are quite small and only in the case of ensemble built on the basis of Naive Bayes classifiers the differences are statistical significant (t-test) [45] i.e., differences among different chunk sizes. The observation is useful because the bigger size of data chunk means that effort dedicated to building new models is smaller because they are being built rarely.

Another interesting observation is that the standard deviation is smaller for bigger data chunk and usually standard deviation of WAE is smallest among all tested methods. It means that the concept drift appearances have the weakest impact on the WAE accuracy.

6 Conclusions

We discussed the ensemble classifier approach applied to data stream classification task. We showed an idea and performance of WAE algorithm, which uses dynamic classifier ensemble i.e., its line-up is formed when new data chunk is come and the decision which classifier is chosen to the ensemble is made on the basis of General Diversity (diversity measure). The decision about object's label is made according to weighted voting where weight assigned to a given classifier depends on its accuracy (proportional) and how long the classifier participates in the ensemble (inversely proportional). It is worth noting that classifier ensemble is a promising research direction for aforementioned problem. Maybe its combination with a drift detection algorithm could have a higher impact to the classification performance.

Acknowledgment. The work was supported by the Polish National Science Center under a grant N N519 650440 for the period 2011-2014.

References

1. Widmer, G., Kubat, M.: Learning in the presence of concept drift and hidden contexts. Mach. Learn. 23, 69–101 (1996)
2. Muhlbaier, M.D., Topalis, A., Polikar, R.: Learn^{++}.nc: Combining ensemble of classifiers with dynamically weighted consult-and-vote for efficient incremental learning of new classes. IEEE Transactions on Neural Networks 20, 152–168 (2009)
3. Hulten, G., Spencer, L., Domingos, P.: Mining time-changing data streams. In: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 97–106 (2001)
4. Lazarescu, M.M., Venkatesh, S., Bui, H.H.: Using multiple windows to track concept drift. Intell. Data Anal. 8, 29–59 (2004)
5. Kuncheva, L.I.: Combining Pattern Classifiers: Methods and Algorithms. Wiley-Interscience (2004)
6. Wolpert, D.H.: The supervised learning no-free-lunch theorems. In: Proc. 6th Online World Conference on Soft Computing in Industrial Applications, pp. 25–42 (2001)
7. Wozniak, M., Grana, M., Corchado, E.: A survey of multiple classifier systems as hybrid systems. Information Fusion (2013)
8. Jain, A., Duin, R., Mao, J.: Statistical pattern recognition: a review. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 4–37 (2000)
9. Hansen, L., Salamon, P.: Neural network ensembles. IEEE Transactions on Pattern Analysis and Machine Intelligence 12, 993–1001 (1990)
10. Xu, L., Krzyzak, A., Suen, C.: Methods of combining multiple classifiers and their applications to handwriting recognition. IEEE Transactions on Systems, Man and Cybernetics 22, 418–435 (1992)
11. Tumer, K., Ghosh, J.: Analysis of decision boundaries in linearly combined neural classifiers. Pattern Recognition 29, 341–348 (1996)
12. Ho, T.K., Hull, J.J., Srihari, S.N.: Decision combination in multiple classifier systems. IEEE Trans. Pattern Anal. Mach. Intell. 16, 66–75 (1994)
13. Breiman, L.: Bagging predictors. Mach. Learn. 24, 123–140 (1996)
14. Schapire, R.E.: The strength of weak learnability. Mach. Learn. 5, 197–227 (1990)
15. Freund, Y.: Boosting a weak learning algorithm by majority. Inf. Comput. 121, 256–285 (1995)
16. Kearns, M.J., Vazirani, U.V.: An introduction to computational learning theory. MIT Press, Cambridge (1994)
17. Angluin, D.: Queries and concept learning. Mach. Learn. 2, 319–342 (1988)
18. Giacinto, G., Roli, F., Fumera, G.: Design of effective multiple classifier systems by clustering of classifiers. In: Proceedings of the 15th International Conference on Pattern Recognition, vol. 2, pp. 160–163 (2000)
19. Ho, T.K.: Complexity of classification problems and comparative advantages of combined classifiers. In: Kittler, J., Roli, F. (eds.) MCS 2000. LNCS, vol. 1857, pp. 97–106. Springer, Heidelberg (2000)
20. Roli, F., Giacinto, G.: Design of Multiple Classifier Systems. World Scientific Publishing (2002)
21. Krogh, A., Vedelsby, J.: Neural network ensembles, cross validation, and active learning. In: Advances in Neural Information Processing Systems, vol. 7, pp. 231–238 (1995)

22. Zenobi, G., Cunningham, P.: Using diversity in preparing ensembles of classifiers based on different feature subsets to minimize generalization error. In: Flach, P.A., De Raedt, L. (eds.) ECML 2001. LNCS (LNAI), vol. 2167, pp. 576–587. Springer, Heidelberg (2001)
23. Sharkey, A.J.C., Sharkey, N.E.: Combining diverse neural nets. Knowl. Eng. Rev. 12, 231–247 (1997)
24. Brown, G., Wyatt, J.L., Harris, R., Yao, X.: Diversity creation methods: a survey and categorisation. Information Fusion 6, 5–20 (2005)
25. Street, W.N., Kim, Y.: A streaming ensemble algorithm (sea) for large-scale classification. In: Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2001, pp. 377–382. ACM, New York (2001)
26. Wang, H., Fan, W., Yu, P.S., Han, J.: Mining concept-drifting data streams using ensemble classifiers. In: Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2003, pp. 226–235. ACM, New York (2003)
27. Kolter, J., Maloof, M.: Dynamic weighted majority: a new ensemble method for tracking concept drift. In: Third IEEE International Conference on Data Mining, ICDM 2003, pp. 123–130 (2003)
28. Zliobaite, I.: Change with delayed labeling: When is it detectable? In: Proceedings of the 2010 IEEE International Conference on Data Mining Workshops, ICDMW 2010, pp. 843–850. IEEE Computer Society, Washington, DC (2010)
29. Kuncheva, L.I.: Classifier ensembles for detecting concept change in streaming data: Overview and perspectives. In: 2nd Workshop SUEMA 2008 (ECAI 2008), pp. 5–10 (2008)
30. Gaber, M.M., Yu, P.S.: Classification of changes in evolving data streams using online clustering result deviation. In: Proc. of International Workshop on Knowledge Discovery in Data Streams (2006)
31. Markou, M., Singh, S.: Novelty detection: a review—part 1: statistical approaches. Signal Process. 83, 2481–2497 (2003)
32. Salganicoff, M.: Density-adaptive learning and forgetting. In: Machine Learning: Proceedings of the Tenth Annual Conference. Morgan Kaufmann, San Francisco (1993)
33. Klinkenberg, R., Joachims, T.: Detecting concept drift with support vector machines. In: Proceedings of the Seventeenth International Conference on Machine Learning, ICML 2000, pp. 487–494. Morgan Kaufmann Publishers Inc., San Francisco (2000)
34. Baena-García, M., del Campo-Ávila, J., Fidalgo, R., Bifet, A., Gavaldá, R., Morales-Bueno, R.: Early drift detection method. In: Fourth International Workshop on Knowledge Discovery from Data Streams (2006)
35. Ramamurthy, S., Bhatnagar, R.: Tracking recurrent concept drift in streaming data using ensemble classifiers. In: Proceedings of the Sixth International Conference on Machine Learning and Applications, ICMLA 2007, pp. 404–409. IEEE Computer Society, Washington, DC (2007)
36. Turney, P.D.: Exploiting context when learning to classify. In: Brazdil, P.B. (ed.) ECML 1993. LNCS, vol. 667, pp. 402–407. Springer, Heidelberg (1993)
37. Widmer, G.: Tracking context changes through meta-learning. Mach. Learn. 27, 259–286 (1997)

38. Bártolo Gomes, J., Ruiz, E.M., Sousa, P.A.C.: Learning recurring concepts from data streams with a context-aware ensemble. In: Chu, W.C., Wong, W.E., Palakal, M.J., Hung, C.C. (eds.) Proceedings of the 2011 ACM Symposium on Applied Computing (SAC), TaiChung, Taiwan, March 21-24, pp. 994–999. ACM (2011)
39. Katakis, I., Tsoumakas, G., Vlahavas, I.: Tracking recurring contexts using ensemble classifiers: an application to email filtering. Knowl. Inf. Syst. 22, 371–391 (2010)
40. Hosseini, M.J., Ahmadi, Z., Beigy, H.: Pool and accuracy based stream classification: A new ensemble algorithm on data stream classification using recurring concepts detection. In: Proceedings of the 2011 IEEE 11th International Conference on Data Mining Workshops, ICDMW 2011, pp. 588–595. IEEE Computer Society, Washington, DC (2011)
41. Partridge, D., Krzanowski, W.: Software diversity: practical statistics for its measurement and exploitation. Information and Software Technology 39, 707–717 (1997)
42. Klinkenberg, R., Renz, I.: Adaptive information filtering: Learning in the presence of concept drifts, pp. 33–40 (1998)
43. Wozniak, M., Kasprzak, A., Cal, P.: Application of combined classifiers to data stream classification. In: FQAS 2013. LNCS(LNAI), vol. 8132, pp. 579–588. Springer, Heidelberg (2013)
44. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. SIGKDD Explor. Newsl. 11, 10–18 (2009)
45. Alpaydin, E.: Introduction to Machine Learning, 2nd edn. The MIT Press (2010)

Efficacy of Some Primary Discriminant Functions in Diagnosing Planetary Gearboxes

Anna Bartkowiak[1] and Radoslaw Zimroz[2]

[1] Wroclaw University, Inst. of Computer Science, 50-383 Wroclaw PL
and Wroclaw School of Applied Informatics, 54-239 Wroclaw PL

[2] Wroclaw University of Technology, Diagnostics and Vibro-Acoustics
Science Laboratory, 50-051 Wroclaw PL
and KGHM CUPRUM R&D Center, Mechanical and Electrical
Engineering Group, 53-659 Wroclaw PL

Abstract. We consider the efficiency of some primary discriminant functions applied in planetary gearbox diagnostics. Real data for planetary gearboxes mounted in bucket wheel excavators working in field condition are elaborated. The aim is to perform condition monitoring (faulty or healthy) of such devices. The raw recorded data (vibration series emitted by the device) were first segmented and transformed to frequency domain using power spectra densities (PSD). Next, 15 variables denoting amplitudes of derived spectra were extracted. This yielded two data matrices A and B of size 1232×15, and 951×15, representing the faulty and the healthy device appropriately. The data are non-Gaussian and the covariances in both groups differ significantly.

Now, using Fisher's discriminant criterion and the kernel methodology, we construct from a learning sample (counting only 600 items) a discriminant function able to provide a discriminant score for distinguishing between the healthy and faulty state of a gearbox. The function proved to be very effective: Both for the learning and the testing data samples (600 and 1483 data vectors respectively) we got 100% correct assignments to the 'faulty' and 'healthy' class, with a conspicuous margin between the two classes. The results are visualized in a 2D plane.

Keywords: condition monitoring, discriminant analysis, healthy state, faulty state, kernels, nonlinearity, gearbox diagnostics.

1 Introduction

Condition monitoring of the state of a gearbox may be done in a multitude of ways, see the survey papers by [11] or [19] for further references. Gearboxes appear worldwide in many devices and machines. The state of a gearbox is important both for economy and safety reasons. A particular attention is paid to planetary gearboxes which represent a complicated architecture of compounds and are costly in exploitation.

The monitoring of the state of such devices is usually carried out by an analysis of vibration recorded by set of sensors (accelerometers) and specialized data

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 24–35, 2013.

acquisition systems (Bruel and Kjaer Pulse system was used here). Additionally, auxiliary channel equipped in tachometer (optical probe used often as rotating shaft speed indicator) was used to perform signal segmentation.

Generally the analysis is performed in two phases: *Firstly* the vibration signals are preprocessed and set of diagnostic features is extracted. The aim of this phase is to obtain from the recorded vibration data a kind of (multivariate) numerical data data suitable for further analysis. Features extraction in frequency domain (Fourier analysis like the PSD method) is the basic approach for rotating machinery at this stage. Another method often uses for this purpose wavelet transform, however this is done rather for localized damage than for the considered here distributed form of change of condition. *The second phase* uses various pattern recognition methods, like descriptive statistical methods, neural networks, discriminant functions and classification assignments. Reduction of dimensionality of the data is an important issue in both phases.

A very important point in the analysis is to have proper data for the analysis. For heavy-duty and high power planetary gearboxes the data are difficult to acquire. Therefore quite a lot of researchers use for their analysis data acquired in laboratory conditions using simulations. The behavior of devices working in field conditions might be different from that observed in a laboratory.

Our analysis is based on real data recorded in field conditions. The main contribution is that using these data and applying Kernel Discriminant function (KDF) it is possible to designate a decision boundary allowing for a 100% separation between the faulty and healthy data. The scores of the KDF are visualized in a 2-D plane yielding a visual proof that the two sets of data are really separable.

In the following, Section 2 introduces 3 primary discriminant functions, among them the Fisher's KDF (Kernel Discriminant Function). Section 3 contains a short description of some essential features of the data serving as the basis for our analysis, also some details on constructing the learning and testing samples. In Section 4 results of our analysis, when applying the 3 discriminant function, are presented. Section 5 contains some discussion and closing remarks.

2 Some Primary Discriminant Functions

There is a multitude of discriminant functions based on diversified criterions. We consider only the two group case when the data are subdivided into two classes (like: 'faulty' or 'healthy') represented by two groups of the observed data. The sought discriminant function should produce for each data vector a score (called also decision score) allowing to assign that vector to one of the considered classes. We will consider only some *basic* discriminant functions which - to our opinion - are advised to start with, when attempting to build a discriminant function providing a decision boundary between two groups of observed data. In next subsections we consider in turn:

(i) Linear discriminant function formulated and solved as ordinary multivariate regression function with predicted values $y = +1$ for one of the groups and $y = -1$ for the other group.

(ii) Classical canonical discriminant function based on Fisher's criterion (called often FDA, Fisher's discriminant analysis).
(iii) Discriminant functions using kernels providing a nonlinear approach.

Other methods of discriminant analysis - and there are many of them - may be found, among others, in [9, 10, 16].

Denotations:

◇ $\mathbf{X} = \{x_{ij}\},\ i = 1, \ldots, n,\ \ j = 1, 2, \ldots, d$ — the recorded matrix of observations, with d columns denoting variables, and n, the number of rows of the matrix. Each row of $\mathbf{X}$ constitutes a data vector $\mathbf{x}_i$ containing data recorded for the ith individual, called also data item. The vector $\mathbf{x}_i$ is written as the column vector $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{id}]^T$; it may be also considered as point located in the data space R^d. Each of the data vectors $\{\mathbf{x}_i\}$ may belong to only one of the two groups of data.

◇ $\mathbf{y} = \{y_i\},\ i = 1, \ldots, n$ – the vector of group labels. Each data vector has its label indicating to which group (class) the given (ith) data vector is belonging. The group membership is crisp: each data vector may belong only to one of the two considered classes.

2.1 Linear Discriminant Function via Linear Regression Analysis

We consider here the regression model

$$y_i = b_1 x_{i1} + b_2 x_{i2} + \ldots + b_d x_{id} + b_0 + e_i, \quad i = 1, \ldots, n \tag{1}$$

The vector $\mathbf{y} = [y_1, y_2, \ldots, y_n]^T$ denotes here the predicted variable, which takes specifically only two values: -1, if the *ith* data vector comes from the faulty class, and $+1$, if the *ith* data vector comes from the healthy class.

The coefficients $b_1, \ldots, b_d, b_0$ appearing in (1) are called regression coefficients, they are usually unknown and have to be estimated in a supervised way, that is from a learning sample, which contains known values od the data matrix $\mathbf{X}$ and of the vector $\mathbf{y}$. The regression coefficients are commonly estimated using the principle of Least Squares (LS), which minimizes the following criterion:

$$SS_e = \sum_{i=1}^{n} (e_i)^2 = \sum_{i=1}^{n} (y_i - b_1 x_{i1} - \ldots - b_d x_{id} - b_0)^2 .$$

To obtain the solution, that is, the vector $\mathbf{b} = [b_1, b_2, \ldots, b_d, b_0]^T$ which yields the minimum of the quadratic form SS_e, we have firstly to define $\mathbf{X}^+$, the augmented data matrix $\mathbf{X}$, as:

$$\mathbf{X}^+ = [\mathbf{X}, \mathbf{1_n}],$$

where $\mathbf{1_n}$ denotes a column of n ones.

With known $\mathbf{X}^+$ and $\mathbf{y}$, the estimates of $\mathbf{b}$, denoted in the following as $\hat{\mathbf{b}}$, are obtained as the solution of the following set of linear equations:

$$(\mathbf{X}^+)\,\hat{\mathbf{b}} = \mathbf{y}, \tag{2}$$

In Matlab the solution $\hat{\mathbf{y}}$ may be obtained simply via the matlab left divide function.

To obtain $\hat{\mathbf{y}}$, the predicted scores of class assignment of vectors contained in a data matrix say $\mathbf{X}_{learn}$ we add to that matrix a column of ones $\mathbf{1}_n$ obtaining the augmented data matrix $\mathbf{X}^+_{learn}$, and perform the multiplication

$$\hat{\mathbf{y}}_{learn} = (\mathbf{X}^+_{learn})\,\hat{\mathbf{b}}.$$

Similarly, using the vector $\hat{\mathbf{b}}$ yielded by the learning sample, we may obtain the predicted values $\hat{\mathbf{y}}_{test}$ for the test sample $\mathbf{X}^+_{test}$ as

$$\hat{\mathbf{y}}_{test} = (\mathbf{X}^+_{test})\,\hat{\mathbf{b}}.$$

Predicted values $\hat{\mathbf{y}}$ near to -1 (generally: negative values) indicate similarity to the class 'faulty', and values of $\hat{\mathbf{y}}$ near to $+1$ (generally: positive values) indicate similarity to the class 'healthy'.

2.2 FDA, Classical Canonical Discriminant Analysis Based on Fisher's Criterion

The BW-Projections Using the between and within Groups Scatter. The main idea of the method is to find a vector $\mathbf{u} = (u_1, ..., u_d)^T$ serving for construction of a new variable (feature) Z with values z_i obtained from the observed data vector $\mathbf{x}_i$ in the following manner:

$$z_i = \mathbf{x}_i^T \mathbf{u}. \quad i = 1, \ldots, n. \tag{3}$$

The values of the new variable, denoted as $z_i,\ i = 1, \ldots, n$, may be considered as projections of the data vectors $\mathbf{x}_i$ onto a new axis called Z.

In the following we will consider the case when the the entire set of data vectors is subdivided into $G = 2$ groups (classes). For the ith data point of the gth group ($i = 1, \ldots, n_g,\quad g = 1, 2$), we denote the projection of the data point $\mathbf{x}_i^{(g)}$ as $z_i^{(g)}$, and the general transformation may be read as:

$$z_i^{(g)} = (\mathbf{x}_i^{(g)})^T \mathbf{u}.$$

Thus, we obtain G=2 groups of projections:

$$\underbrace{z_1^{(1)}, ..., z_{n_1}^{(1)}}_{\text{1st group}}, \underbrace{z_1^{(2)}, ..., z_{n_2}^{(2)}}_{\text{2nd group}} \tag{4}$$

For values of the variable Z appearing in formula (4) above we may calculate group means $\bar{z}.^{(g)},\ (g = 1, 2)$, and the overall mean $\bar{z}.^{(\cdot)} \equiv \bar{\bar{z}}$ (this is the mean of all data vectors, without subdivision into the 2 groups). Moreover, basing on the values given in (4), we may define and calculate ZW, the *within group scatter* of the variable Z, and ZB, the *between group scatter* of the variable Z, as

$$ZW = \sum_{g=1}^{2} \sum_{i=1}^{ng} (z_i^{(g)} - \bar{z}.^{(g)})^2, \quad ZB = \sum_{g=1}^{2} n_g (\bar{z}.^{(g)} - \bar{\bar{z}})^2.$$

Which projections are the best for the purpose of discriminant analysis? Consider the following issues. For a good discriminant function
◇ points-projections *belonging to different groups* of data *should be separated as much as possible*,
◇ points-projections *belonging to the same groups* should be possibly much concentrated around their group-means.

Taking this into account, R.A. Fisher has formulated the following criterion, which is called today **Fisher's criterion**:

$$J(z) = \frac{ZB}{ZW} = \frac{\sum_{g=1}^{2} n_g(\bar{z}.^{(g)} - \bar{z}.^{(\cdot)})^2}{\sum_{g=1}^{2}\sum_{i=1}^{n_g}(z_i^{(g)} - \bar{z}.^{(g)})^2}. \tag{5}$$

To indicate for some differentiation between the groups, the criterion $J(z)$, evaluated for real data, should have a value larger than 1; the larger the better.

To find the best vector **u** maximizing the criterion $J(z)$ we express (5) in terms of **u** and **X**; usually **X** is centered to zero, that is, all column means are equal zero. After some algebra manipulations we obtain

$$J(z) = J(\mathbf{u}; \mathbf{X}) = \frac{\mathbf{u}^T\mathbf{B}\mathbf{u}}{\mathbf{u}^T\mathbf{W}\mathbf{u}}\,, \tag{6}$$

where

$$\mathbf{B}_{d\times d} = \sum_{g=1}^{2} n_g(\bar{\mathbf{x}}.^{(g)} - \bar{\mathbf{x}}.^{(\cdot)})(\bar{\mathbf{x}}.^{(g)} - \bar{\mathbf{x}}.^{(\cdot)})^T, \tag{7}$$

$$\mathbf{W}_{d\times d} = \sum_{g=1}^{2}\sum_{i=1}^{ng}(\mathbf{x}_l i^{(g)} - \bar{\mathbf{x}}.^{(g)})(\mathbf{x}_i^{(g)} - \bar{\mathbf{x}}.^{(g)})^T. \tag{8}$$

The above equations show clearly that the criterion J may be considered as a function of the sought unknown vector **u** and the known data matrix **X**, which serves for calculations of the between and within group scatter matrices **B** and **W**, each of size $d \times d$. Applying Lagrange's methodology and using the additional restraint $\mathbf{u}^T\mathbf{W}\mathbf{u} = 1$, one arrives to the two-matrices eigenvalue problem:

$$(\mathbf{B} - \lambda\mathbf{W})\mathbf{u} = \mathbf{0}. \tag{9}$$

which is well known in matrix algebra. It is shown, that for $G = 2$ groups there is an unique vector **u** satisfying eq. (9). Moreover, the constant λ is equal to the quotient $J(z)$ appearing in formula (5), evaluated at the solution **u**. In other words, the constant λ is equal to the ratio of the between group scatter divided by the within group scatter defined above. It satisfies the inequality:

$$\lambda \geq 0$$

Large values of λ ($\gg 1.0$) indicate a big differentiation between projections of the two groups of data.

Using the obtained vector $\mathbf{u}$ for the projection $z = \mathbf{x}^T\mathbf{u}$ we obtain a new variable Z satisfying Fisher's criterion (5). The technique of constructing such variables is called *Canonical Discriminant Analysis* (*CDA*), the obtained projections Z are called *Canonical variates* or *CDA scores*. The CDA scores may be calculated and displayed both for the learning and testing samples of data.

The BT Projections Using the between Groups and Total Scatter

The criterion $K(z)$ formulated defined in formula (5) might be formulated in another way. Let T denote the total data scatter matrix:

$$\mathbf{T}_{d\times d} = \sum_{g=1}^{2}\sum_{i=1}^{ng}(\mathbf{x}_l i^{(g)} - \bar{\mathbf{x}}.^{(\cdot)})(\mathbf{x}_i^{(g)} - \bar{\mathbf{x}}.^{(\cdot)})^T, \tag{10}$$

with $\bar{\mathbf{x}}.^{(\cdot)}$ denoting the overall mean calculated from all the data.

Now our aim is to find the vector $\mathbf{v}$ maximizing the following index denoting the ratio of the between group to the total scatter

$$\eta = \frac{\mathbf{v}'\mathbf{B}\mathbf{v}}{\mathbf{v}'\mathbf{T}\mathbf{v}} \tag{11}$$

The solution of the problem (11) is obtained in a similar way as that of (5); it leads to the two-matrix problem

$$(\mathbf{B} - \eta\mathbf{T})\mathbf{v} = \mathbf{0}. \tag{12}$$

Now the value *eta* evaluated at the solution $\mathbf{v}$ of (12) takes values from the interval

$$0 \le \eta \le 1.0$$

Big values of η indicate for a good separability of the groups.

Similarly as for Fisher's projections using the $\mathbf{B}$ and $\mathbf{W}$ matrices, the values z_i of the *BT-projections* are obtained as:

$$z_i = \mathbf{x}_i^T\mathbf{v}, \tag{13}$$

where $\mathbf{v}$ is the solution of (12). The projections z_i obtained that way may be visualized similarly as those obtained from the BW-projections.

Let us emphasize: The canonical discriminant variables represent projections to lower dimension subspaces. They serve in first place for visualization of the projected points. The differentiation of the projected points follows from the assumed Fisher's criterion. The group assignment is relatively easy only in the 2-group case. Having $G > 2$ groups, one should use for class assignments a specialistic tool, e.g. Mahalanobis distances or k-nearest neighbor methods.

2.3 Discriminant Functions Using Kernels

Kernel methods may be defined and used in a number of ways, see for example [15, 13, 14, 6–8]. One might say shortly that this is canonical discriminant analysis carried out in an extended space $\mathcal{F}$ obtained by a non-linear mapping of the original data. The applied mapping takes into account various nonlinear relations between the observed variables which makes that in the extended feature space $\mathcal{F}$ the classical algorithms (like CDA) become more powerful.

Let φ denote the transformation carrying out the mapping of the observed vector $\mathbf{x}$ ($\mathbf{x} \in R^d$) to the extended feature space $\mathcal{F}$. The function φ is called the mapping function. The extended space $\mathcal{F}$ must have defined an operator called scalar product for every pair of its elements $\varphi(\mathbf{x}_i)$, and $\varphi(\mathbf{x}_j)$. The scalar product is defined via so called dot product for pairs of data points obtained as images of the mapping φ of points $\mathbf{x}_i$ and $\mathbf{x}_j$ from R^d to $\varphi(\mathbf{x}_i)$, and $\varphi(\mathbf{x}_j)$ in $\mathcal{F}$:

$$k(\mathbf{x}_i, \mathbf{x}_j) = (\varphi(\mathbf{x}_i) \cdot \varphi(\mathbf{x}_j)), \tag{14}$$

where the symbol '·' means the scalar product between the vectors-images $\varphi(\mathbf{x}_i)$ and $\varphi(\mathbf{x}_j)$ evaluated in $\mathcal{F}$. Let us point out, that the kernel matrix $\mathbf{K}$ is of size $n \times n$, with n being the size of the learning sample, thus its evaluation may be computationally demanding.

Performing the mapping $\varphi(\mathbf{x})$ might be quite cumbersome – this depends from the applied mapping φ, in particular from the dimension of $\mathcal{F}$ which is usually much larger than that of R^d. However, by looking at the transformations carried out using particular mapping functions φ it was found that the elements of the kernel matrix $\mathbf{K}$ defined in (14) may be evaluated directly from the points $\mathbf{x}_i$ and $\mathbf{x}_j$ located in R^d *without* considering their images in $\mathcal{F}$. Then, if we are able to formulate the solution of our task in terms using only the elements of the kernel matrix $\mathbf{K}$, we are the winners: we do not need anymore the mapped points from $\mathcal{F}$. This possibility is called 'the kernel trick'.

For example, this is true when considering as φ the Gaussian or polynomial kernels and having as task canonical discriminant analysis or principal component analysis. Detailed algorithms how to do it in the case of discriminant analysis for G groups of data are shown, e.g., in [14] (when using the scatter matrices $\mathbf{B}, \mathbf{W}$) or [6] (when using the scatter matrices $\mathbf{B}, \mathbf{T}$). The algorithms are somehow lengthy and we do not show them here.

More details on the methods and examples of applications may be found in [6–8, 2–4].

3 Experiments with the Gearbox Data

In the following we will show an analysis conducted using true data, that is recorded from machines working in field conditions. The data were recorded by Bartelmus and Zimroz [1]. We had 2 groups of data contained in 2 data matrices A of size 1232 × 15 and B of size 951 × 15 obtained from vibration data of two gearboxes, the one in a faulty and the other in a healthy condition. The data were the basis of an elaboration presented in [1].

The raw recorded data (vibration series of the devices) were firstly segmented and transformed to frequency domain using power spectra densities. Next, 15 variables denoting amplitudes of derived spectra were obtained. This yielded two data matrices (A and B of size 1232 × 15, and 951 × 15 appropriately) characterizing the state of the faulty and the healthy device. The rows of the matrices constitute data vectors (called also data items) living in the 15-D space R^{15}. Thus we have two samples of faulty and healthy data containing n1=1232 and n2=951 data items appropriately.

Taking as a new feature the sum of all the 15 variables, Bartelmus and Zimroz [1] were able to classify – on the base of the proposed feature – about 80 % of all data vectors. To classify the remainder, they needed an external variable indicating for the actual load of the working excavator.

The data were more thoroughly investigated in [17, 5, 18]. It appeared that the distribution of the variables is not Gaussian, the data contains a considerable number of outliers, moreover, the covariance structure in the two groups (faulty and healthy) is markedly different. It became evident, that a proper analysis of the data needs non-linear methods [18].

Now we have subdivided the entire data set into the learning and testing samples. The learning sample was obtained from randomly chosen 300 rows of the A matrix and other 300 rows of the B data matrix; this yielded the learning sample of size 600 × 15. The remainder of the data (932 rows from A and 651 rows from B) yielded the testing sample of size 1583 × 15.

4 Results

4.1 Results When Using Linear Regression Method

The linear regression was supposed to predict the class score given by the values:
$y_i = -1$ for data vectors i belonging to the faulty class, and
$y_i = +1$ for data vectors i belonging to the healthy class.

The calculations were done in Matlab. The estimates of the regression coefficients $\hat{\mathbf{b}} = [\hat{b}_1, \hat{b}_2, \ldots, \hat{b}_{15}, \hat{b}_0]^T$ were obtained using the learning sample and the Matlab function mldivide, as described in Section 2.1.

The obtained estimates $\hat{\mathbf{b}}$ were used to obtain the predicted values of the regression. This yielded the predicted vectors of discriminant scores $\hat{\mathbf{y}}_{learn}$ for the learning and $\hat{\mathbf{y}}_{test}$ for the testing samples. The obtained scores are visualized in Figure 1.

We have used the following decision rule:

If for a data vector $\mathbf{x}_i$ *its predicted value satisfies* $\hat{y}_i \leq 0$, *classify that data vector as 'faulty', otherwise — classify it as 'healthy'.*

Looking at the results shown in Fig. 1 we find that:

In the *learning sample* we have 3 incorrect assignments for a total of 600 items. The erroneous assignments happened in the faulty group, where 3 faulty items have been assigned to the healthy group. This makes the **error rate equal to 0.0050**.

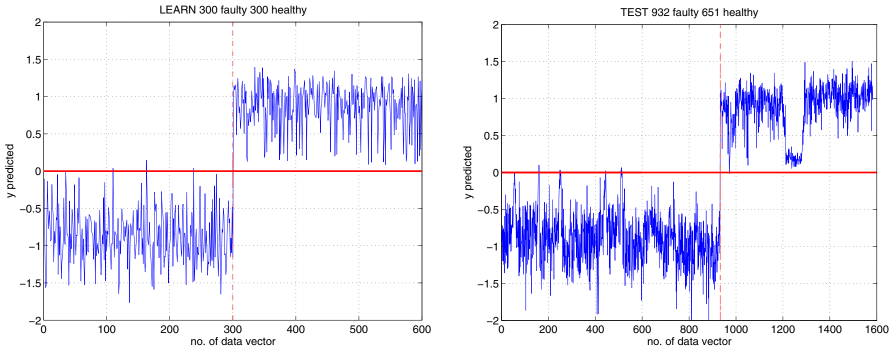


Fig. 1. Discriminant scores obtained via linear regression method with target values $y = -1$ for the faulty and $y = +1$ for the healthy state of the machine. Left panel: learning sample, first 300 faulty and next 300 healthy items. Right panel: testing sample, first 932 faulty and next 651 healthy items.

In the *testing sample* we have 11 incorrect assignments for a total of 1583 items: 10 faulty items were assigned as healthy, and 1 healthy item was assigned as faulty. This makes the **error rate equal to 0.0069**.

Thus the linear discriminant function is a good choice for using it as a discriminant function, especially as it is very easy to carry out.

4.2 Results from Canonical Discriminant Analysis

The main result from CDA - when using BW projections - is shown in Figure 2.

We have calculated canonical discriminant functions CV1 and CV2 using the BW-projections based on Fisher's Criterion (5) and using own software programmed in Matlab. The main result is shown in Figure 2. Looking at that figure one can grasp at once how different is the spread of points in the two groups of data.

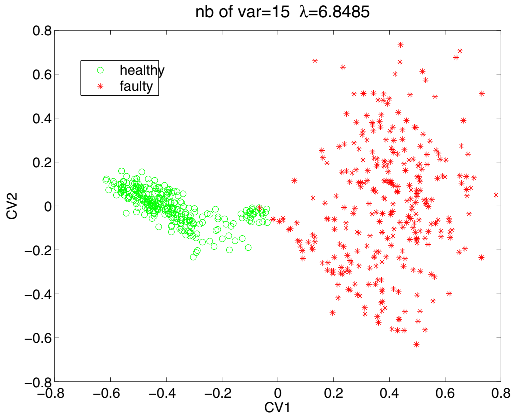


Fig. 2. Display of two canonical discriminant functions constructed from d=15 dimensional data. The first canonical variate CV1 has discriminative power $\lambda = 6.8485$. The second CV2 has discriminative power equal to 0, however is helpful in perceiving the spread of the data points in the two groups.

The first canonical variate CV1 has discriminative power lambda=6.8485. We have constructed a second canonical variate which has discriminative power equal to 0, however helps in visualization of the two data clouds representing the faulty and healthy data points.

Concerning class assignments, we have the following results:
Learning sample: 3 erroneous assignments in the faulty class. Thus the error rate equals $3/600 = 0.005$.

Testing sample: There are 24 erroneous assignments in the faulty class. Thus the error rate equals $24/1583 = 0.0154$.

4.3 Results of Kernel Canonical Discriminant Analysis

The calculations were performed using BT projections elaborated by Baudat and Anouar [6] and using their software gda downloaded from www.kernel-machines.org. As previously, the entire observed data were subdivided into the learning sample and test sample – the same, as used in subsections 4.1 and 4.2. The original d=15 dimensional data were mapped to the feature space $\mathcal{F}$ where the kernel matrix $K = \{k(\mathbf{x}_i, \mathbf{x}_j\},\ i, j = 1, \ldots, n$ was constructed from the data vectors belonging to R^d using the Gaussian kernels with parameter $\sigma = 0.7$:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \exp(\,-(\mathbf{x}_i - \mathbf{x}_j)^T(\mathbf{x}_i - \mathbf{x}_j))/\sigma \tag{15}$$

In such a way a non-linear mapping of the original data into $\mathcal{F}$ was obtained.

The proper analysis was carried out in the extended space $\mathcal{F}$. The vector $\mathbf{v}$ performing the BT-projections onto new axis Z positioned now in $\mathcal{F}$ was obtained using the kernel matrix $\mathbf{K}$ whose elements were calculated using formula (15). Next, using only information from the kernel matrix $\mathbf{K}$, the projection

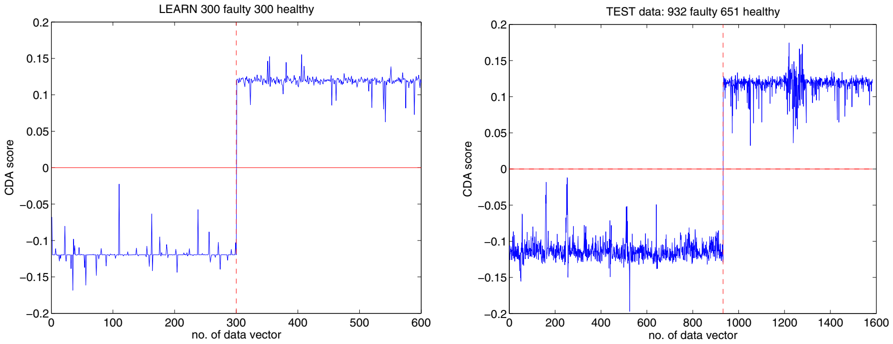


Fig. 3. Kernel discriminant function obtained from Gaussian kernel with $\sigma = 0.7$. Negative and positive scores indicate the faulty and healthy state of the machine. Left panel: learning sample, first 300 faulty and next 300 healthy items. Right panel: testing sample, first 932 faulty and next 651 healthy items. Subsequent data items are linked by line segments. All items in both of the classes are classified perfectly.

vector **v** was constructed. This allowed us to construct new variable Z satisfying the criterion (11), however valid for the mapped data in $\mathcal{F}$. Next, using the functions spreadGda and plotGda (see [6]), it was possible to recover the projections $z_i, i = 1, \ldots, n$ for each individual data vector from the learning sample ($n_{learn=600}$) and test sample ($n_{test} = 1283$). The obtained scores z_i obtained now are analogous to the scores z_i defined for the classical canonical discriminant analysis (see eq. (13)) however the obtained now (from spreadgda and plotgda) are projections from the non-linear mapping valid in $\mathcal{F}$. They are shown in Figure 3. One may notice that **the projections** from the faulty and healthy data vectors **are completely disjoint**. The criterion η amounts $\eta = 0.9938$.

5 Discussion and Concluding Remarks

Using real data gathered in field conditions we have investigated the usefulness of some primary discriminant functions in diagnosis of the faulty or healthy state of two planetary gearboxes. Thanks to applying a nonlinear transformation of the gathered data (using Gaussian radial basis functions) to an extended feature space $\mathcal{F}$ we have obtained a perfect decision boundary between the faulty and healthy data. The function designating decision boundary was obtained from a relative small learning sample counting 300 faulty and 300 healthy data vectors. The behavior of the obtained decision boundary was checked using a larger test sample counting 1583 data vectors. All of them were correctly classified to the proper group of their origin. We were also able to represent the projections from the extended feature space in an ordinary 2-D plane, as shown in figs. 1 and 3.

The elaborated data were not-normal and with covariance matrices differing substantially [17, 18]. Despite that the constructed discriminants have worked surprisingly effectively.

References

1. Bartelmus, W., Zimroz, R.: A new feature for monitoring the condition of gearboxes in nonstationary operating systems. Mechanical Systems and Signal Processing 23(5), 1528–1534 (2009)
2. Bartkowiak, A., Evelpidou, N.: Visualizing Some Multi-Class Erosion Data Using Kernel Methods. In: Rizzi, A., Vichi, M. (eds.) Proceedings in Computational Statistics, 17th Symposium Held in Rome, pp. 805–812. Physica-Verlag (2006)
3. Bartkowiak, A., Evelpidou, N.: Visualizing of some multi-class erosion data using GDA and supervised SOM. In: Saeed, K., et al. (eds.) Biometrics, Computer Security Systems and Artificial Intelligence Applications, pp. 13–22. Springer (2006)
4. Bartkowiak, A., Evelpidou, N., Vasilopoulos, A.: Visualization of Five Erosion Risk Classes using Kernel Discriminants. In: Pejaś, J., Saeed, K. (eds.) Advances in Information Processing and Protection, ch. 10, pp. 169–178. Springer (2007) ISBN: 978-0-387-73136-0 (Print) 978-0-387-73137-7 (Online)
5. Bartkowiak, A., Zimroz, R.: Outliers analysis and one class classification approach for planetary gearbox diagnosis. Journal of Physics: Conference Series 305(1), art. no. 012031 (2011)

6. Baudat, G., Anouar, F.: Generalized discriminant analysis using a kernel approach. Neural Computation 12, 2385–2404 (2000)
7. Camastra, F.: Kernel Methods for Computer Vision, Theory and Applications, `http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.23.9315`
8. Camastra, F.: Kernel Methods for unsupervided Learning. PhD thesis. DISI-TH, Universita di Genova (2004)
9. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. Wiley (2001)
10. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning; Data Mining, Inference and Prediction, 2nd edn. Springer, New-York (2010)
11. Jardine, A.K.S., Lin, D., Banjevic, D.: A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mech. Syst. Signal Process. 20, 1483–1510 (2006)
12. Liu, Z., Qu, J., Zuo, M.J., Xu, H.B.: Fault level diagnosis for planetary gearboxes using hybrid kernel feature selection and kernel Fisher discriminant analysis. Int. J. Adv. Manuf. Technol., doi:10.1007/s00170-012-4560-y
13. Mika, S., Rätsch, G., Weston, J., Schölkopf, B., Smola, A., Müller, K.-R.: Constructing descriptive and discriminative nonlinear features: Rayleigh coefficients in kernel features spaces. IEEE Trans. PAMI 25(5), 623–628 (2001)
14. Müller, K.-R., Mika, S., Rätsch, G., Tsuda, K., Schölkopf, B.: An introduction to kernel-based learning algorithms. IEEE Trans. on Neural Networks 12(2), 181–202 (2001)
15. Shawe-Taylor, J., Christianini, N.: Kernel Methods for Pattern Analysis. Cambridge University Press, UK (2004)
16. Trendafilov, N., Vines, K.: Simple and interpretable discrimination. Comput. Stat. Data Anal. 53, 979–989 (2009)
17. Zimroz, R., Bartkowiak, A.: Investigation on spectral structure of gearbox vibration signals by principal component analysis for condition monitoring purposes. Journal of Physics: Conference Series 305(1), art. no. 012075 (2011)
18. Zimroz, R., Bartkowiak, A.: Multidimensional data analysis for condition monitoring: features selection and data classification. In: CM 2012—MFPT 2012, London, June 11-14. Electronic Proceedings, art no. 402, pp. 1–12 (2012)
19. Zimroz, R., Bartkowiak, A.: Two simple multivariate procedures for monitoring planetary gearboxes in non-stationary operating conditions. Mech. Syst. Signal Process. 38(1), 237–247 (2013)

Identification of Persons by Virtue of Hand Geometry

Anna Plichta[1], Tomasz Gaciarz[1], and Szymon Szomiński[2]

[1] Cracow Univeristy of Technology Department of Computer Science, ul.Warszawska 24 Cracow 31-155, Poland

[2] AGH University of Science and Technology, Al. A. Mickiewicza 30 Cracow 30-059, Poland

Abstract. Biometry is a method of recognition and identification of people by means of their physical and behavioral features. These features can pertain to papillar ridges, face, hand, iris, handwriting, the way we type or even to the network of the veins within one's wrist. The biometric identification prevents the unauthorized person from accessing the ATM, personal computer, computer networks, mobile phones, household alarm systems etc.

Keywords: biometry, nearest neighbour method, image processing.

1 Introduction

Recently, biometric techniques have been one of the most flourishing branches of the ICT. As the amount of the processed data increases, the tightening of the access control is more and more important. The above-mentioned control pertains both to the room access and to the logical control over the individuals using the particular data or programs. The traditional techniques, based on magnetic or chip cards or on password systems no longer meet the contemporary security requirements.

The biometric methods of identification provide the high security level and are very convenient. They can be applied separately or together with the traditional access control solutions in order to prevent the room and computer networks from the nefarious use by the unauthorized people.

Currently, the most popular biometric techniques are based on:

1. papillar ridges
2. hand geometry
3. speech dynamics
4. iris pattern

Access systems based on hand geometry are often used to verify one's identity. Apart from being highly secure, they are quite cheap, easy to use. They are also thought not to breach the privacy of the users. First of all, the shape of the hand is not considered, at least by the majority of people, to be very personal or intimate feature, contrary to the fingerprints which in fact we remain

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 36–46, 2013.

on thousands of surfaces and things, every day. We are accustomed to giving handshakes and touching many things with our hand. Therefore, we are seldom afraid of the measuring appliance, whereas people who use iris-checking systems are often unwilling to look straight into camera lens. Secondly, features pertaining to the hand geometry do not provide the amount of information sufficient to identificate a person. Moreover, systems based on the hand geometry measurement avail of simple measurement methods and hence are inexpensive to make and easy to use. In such systems it takes much less time for the user to learn how to avail of the appliance in comparison to other biometric methods. All these features make the systems based on hand geometry measurement most widely approved biometric methods, as they seemingly do breach almost no cultural, social, psychological or religious rules.

Despite the fact that few working people lack hands, the system can be make available for, at least some, handicapped people. If necessary, the system can be integrated with other biometric techniques, especially with papillar ridges measurement or palmprints checking.

The hand biometry can function improperly if the hand is seriously wounded, break or swollen and the upper extremity injuries are perhaps the most frequent effects of accidents while working. To partially solve that problem, both hands of each worker should be put into system.

Biometric systems based on hand geometry avail of the information which pertains to the physical dimensions of the hand such as length and width of the fingers (measured in various places) or of some geometric features of the metacarpus. It should be stressed, that other features such as fingerprints, palmprints pattern, infrared photography of the hand or the pattern of blood vessels of the hand are considered separate biometric methods. They are connected to the hand, but require different processing. The majority of the above-mentioned techniques can be used in one appliance, together with the system based on the hand geometry measurment.

The data pertaining to the shape of the hand are computed from the photography of the hand taken by virtue of some particular techniques of image processing. Then, the geometric features of the hand are determined. In order to compare the hand to the pattern in the system, various classification techniques are used. The process of comparison can be considered as determining the distance between the pattern and the hands being verified. If the hand of the user matches the pattern, the distance is small (the current image of the hand and the pattern are similar enough). Otherwise, the distance is large.[1,2,4,5]

2 Recognition Process

2.1 Data Acquisition Methods

The first step to identify the user is to acquire the data pertaining to the features of his hand - by means of taking photograph. It can be taken in many ways.

For the sake of the identification method:

1. the angles between fingers should be proper
2. the photo should be taken perpendicularly to the camera lens; otherwise, the surface of the hand and fingers is different (we avail of the surfaces of fingers and hand - these are some features we want to acquire)
3. the distance between the lens should be constant; otherwise, the hand put too close to the lens would be larger and all the features would be no longer valid.

There are some professional appliances acquiring such features available to buy, for instance Handkey II or ID3D Handkey. They enable for acquiring geometric features of the hands, fingerprints or palmprints. In these appliances the span between fingers is always the same due to used pegs and the hand is isolated from the background, so the taken photograph is of very good quality.

In household, we can fix the camera and the contrastive mat on which we put hand. That method lacks pegs so the additional module correcting finger span is required.

One can also solve that problem via constructing the appliance consisting of pane of glass and the camera. The hand should be put to the glass and the lens should be installed in a fixed distance to the pane. It is typical of background in such solution to vary each time. The finger span can also vary. However, these problems can be sorted out.

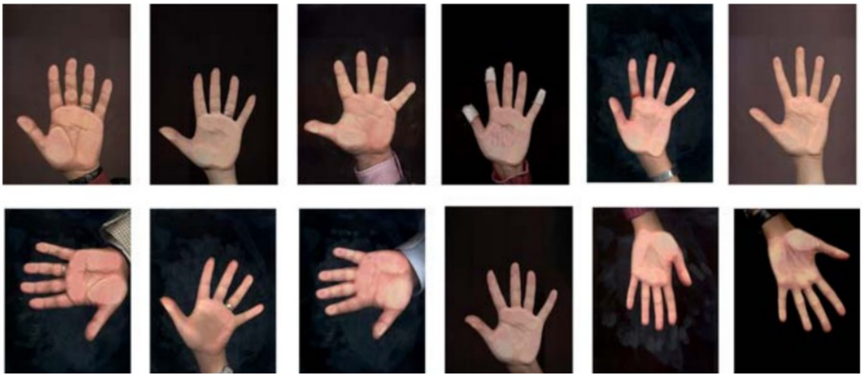


Fig. 1. Examples of the photo images of the hands on glass pane

The most user friendly method of these (and the most esthetic) is rotating the hand when the camera is fixed. For the further proceeding the system chooses the photo image being the most perpendicular to the lens. Unfortunately, the distance between the hand and the lens is problematic in this method. As we avail of many features pertaining to determining distance (fingers, hand length, circumferences, peripheries etc.), the image scaling module is required.

2.2 Segmentation

In all methods which do not make use of pegs between the fingers and thus allow for variable finger span the hand segmentation module is mandatory. It is also required that for each hand the lines between the fingertops and points between them do not adjoin. There are some flaws resulting in the necessity of rejecting the particular hand images.

2.3 Image Processing

The acquired photo image undergoes processing. It is assumed that the image had underwent scaling and segmentation. The processing comprises the following steps:

1. background detection, binarization,
2. rotation and shifting of the hand,
3. possibly, correcting the marks left by the ring,
4. background detection, binarization,
5. correcting the angles between the fingers,
6. cutting the wrist

Table 1. Angles between the fingers and the pivot line

Thumb	*Indexfinger*	*Middlefinger*	*Ringfinger*	*Littlefinger*
150	120	100	80	60

The above-mentioned figures show the steps to achieve the images of quality sufficient to be the samples in the carried out tests [4]

The samples are achieved as in the Fig.2:

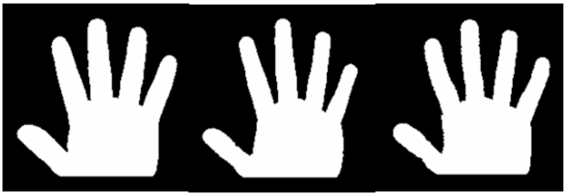


Fig. 2. The samples

The samples used in the tests were prepared and normalized by prof. dr Bulent Sankur [3] by virtue of the dedicated identification system availing of the hand geometry. The samples consisted of the photographic images taken from 755 individuals. From each individual 3 photographs were taken. The dimension of the photograph was 200 x 200 pixels.

3 The Features of the Hand Geometry

The following set of features should be found:

1. 5x finger length
2. 5x base area of the finger
3. 4x the largest circle inscribed in the upper part of the fingers (bases of the thumb)
4. 4x the largest circle inscribed in the lower part of the fingers (bases of the thumb)
5. 1x the largest circle inscribed in thumb
6. 5x area of fingers
7. 5x circumference of the fingers
8. 4x distance between the lowest point of the thumb and the centers of the bases of other fingers.
9. 4x the sides of the rectangle around the middle of the palm (red lines in the figure)

4 The Way of Comparison of the Features

The input data for the application determining palm features is the vector of 39 numbers. These values are not normalized. Their scope depends on the character of the particular feature. Therefore, the features cannot be compared to one another, as each of them has completely different scope of the values. To sort that problem out, the feature vector should undergo normalization [3].

4.1 Normalization

The normalization process is done separately for each feature. Firstly, the minimal and maximal values of each feature are found. Then, the value of each feature is transferred into the scope 0,0 - 1,0. For instance, the value of the certain feature in a sample is 15. The minimal value is 10 and the maximal is 20. In this case, the new value of the feature is 0,5.

After the normalization each feature is in the scope 0,0 - 1,0, so the all features have the same weight.

4.2 The Nearest Neighbour Method

The nearest neighbour method was chosen to comparing the samples. In consists in measuring the distance between the points represented as the feature vector in n-dimensional space and choosing the nearest points. In our case, the space has 39 dimensions.

Each individual inserted into the system is provided with some samples of his palm. That set of samples is so called class. As the points in such group lie very close to one another, we can assume that it is highly probable, that the currently examined point and the point nearest to it belong to the same class.

Below we provide the trivial example of searching for the nearest neighbour. In this case there are two classes of points 'A' and 'B'. Additionally, the space is two-dimensional.

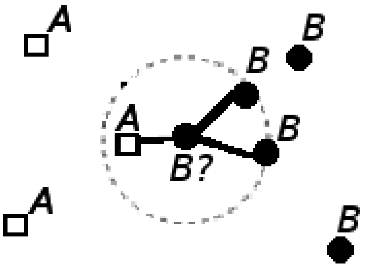


Fig. 3. Example of searching for the nearest neighbour

4.3 Metric

The method of measurement of the distances is also crucial. One can avail of various metrices. We have chosen the Euclidean metric. It is very popular, efficient and easy to imagine. Euclidean metric can be represented with the following formula:

$$d_2(x,y) = \sqrt{\sum_{i=1}^{N}(x_i - y_i)^2} \quad \text{for } x,y \in \mathrm{I\!R}^N \tag{1}$$

4.4 Weights

In order to make the method more efficient we introduced the weight for each feature, too. As it is very difficult to choose the proper set of weights, it must be done automatically.

The value dispersion of each feature was considered the main factor to be taken account of while determining the weights. The more various the values were, the larger the weight was. Standard deviation is a very good indicator of dispersion and thus we took it for weights.

The standard deviation is represented with the following formula:

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2} \tag{2}$$

5 Processing of the Sample and Searching for Features

5.1 Preparing the Contour Points Vector

Output data - the vector embracing the points defining the contour of the palm (starting from the pixel lying one pixel-position higher than the bottom left point of the palm, clockwise). The last point of that vector is the bottom left point of the palm (Fig.4 A and B).

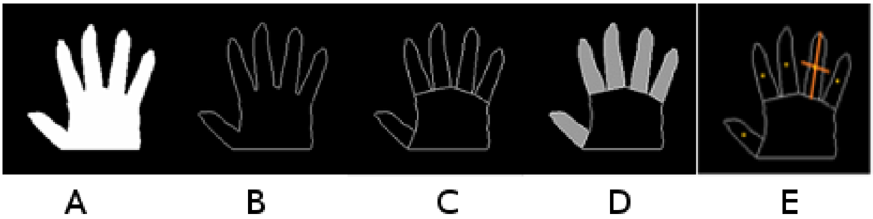


Fig. 4. A: Input data; B: Normalized palm image; C: Processed with the boundary-fill algorithm; D: Sum of coloured pixels; E: The middle of the line linking the middle of the nail

Creating the palm contour is a two-step process. First of all, all the white pixels are removed from the interior of the palm. Subsequently, the width of the remaining contour is one pixel and each pixel has only two neighbours. Then, the image is turned into the series of points. While searching the points the distances between these points and the referential point are also counted.

5.2 Finding Fingertips and the Spaces between the Fingers

The distance between the points of contour and the referential point is presented in the Fig.5. In that figure the maxima stand for the fingertips whereas the minima stand for the spaces between the fingers. According to the figure, minima and maxima can be flat. Each extremum is considered as an interval, because it cannot be unambiguously determined which pixel has the maximal value. In order to find these intervals one uses the find_extremums function.

5.3 Finding the Features Pertaining to the Fingers

If the key points are found, calculating the features is usually quite easy. In fact, it consists of simple mathematic operations for instance calculating the distance between the points. The function calc_fingers_feature calculates the features of the fingers.

However, the find_farrest_point function is worth mentioning because it gives back the nearest point in the given group to the point given in the second parameter. The middle of the base of the finger is chosen for that particular point.

5.4 Finding the Points of the Base of the Palm (bottom left and bottom right point of the palm)

The input data is the vector containing the contour points. It is intended to fill the areas of the bottom left and the bottom right point in the structure of the palm. The bottom left point of the palm is the last point of the contour. The bottom right point can be calculated assuming that it is the point of the same coordinate OY as the bottom left point and the value on the OX coordinate is as high as possible.

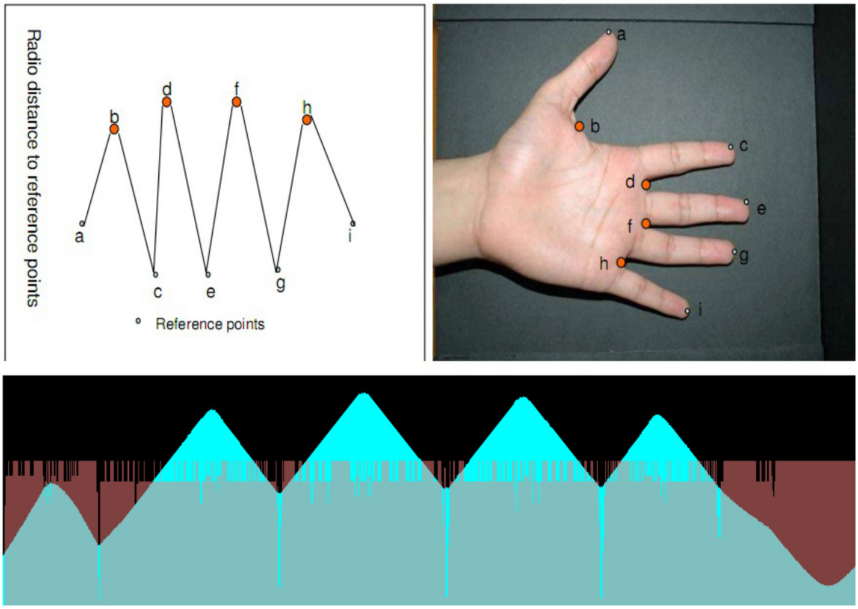


Fig. 5. The distance between the points of contour and the referential point

5.5 Calculating the Features of the Palm

The total area is equal to the number of the white pixels in the normalized image (Fig.4A).

The total palm legth is equal to the distance between the OY coordinate of the bottom left point of the palm and the minimal OY coordinate among the centres of the nails.

Other palm features are calculatedby means of thecalc_distance function.

5.6 Calculatinf the Area of the Fingers

In order to calculate the area of the fingers we avail of the algorithms pertaining to filling the boundary-areas. The input data is black image with the white palm contour, as in the Fig.4B.

The first step is to separate the fingers from the whole palm (by means of drawing lines linking the points between the fingers) (Fig.4C).

Then, each finger is processed with the boundary-fill algorithm. All the pixels surrounded be the contour (by the closed line made of one-colour pixels) are given the same colour)

The pixels given the new colour are counted. The area of the particular finger is equal to the sum of coloured pixels in its boundary-area (Fig.4D).

The fill-seed is the pixel form which we start filling the boundary-area. It must be placed within the area to fill. In our algorithm it is the point in the middle of the line linking the middle of the nail and the middle of the base of the finger (Fig.4E).

Other pixels are found thanks to the fact, that each pixel can be reached only from other pixels lying in four directions (up, down, left, right).

The above-mentioned algorithm is one of the seed fill algorithms. The fill seed is placed inside the area to fill and then it is propagated (sowed) in four directions. If the seed meets the fertile ground it fills it with colour and tries to propagate in four directions. The entire area is filled with the particular colour this way. The seed finds the fertile ground only if the pixel to propagate has different colour than the contour colour.

While sowing the number of sown seeds for the particular areas (fingers) is counted. The achieved number is equal to the area of the particular fingers.

5.7 Finding Circles Inscribed into Fingers

Finding circles inscribed into fingers is similar to inscribing the circle into any boundary-area for instance square, trapezoid or pentagon. To calculate the maximal circle inscribed into the boundary-area we used the Euclidean distance transform algorithm which was described in details and optimized by Pedro F. Felzenszwalb and Daniel P. Huttenlocher [6]. The principle of operation of the algorithm:

As the input data one should input the binary table (binary image);

1. the zero value stands for the boundary-area in which we want to inscribe the circle.
2. the one value stands for other pixels

As the output data we obtain the table in which all zeros were replaced with the squares of the maximal distance between the radius of the circle inscribed into the boundary-area and having the centre in a given point. Then we find the maximal value in the obtained table. In our case, it is 9. According to the algorithm, the middle of the largest circle inscribed into the given boundary-area is in the point having the biggest value (in our case, 9) The radius length is equal to the square root of the given number (sqrt (9) =3).[6]

6 The Carried Out Tests

6.1 The Method of Carrying Out Tests

Tests are carried out by means of the dedicated application written in c# language. It loads the test set consisting of the consecutive feature vectors and the names of related files.[7]

Because the NN method was chosen, the application takes down nearest neighbours of the sample counting from the one nearest in relation to the Euclidean metric up to the first. Then we seek for the first element of the same class as the examined sample (it means that both samples are the representation of the

Table 2. The exemplary excerpt from the output of the application

```
107: 107 22  202 332 227 199 171 41  185 171
107: 22  22  183 107 332 125 107 197 283 282
108: 108 70  65  243 65  243 44  557 536 362
108: 08  65  65  464 65  371 70  44  243 371
108: 649 509 243 405 620 537 443 141 260 509
109: 109 109 71  71  5   71  74  117 149 149
```

palm of the same user) When than element is found, we increase the value of the numerator related to the position in which it was adjusted.

The first number stands for the number of the currently examined class. The consecutive numbers stand for the classes of the nearest neighbours. As we see, in the second line the nearest neighbour for that sample has the class number 22 and the correct class was found in the fourth position.

6.2 The Results of the Tests

The tests were carried out of three sets of samples. The samples were chosen randomly. Each individual has 3 palm samples so the class size is 3, too. It is a very small value and it should be increased to 5 or 10 to improve the results.

Table 3. Tests results

$Modification$	$Test1(289samples)$	$Test2(199samples)$	$Test3(2225samples)$
1	90,31%	86,93%	65,66%
2	93,43%	90,95%	72,58%
3	94,46%	92,96%	77,08%
4	96,19%	92,96%	79,60%
5	96,89%	94,47%	81,84%
6	97,23%	94,97%	83,42%
7	97,92%	96,48%	84,54%
8	98,27%	96,98%	85,35%
9	98,62%	96,98%	86,25%
10	98,62%	96,98%	86,97%

7 Conclusions

According to the tests, the described method is a safe and effective identification method.

The results of the test 3 are not good. It is because the number of palm samples per each individual was very small. Increasing that number is expected to improve the results. In the test 1 the total number of samples was one less order smaller, which improved the results greatly.

The notation of results may seem dubious. However, the notation 3 - 77,08% means that there is 77% probability that there is a sample of the same class as the examined one among the three nearest neighbours. That information is of some importance in relation to the systems aiding the decision whether to grant user the access to the room, content or system or not.

References

1. Yoruk, E., Dutagaci, H., Sankur, B.: Hand biometrics. Image and Vision Computing 24, 483–497 (2006)
2. Yoruk, E., Dutagaci, H., Sankur, B.: Comparative analysis of global hand apperance-based person recognition. Journal of Electronic Imaging 17(1), 011018 (2008)
3. Bulatov, Y., Jambawalikar, S., Kumar, P., Sethia, S.: Hand recognition using geometric classifiers. In: Zhang, D., Jain, A.K. (eds.) ICBA 2004. LNCS, vol. 3072, pp. 753–759. Springer, Heidelberg (2004)
4. Yoruk, E., Konukoglu, E., Sankur, B., Darbon, J.: Shape-Based Hand Recognition. IEEE Transactions on Image Processing 15(7) (July 2006)
5. Jiang, X., Xu, W.: Contactless hand recognition, `http://www.cs.cmu.edu`
6. Jiang, X., Xu, W., Sweeney, L., Li, Y., Gross, R., Yurovsky, D.: New directions in contact free hand recognition. In: IEEE International Conference on Image Processing, ICIP 2007, vol. 2, pp. 389–392 (2007)
7. Felzenszwalb, P.F., Huttenlocher, D.P.: Distance Transforms of Sampled Functions. Cornell Computing and Information Science (2004)
8. Bogazici University Signal and Image Processing Laboratory, `http://www.busim.ee.boun.edu.tr`

User Authentication for Mobile Devices

Marcin Rogowski[1], Khalid Saeed[2], Mariusz Rybnik[3],
Marek Tabedzki[4], and Marcin Adamski[4]

[1] King Abdullah University of Science and Technology,
Thuwal, Kingdom of Saudi Arabia
marcin.rogowski@kaust.edu.sa
[2] Faculty of Physics and Applied Computer Science,
AGH University of Science and Technology, Cracow, Poland
saeed@agh.edu.pl
[3] University of Bialystok, Bialystok, Poland
mariuszrybnik@wp.pl
[4] Faculty of Computer Science,
Bialystok University of Technology, Bialystok, Poland
{m.tabedzki,m.adamski}@pb.edu.pl

Abstract. The paper is intended as a short review of user authentication problem for mobile devices. The emphasis is put on smartphones and tablets, that nowadays are very similar to miniaturized personal computers with much more sensors of various origin. The sensors are described with remarks on their usefulness for user authentication. Deficiencies of traditional user authentication methods based on knowledge are pointed out and the need for new – more secure but also comfortable – user authentication mechanisms is reasoned. Preliminary user authentication systems employing biometric features are discussed and hence the generally unused potential of biometrics for mobile devices is demonstrated.

Keywords: user authentication, biometrics, mobile devices, touchscreen.

1 Introduction

Mobile devices are constantly getting more popular and resourceful.

Nowadays they serve purposes of: personal information managing, various modes of communication, web browsing, documents editing, media presentation, etc., therefore largely replacing desktop computers. They contain large amounts of personal data and passwords. In 2005, even before the smartphone revolution had started, it was estimated that over 80 percent of new critical data is stored on mobile devices [1]. Increasingly, especially in post-2007 era, very similar services and applications are used on mobile devices and personal computers. Whilst security barriers were implemented on PC's for many years, it is a relatively new trend on the mobile devices. The need of securing aforementioned hundreds of millions of devices is obvious, with unauthorized access being the most common threat. The threat is countered by *user authentication*, being also the most

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 47–58, 2013.

important user-dependent issue (hardware and operating system security being rather a matter of device developer than user).

The mobility results in high risk in comparison to desktop computers. Obviously, the chance of an attempted unauthorized access is higher when the device is carried and used in public. Few methods of securing mobile device do exist, PIN (a kind of password) being the most common. The methods mainly secure devices from unauthorized access when mobile device is unattended by the legitimate user.

Certainly one cause of the deficiencies of user authentication for mobile devices is the expected access time. In case of PCs, usually the user authenticates and then uses the machine for an extended time, so she is more likely to accept more sophisticated and cumbersome security check. For mobile devices, usually the user accesses the device often and for short periods of time, what makes a trustworthy user authentication less desirable, assuming it requires some effort. The time required to authenticate should be taken into account and – for the comfort and cooperation of the user – be minimized.

The paper is organized as follows: section 2 presents the data sources available from mobile devices' sensors, section 3 discuses user authentication methods. Due to space limitation, the paper shows only the most striking-out approaches, that are especially interesting taking into account mobile devices functions and specifics. Finally the last section summarizes the paper and gives further predictions on the mobile market development.

2 Sensors of Mobile Devices

Mobile devices are currently equipped with many sensors that provide much more data than traditional communication channels of desktop computer – keyboard, mouse and screen. The key difference in the interaction with mobile devices and desktop computers is the interface. We rarely see a physical keyboard and almost never see a mouse attached to a mobile device. Most of the time touchscreens are used to interact with the device and replace both mouse and keyboard, with virtual keyboard displayed on the screen and typed using the display embedded in the device. This is the most important technology advance, that combine means of outputting visual data with inputting both: discrete data (characters and GUI interaction) and spacial data (touching, swiping, gestures at the specific coordinates).

Touchscreen provides more features than traditional interaction with a desktop computer. Physical keyboard is replaced by a virtual one and in addition to timing of keystrokes, also the size of each finger pressing a screen can be measured, pressure is estimated and the exact position of the finger pressing a key in relation to the position of this particular key pressed can be analyzed. In contrast to a PC, mobile devices – whether tablets or phones – produced these days, also have a plethora of other sensors most often including: an accelerometer, a gyroscope, a proximity sensor, an ambient light sensor, two microphones, one or two cameras and sometimes even a magnetometer. Some of these sensors obviously can provide more information about the user and his behavior and help differentiate between users. Table 1 presents the most important sensors, along with a quick remarks on their usefulness for user authorization.

Table 1. Sensors of mobile devices

Sensor	**Remarks**
Physical keyboard	usually no pressure control and no dwell registering
Touchscreen	exact coordinates and ”touch size” may be used as information
Microphones	usually two microphones, one for user voice and the other for noise-canceling
Camera	High quality image (8 Mpix and more)
Video camera	High quality video (for example HD quality)
Accelerometers	usable in gait dynamics [2] or in multi-biometric authentication
Gyroscope	not directly applicable to user authentication
Proximity sensor	usually detects only presence of nearest object (binary information)
Ambient light sensor	environment lightness, not directly applicable to user authentication
GPS	not applicable to user authentication
Compass	not applicable to user authentication

What is important, most of the data coming from the aforementioned sensors can be collected transparently to the user. That creates two possibilities: either to use it alongside the traditional methods discussed before (that is to strengthen the password rather than replace it) or to continuously authenticate the user. In the second approach, data is collected and analyzed all the time as the user ordinarily uses the device. With such *background authentication system*, even if the device is stolen in the authenticated state (for example grabbed from the users hand), an algorithm analyzing particular data may determine that the user using the device is no longer the legitimate one and may block the device.

3 User Authentication for Mobile Devices

Most of the user authentication methods may be adopted for mobile devices. Also, due to mainly instantaneous access time expected by the user, they do not seem to work very well because they are used in less secure versions.

Few of available sensors are widely involved to authenticate users of mobile devices. Mostly conventional security precautions are used, even when some sensors can enhance the existing methods and provide raised level of security. Many of the discussed characteristics do not require user cooperation (or even their knowledge) and can be collected in the background. That allows creation of a system that will continuously authenticate the user, based on his actions, without harassing the legitimate user. The goal of such system would be to block access to the device immediately after detection of suspicious behaviour. This may not only prevent an unauthorized access to the device when it is stolen in authorized state, but would also make compromising or breaking the password using brute-force methods virtually impossible.

Commonly PCs are shared by more than one person, contrastingly, mobile devices are almost never shared. This fact encourages to use biometric features for user authentication. It should also be noted that as any password, a biometric password can sometimes be lost. Certainly unpleasant situations as losing ones physical feature like a fingerprint might happen and despite being unlikely, some backup method of authentication should be thought of.

Following sections will discuss different user authentication methods, grouped into knowledge based methods and biometric methods, as token-based methods for mobile devices are virtually non-existent.

3.1 Knowledge Based Methods

Knowledge based methods are based on exclusive user knowledge of some sort. Passwords, PINs, pattern locks and graphic passwords are based on user memory, therefore unfortunately require some effort and are very prone to forgetting.

Password. The most popular desktop securing method – a password – is also ported in the same form to mobile devices. This however, is not as acceptable by the users as for desktop computers. A mobile device user expects almost instant access, and it is unlikely, that the user will voluntarily sacrifice convenience for security. Many will easily agree for a minor discomfort of a few seconds of delay, but it is unlikely that they would agree to a password as difficult as the passwords we use nowadays on desktop computers.

There are some publications describing what characterizes a good password – e.g. *complexity*, *uniqueness* and *secrecy* rules proposed in [3]. Unfortunately users most likely compromise one or more of these rules for their comfort. On a mobile device, designed to be comfortable, users are even most likely to jeopardize the *complexity* rule.

PIN – Personal Identification Number is a special case of a password, known well from ATM machines. In banking it dates back to 1966 when James Goodfellow patented a PIN derivation scheme [4]. Despite contemporary standards requiring its length to be 4–12 digits, most often used numbers are still of the length of 4 digits only. This carries a clear disadvantage as it can be easily calculated – there are only 10 000 possible PINs so the level of security proposed does not seem to be high. In banking, it is usually coupled with blocking the card after three failed attempts and hence some level of security is ensured – probability of guessing a PIN within three attempts is only 0.06%. It also still remains relatively comfortable for the legitimate user.

PIN was also adopted by smartphone makers as a security measure. The length is no longer fixed to 4 with the newest Android system specifying the length of a PIN to be between 4 and 17 characters. Apple still uses the length of 4 for iPhone and iPad but Simple passcode mode may be turned off changing a PIN into a password, as discussed before. Regrettably, there should be no illusion that mobile users will actually use a long number. Most likely the biggest group will use 4 digits and many of those will be birth dates, patterns on keyboard or common passwords like 1234. Recent study has clearly shown these tendencies [5]. After analyzing 204 508 PINs from iPhones it turned out that 46 of the possible combinations were not even used and 4.3% of all the PINs were 1234. Another problem with the use of a PIN on a mobile device is that it would not be acceptable, as in banking, to cause major discomfort for a user after he or she inputs a wrong number three times. The number can be entered numerous times during a day on a mobile phone, so mistakes, even many times in a row may happen. The solution used now by Android developers is locking the device for a predetermined period of time so it protects the device from spontaneous brute-force attacks when it is unattended for a few minutes. However, this simple lock is not going to protect the device if it can be accessed for a longer period of time. Apple included an option to erase all the data on the phone after the PIN is entered incorrectly for 10 times, but this solution may frustrate users, especially if they have kids or if a user with malicious intent gets access to their device even for a few minutes.

Another problem is that the PIN encourages users to pick the numbers that create a pattern on a keyboard and makes the password predictable. In general, a PIN can be considered as a special case of password that carries all its deficiencies.

Pattern Locks. Another form of a security measure that can be used on contemporary touchscreen devices is a pattern lock. Instead of a PIN, the user is required to connect dots in a predefined order. Surely, that kind of pattern may be easier to memorize and does not cause major discomfort for the user.

The researchers from the University of Pennsylvania explored deficiencies of pattern locks in [6]. As calculated there, in general there are about 1 million possible patterns using 9 points. However, if constraints of implementations are taken into account – for Android, if a point lays between two selected points, it also has to be selected – there are only 389 112 combinations left. If the users comfort is also taken into account, it turns out, that more than a half of these combinations contain patterns that are not comfortable for the user to enter. As the users pick convenient patterns, it is not likely that they will pick one of these special cases and as a result search space can be narrowed down to 158410 likely combinations.

The discussed paper exposes another drawback of using pattern locks on touchscreen devices: oily residue left behind by finger. Using photo-editing software they were able to successfully expose the trace of a finger and in the report they state the method worked in 92% of the cases.

A solution to the problem was proposed by *Whisper Systems* with their product *Whisper Core* offering smudge-resistant screen unlock patterns [7]. The solution proposed is to force the user to replace pattern-smudge with another smudge – namely wiping the entire screen with a finger. It is some solution to the problem but the problem with wider adaptation of the method may be again, the comfort of the user. This method adds at least two additional strokes of a thumb so for some users it may double the effort of unlocking a screen.

Another risk of using a pattern lock is that it is quite easy to be compromised. It is usually a characteristic pattern and it can be easily observed and memorized.

Graphic Passwords. Replacing PIN with a graphic password is another idea of using the nature of touchscreens to improve upon the security of the PIN. One of the ideas was introduced by J. Citty in [8]. Instead of using a 4-digit sequence, 16 images partitioned into four parts are used. To verify the user, they are required to select the correct parts of the correct images in the right order. This change increases the number of possible password combinations from 10^4 for a 4-digit PIN to over 10^7. The sequence entry time is initially about two times that of a PIN, but as users practise, it gets close to it.

3.2 Biometrics

When authenticating with a password, a PIN or a pattern, the authentication is a binary problem – either the predetermined pattern matches or it does not. When using biometric features, the problem is more complex. Whether it is a fingerprint, a face image or any other biometric feature, it is very unlikely that the obtained feature will exactly match the one collected whilst enrolling the user. There is always a tradeoff between False Acceptance Rate and False Rejection Rate. Minimizing FAR makes the system more secure but at the same time causes discomfort to the users by making FRR higher and vice versa.

Face Recognition. Face recognition is a very well known biometric feature analysis. Recently with development of hardware and processing power, it has been used with success in many environments [9].

Considering mobile devices applications at large: *Face Unlock* is a new feature added to Android system in late 2011. It is also the first biometric feature used there. It has to be admitted that this implementation of unlocking using face image works instantaneously. Unluckily, for this level of comfort, a price had to be paid.

Clearly, to achieve the speed and low ”insult rate” as FRR may be called, developers of Android had to sacrifice FAR therefore making *Face Unlock* less secure. It is even incorporated as a standard warning to the user that *Face Unlock* is less secure than other methods and persons looking similar will be able to get access to the device. What is not mentioned is that also people having a picture of a person that looks similar to the user will be able to get access to the system as a simple camera will not be able to tell the difference between the actual face and the picture of the face being in front of it.

There are also a few different problems with *Face Unlock*. The environment lighting plays a big role and the method will simply not work when it is too dark or too bright. Another fact is that the position of the device when taking a picture is important, so it will not be possible to use this method to unlock a phone or a tablet lying flat on the desk or to do it discreetly in public without positioning a device in front of ones face. There is also a requirement for the device, as it has to be equipped with a front-facing camera to make it comfortable to use.

Keystroke Dynamics. The users typing style was shown to be a feature differentiating fairly well between users. The idea (for personal computers) first came as early as 1975 when it was mentioned by R. J. Spillane [10] and has since then been modified and improved on multiple occasions with the results reported as high as 99-100% correct classification. In general, in almost all the works on keystroke dynamics the user is asked to type a particular password multiple times.

The same characteristics that were used on a physical keyboard may be usually used on the touchscreen keyboard. Many more properties may be registered including exact location of finger on the virtual button, the size of touching finger, the pressure, and the changes in position coming from both the accelerometer and the gyroscope. This gives a huge potential for keystroke dynamics to be used for user authentication.

The same approaches as in traditional keystroke dynamics can be used – a fixed-text authentication and a free-text authentication. In the case of the fixed-text authentication, the user will perform the verification procedure as usual – whether it is a pattern lock, PIN or a password and not only correctness of the combination would be evaluated but also how they did it. There are a number of features that can be extracted to help determine if a user is a legitimate one. For traditional keystroke dynamics, the *dwell* time of a single key press is universally used. Most commonly it is combined with the so called *flight* time – time from releasing a key to pressing a subsequent one in the password. These characteristics proved to be reliable and there are no significant improvements over these results if additional features are extracted. It has to be considered if the problem is exactly the same on a small touch screen – usually around 4 inches in diameter for a phone and around 10 inches for a tablet – as on a full-sized physical keyboard.

Some preliminary research was done a few years ago on mobile phones when touchscreens were not prevalent in the market. A device with a small, thumb-typed physical keyboard was used by [11] and some interesting observations were made. In contrast to the full-sized physical keyboard, in this case, the dwell time of any single key did not prove to be a reliable differentiating characteristic and the resulting error rate was near 50%. Using the flight time between pairs of keys resulted in a much better 12.2% error rate on a group of 50 participants. The results obtained point out that the fact that the depression time of a key is not a good discriminating feature may be caused by the use of thumb-typed keyboard.

Initial research done on a device with 4.3 inch touchscreen shows that dwell and the flight times both give an error rate higher than 10% if not combined with additional features. The results obtained combining different characteristics, for example the dwell time and the size of the finger, are very promising but as the area is relatively new, comprehensive results are not yet available. Another interesting approach was evaluated for the full-sized physical keyboard [12], but may be relevant to mobile devices: free-text keystroke dynamics authentication running in the background and monitoring the user behavior. The motivation for this approach is the risk that an impostor gets the access to the computer when the user is logged in and will continue working as if he was authorized. Because of the very nature of mobile devices, this risk is amplified and, if implemented well, the continuous authentication approach may prove to be very relevant and useful in the new area.

In the case of a virtual keyboard displayed on the touchscreen, there is also the possibility to extract exact coordinates where the keys were pressed, the size of the users fingers that touch the screen and sometimes the pressure put on the display. Initial research has shown the finger size to be a very promising differentiating feature while the pressure, on physical keyboards, was shown to be as effective as latency information [13]. A huge advantage was clear when the latency and the pressure information were combined – accuracy was significantly improved.

Gestures. Recently a study of biometric-rich gestures was conducted by a team led by Prof. Nasir Memon and published in [14]. The authors used tablets and asked users to perform particular gestures such as closing, opening, and various rotations, all in many different versions with different fingers fixed. The user-defined gesture was also allowed and in this case the movement of five fingertips while the user was signing his signature was recorded. In the verification phase, the input is compared to the stored template and based on the similarity, the user is authenticated or not. A Time Warping algorithm is used to compare the similarity of the two sequences. 34 users were asked to participate. From the predetermined methods, counter-clockwise rotation of all five fingers gave the best results with 7.21% Equal Error Rate, whilst the average was 10%. The researchers also evaluated the performance when two gestures were used. In this case the error rate was significantly lower and as low as 2.58% for clockwise rotation of five fingers followed by counter-clockwise rotation of four fingers with fixed thumb. This result indicates that different gestures make different characteristics of hand or the users interaction style stands out. The user-defined gesture was also very effective and the error rate was 2.88% in this case. It is important that most of the users said that the gestures are pleasant to use and a good number said they were *excited* to use them. 25 out of 29 users preferred this method over a text password and all 29 users thought it would be faster.

Touchscreen Dynamics versus Mouse Dynamics. Mouse dynamics is a method of evaluating the users specific style of moving a mouse. Different researchers analyzed different characteristics, often including deviation from the straight line, a user-specific ratio of dragging, clicking, average speed etc.

On a touchscreen device, the users finger takes the role of the mouse. Similarly as in physical/virtual keyboard comparison, using a touchscreen provides all the same information and a few features more. Not only X and Y coordinates in time can be recorded, but additionally there is also the information about the size and the pressure. What may be important for some concepts, like gesture recognition, the most modern touchscreens support multi-touch, i.e. more than one finger interacting with the screen at the same time.

One of the interesting and relevant approaches was introduced in 2005 by Hashia et al. [15]. The teams system asks users to connect some dots. Dots are shown one at a time and the user is simply required to move the mouse from one dot to another. The coordinates of the mouse are recorded every 50ms and the speed, the deviation from a straight line and the angle are calculated. In the verification phase, these values are compared to the ones recorded during the enrollment phase. The resulting error rate is 15%, which is not that bad considering how simple the statistical model is. It is quite easy to notice the similarities between this method and the pattern lock used in Android phones. Authenticating using a pattern lock is basically the same task – connecting dots, but only the order in which the dots are connected is evaluated. It is a very simple extension to also include some statistical information about the speed, the angle or the deviation from a straight line, as in the work discussed, to the same system. The change will be transparent to the user but may result in increasing the security level.

Hashia et al. also discuss the continuous authentication introduced before or as they call it – passive authentication. Instead of requiring the user to move the mouse over dots, whole regions of screen are treated as dots. They record the movements, and every two minutes analyze them and authenticate the user. No detailed performance characteristics are provided but an average time of 2 minutes for which the intruder was allowed to use the computer and 5 minutes after which the actual user was considered an intruder which suggests that False Rejection Rate may be alarmingly high. Passive authentication could be implemented on a mobile touchscreen as well. Slightly different approach was evaluated in [16]. Only the speed of the mouse between pre-determined points was analyzed and using minimal eigenvalues of a Toeplitz matrix for classification resulted in accuracy of almost 70%. On a touchscreen, data about pressure and size of the fingers could also be used, giving more discriminative features and the result would be most likely improved.

Voice Recognition. There are two approaches to voice (speaker) recognition – text-dependent and text-independent. Both could be used on a mobile device, especially a phone. If the voice analyzed in background during the conversation does not match the pattern stored beforehand, an additional authentication may be triggered. This method will protect users from the impostors that managed to come into possession of the primary password.

The speaker recognition due to some inconveniences instead to be used standalone may be used for multi-biometric systems discussed later. The performance of text-dependent speaker recognition systems is estimated to be similar to those of signature at error rate of about 2% [17].

Enhanced Pattern Lock. The authors in [18] proposed an approach to authenticate users on the basis of the correctness of the pattern they enter and the way they do it. This is a direct translation of keystroke dynamics enhanced password to a pattern lock. Similarly as in Memons work [14], a Dynamic Time Warping algorithm is used to compare the similarity between the reference and the current input. Here x-y coordinates, time, size, pressure and speed are collected. Based on the data set of 31 users, researchers have achieved 77% accuracy with 19% false rejection rate and 21% false acceptance rate. Modifying the threshold used for some of the users helped to further improve the accuracy. It also suggests, that machine learning approach can be evaluated for the same problem and it should be checked how its performance compares to simple thresholds based on the Dynamic Time Warping algorithm.

Gait Analysis. In [2] authors used the data from the mobile phone accelerometer to distinguish between users, basing on the way they walk (gait dynamics). The achieved equal error rate of 20.1% is higher than in other works on the subject, but only a standard mobile phone was used and not specialized equipment or video recording like in the other cases.

Multi-biometric Authorisation. Certainly the abundance of sensors providing various, also biometric data allows to create some kind of a system that will combine these inputs and authenticate the user based on the combination of features. One such system was proposed by the researchers from MIT in 2003 [19] using a personal digital assistant device with a resistive touchscreen but no phone option. Their system investigates the usage of two-way multi-biometric authentication combining the face image and the speaker recognition. The image of the face is taken using the embedded 640x480 resolution camera. For the speaker recognition a built-in microphone is used which registers users voice pattern when pronouncing a password consisting of three two-digit numbers. The results obtained are promising with the error rate below 1% when analyzing both the face and the voice.

As mentioned, the important factor is user comfort. Efficient enrollment is also important and it is unlikely that users will chose to replace their PINs and patterns with a multi-modal system which takes an average of 30 minutes to set up – MIT researchers required users to take 25 face images in different lighting settings and to recite 16 generated pass phrases.

4 Conclusions

Mobile devices are becoming ubiquitous but their specifics causes security issues. Security measures are frequently omitted by users for the sake of comfort. Unauthorised access or loss of mobile devices is costly, as increasing amount of personal data is stored on them. Biometric solutions are rare and mostly are only available as third party applications, therefore unknown or disregarded by large audition. *Face Unlock* introduced to Android in late 2011 is the first biometric security measure available in large scale.

Hardware sensors of mobile devices are numerous and much biometric data is available. Possible applications of the biometric data available on mobile devices were discussed. Some of the new methods already experimentally implemented are described, others pointed out as possible research directions. Relevant approaches on different hardware, like mouse dynamics, were reviewed to assess their usability on the new type of devices.

Hopefully, for the sake of security, new security measures employing biometric features – however maintaining the users comfort – will be widely adapted for mobile devices, eventually becoming largely available as an integral part of mobile operating systems.

Acknowledgement. This work was partially supported by AGH University of Science and Technology in Cracow, grant no. 11.11.220.01.

References

1. Allen, M.: A day in the life of mobile data. In: Mobile Security. British Computer Society (2005), `http://www.bcs.org/server.php?show=conWebDoc.2774` (accessed May 14, 2012)
2. Derawi, M.O., Nickel, C., Bours, P., Busch, C.: Unobtrusive User-Authentication on Mobile Phones Using Biometric Gait Recognition. In: Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), pp. 306–311 (2010), doi:10.1109/IIHMSP.2010.83
3. Burnett, M., Kleiman, D.: Perfect Passwords. Syngress, Rock-land (2005)
4. Ivan, A., Goodfellow, J.: Improvements in or relating to Customer-Operated Dispensing Systems. UK Patent #GB1197183 (1966), doi:10.1049/el:19650200
5. Bonneau, J., Preibusch, S., Anderson, R.: A birthday present every eleven wallets? The security of customer-chosen banking PINs. In: Keromytis, A.D. (ed.) FC 2012. LNCS, vol. 7397, pp. 25–40. Springer, Heidelberg (2012), `http://www.cl.cam.ac.uk/~jcb82/doc/BPA12-FC-banking_pin_security.pdf` (accessed May 14, 2012)
6. Aviv, A.J., Gibson, K., Mossop, E., Blaze, M., Smith, J.M.: Smudge Attacks on Smartphone Touch Screens. In: Workshop on Offensive Technology (2010), `http://static.usenix.org/event/woot10/tech/full_papers/Aviv.pdf` (accessed May 14, 2012)
7. Whisper Systems WhisperCore, `http://whispersys.com/screenlock.html` (accessed May 14, 2012)
8. Citty, J., Tapi, D.R.H.: Touch-screen authentication using partitioned images. Elon University Technical Report (2010), `http://facstaff.elon.edu/dhutchings/papers/citty2010tapi.pdf` (accessed May 15, 2012)
9. Tolba, A.S., El-baz, A.H., El-harby, A.A.: Face Recognition: A Literature Review. International Journal of Signal Processing 2(2), 88–103 (2006)
10. Spillane, R.: Keyboard Apparatus for Personal Identification. IBM Technical Disclosure Bulletin 17(3346) (1975), doi:10.1109/MSP.2004.89
11. Karatzouni, S., Clarke, N.: Keystroke Analysis for Thumb-based Keyboards on Mobile Devices. In: Venter, H., Eloff, M., Labuschagne, L., von Eloff, J., Solms, R. (eds.) SEC 2007. IFIP, vol. 232, pp. 253–263. Springer, Boston (2007)
12. Rybnik, M., Tabędzki, M., Saeed, K.: A Keystroke Dynamics Based System for User Identification. In: Proceedings of the 7th International Conference on Computer Information Systems and Industrial Management Applications: CISIM 2008, pp. 225–230. IEEE Computer Society (2008), doi:10.1109/CISIM.2008.8

13. Loy, C.C., Lim, C.P., Lai, W.K.: Pressure-based Typing Biometrics User Authentication using the Fuzzy ARTMAP. In: Neural Network International Conference on Neural Information Processing (2005), http://www.eecs.qmul.ac.uk/~ccloy/files/iconip_2005.pdf (accessed May 14, 2012)
14. Sae-Bae, N., Ahmed, K., Isbister, K., Memon, N.: Biometric-rich gestures: a novel approach to authentication on multi-touch devices. In: Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems (CHI 2012), pp. 977–986. ACM, New York (2012), doi:10.1145/2207676.2208543
15. Hashia, S., Pollet, C., Stamp, M., Hall, M.Q.: On Using Mouse Movements As a Biometric. In: Proceedings of the International Conference on Computer Science and its Applications (2005), http://www.cs.sjsu.edu/faculty/pollett/papers/shivanipaper.pdf (accessed May 15, 2012)
16. Tabędzki, M., Saeed, K.: New Method to Test Mouse Movement Dynamics for Human Identification. In: KBIB 2005 Conference, Tom I, Computer Science Telemedicine Systems, pp. 467–472. Czestochowa Technical University Press, Poland (2005) (in Polish), http://home.agh.edu.pl/~saeed/arts/2005%20KBIB.pdf (accessed May 15, 2012)
17. Myers, L.: An Exploration of Voice Biometrics. GSEC Practical Assignment (2004), http://www.sans.org/reading_room/whitepapers/authentication/exploration-voice-biometrics_1436 (accessed May 15, 2012)
18. De Luca, A., Hang, A., Brudy, F., Lindner, C., Hussmann, H.: Touch me once and I know it's you!: implicit authentication based on touch screen patterns. In: Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems (CHI 2012), pp. 987–996. ACM, New York (2012), doi:10.1145/2207676.2208544
19. Hazen, T.J., Weinstein, E., Park, A.: Towards robust person recognition on handheld devices using face and speaker identification technologies. In: Proceedings of the 5th International Conference on Multimodal Interfaces (ICMI 2003), pp. 289–292. ACM, New York (2003), doi:10.1145/958432.958485

Modified kNN Algorithm for Improved Recognition Accuracy of Biometrics System Based on Gait

Marcin Derlatka

Bialystok University of Technology,
Wiejska Street 45C, 15-351 Bialystok, Poland
m.derlatka@pb.edu.pl

Abstract. The k-nearest neighbors classifier is one of the most frequently used. It has several interesting properties, though it cannot be used in utilitarian biometric systems. This paper proposes the modification of kNN algorithm which ensures correct work even in the case of an attempt at access to the system by unauthorized people. Work of the algorithm was tested on data presenting ground reaction forces (GRF), generated during human's walk, obtained from measurements carried out on over 140 people. Differences between particular strides were determined with the Dynamic Time Warping (DTW). The recognition precision obtained was as high as 98% of biometric samples.

Keywords: k-nearest neighbors, human gait recognition, ground reaction forces.

1 Introduction

Biometrics understood as a science of recognizing a human on the basis of his/her physical and/or behavioral features becomes increasingly popular and increasingly courageously enters our life. The number of devices (e.g. computers) and institutions (e.g. banks) where a particular type of biometrics is used is still increasing. Every biometric system, irrespective of the feature this system is based on, includes attribution of a 'biometric signature' provided by a person applying for access to resources, to one of the classes registered in the data base. Obviously, if the provided biometric signature is not similar enough the class pattern stored in the data base, there should appear information on a denial of access to the resources.

Among the most intuitive classifiers are the classifiers in which similarity is defined as the inverse of the distance in the feature space, e.g. the kNN classifier. In this case the input pattern is attributed to the class which includes most of its nearest neighbors. Intuitiveness, quite good generalizing properties and easiness of implementation result in the fact that the biometric systems working the basis of the kNN algorithm are frequent in literature [9], [12]. Unfortunately, this classifier, despite its several modifications enhancing both the properties

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 59–66, 2013.

and the quality of classification [5] [12], has some features that prevent its direct application in biometric systems. They are:

- attributing a given point of the space of features to one of the classes even when the distance, though minimal, is as long as we should talk about the lack of similarity to all patterns; in this situation the desired answer of the classifier is the information of unrecognizing the given person;
- while making decisions taking into consideration the constant number k of the nearest neighbors, even if part of them are relatively far from the point under classification.

These features, occurring also among other classifiers, lead to the fact that papers connected with biometrics very often assume that we deal with a closed group of people and we are not endangered with an attempt at getting to the resources by the people from outside of this group [11]. Certainly, such an approach limits the potential application field of a given system.

One of more interesting behavioral biometrics is human gait. Gait is a very complex human activity. It is a symmetrical and repetitive phenomenon in its normal form. The movement of each limb is partitioned for the support phase - when the foot is in contact with the ground and the swing phase when the foot is lifted and moved forward. The human gait is developed during child growth. It is assumed that the human gait pattern is evaluated till the child is seven years old and next stay almost unchanged till his death. The human gait is the result of synergistic activity of: bones, muscles and nervous systems. The cooperation of those three systems makes the human movement unique for every person.

Human recognizing using gait is not a new issue. It was Cutting and Kozlowski [2] who already conducted researches which demonstrated that a human has a skill of recognizing the people he knows from a long distance by their way of moving, even if they wear clothes different than usual and have changed their hairstyle. Gafurov [4] distinguish the recent methods in human gait recognizing depending on the signals registered. They are methods based on the data obtained from:

- video;
- floor sensor;
- portable sensors.

In case of video based methods the picture registered is usually converted frame by frame into silhouette sequences. Subsequently, depending on the applied methods there occurs an attempt at reading of selected parameter's of the human's gait and classification of a person [1],[6]. Advantages of these methods are undoubtedly a possibility of free motion of the person under examination, identification of many people simultaneously and these people's approval of the presence of cameras in buildings. Problems with recognizing people in the systems based on video cameras are generated by the examined person's change of clothes, a possibility of covering the examined person by objects or other people, changes in lighting as well as the sensitivity of certain parameters sought for to the position angle of the object in relation to the camera[6]. These disadvantages do not occur in the group of methods based on the measurement of the interaction of the ground with the lower limbs of the person under examination (floor sensor based). Here the examined person must walk along an appropriately-prepared measurement path equipped with force plates [3] or a special floor with a network of photo interrupter sensors [11]. In the last method mentioned above the person under examination fully cooperates during the time of measurement. He/she is provided with measurement equipment, such as: accelerators [4], or opto-reflective markers [7].

The main objective of this paper is to demonstrate a modified kNN algorithm which can be applied in biometric systems. The work of the algorithm has been tested in author own research on human gait.

2 Ground Reaction Forces

In the biomechanical approach, the ground reaction force (GRF) is the force which is acting on the body as a response to its weight and inertia during the contact of the human plantar with the surface. The all three components of GRF are used in the presented work. They are: anterior/posterior F_x, vertical F_y and medial/lateral F_z components of GRF. The common profiles of the GRF components are presented in Fig. 1 (a-c).

The anterior/posterior component has two main phases. The value of F_x is negative in the first phase. It is a result of the deceleration of the investigated lower limb, in this case the force direction is opposite in direction of walking. The minimum of the deceleration phase is most often reached a moment before

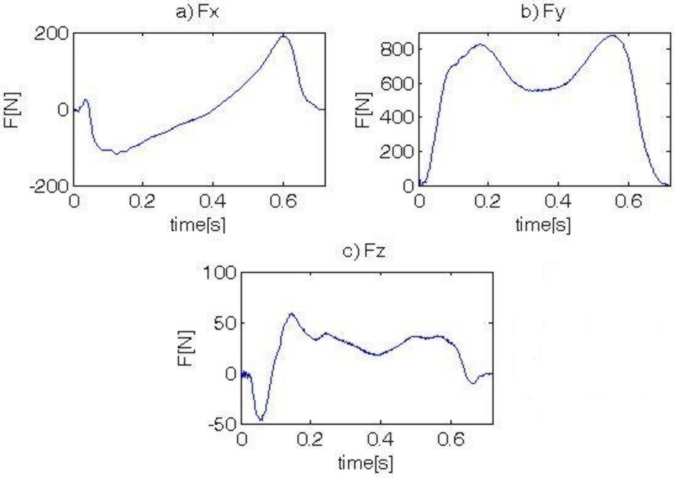


Fig. 1. Components of GRF in: a) anterior/posterior, b) vertical, c) medial/lateral direction, during the support phase of the left lower limb

the maximum of the limb-loading phase in the vertical component of GRF. The value of F_x is positive in the second phase, respectively. The maximum of the acceleration phase is reached when the toe-off phase starts. There are three extremes in Fig. 1b. They correspond to: the maximum of the limb-loading phase, the minimum of the limb-unloading phase and the maximum of the propulsion phase (a moment before the toe off). It is not difficult to point to the same extremes for the medial/lateral component of GRF as for the vertical GRF.

3 The Modified k-NN Classification Algorithm

The k-nearest neighbors algorithm was modified for the needs of biometrics in such a way that it enables to make a decision depending on the degree of similarity between the case considered and the prototype patterns connected with a concrete user. To determine a similarity between the patterns author used a well-known DTW (Dynamic Time Warping) transform, which reduces the difference between time courses to the cost of adjusting one time series to another. The distance between n and m patterns has been calculated according to following formula:

$$D(n, m) = \sum_{c=1}^{6} D_c \tag{1}$$

where: D_c is the DTW distance between the c components of GRF for patterns n and m.

Values of similarity thresholds were defined for each user separately taking into consideration different dispersion of prototypes in the feature space for particular users. Consequently, for i-th user an average distance ρ_i between particular prototypes was calculated. The distance limit connected with a concrete threshold was calculated according to formula:

$$\rho_{\theta i} = (1.5 - \theta)\rho_i \tag{2}$$

where: θ threshold; θ=$\{0.1, 0.2, \ldots, 0.9\}$ and 1.5 is constant chosen arbitrary. It is worth noting that adopting such a definition means that for the threshold $\theta = 0.5$ all the points whose distance from the prototypes is longer than the value of average ρ_i of a given person will be treated as too dissimilar to a given class and will not be taken into account while making decisions (see point 4 of the following algorithm).

The modified k-NN classification algorithm was as follows:

1. Determine distance D of the biometric pattern under examination from all prototype patterns in the data base.
2. Select k prototype patterns whose distances D to the pattern under examination are the shortest.
3. By majority vote determine the ID of the user in the data base. If two or more users are equally numerously represented among the patterns selected in point 2 (or the remaining after rejection in point 4) - select the one whose average distance from the selected pattern is the shortest.

4. Reject k' prototypes, for which distance D is longer than $\rho_{\theta i}$ for the given threshold θ of the user whose ID was selected in point 3.
5. Check in compliance with the procedure in point 3, whether ID for K=k-k' of prototypes remained unchanged. If so, we finish the classification assigning the examined biometric pattern to the ID class. If not, return to point 3.
6. In the case when k=k' (K=0) we recognize that a given biometric pattern cannot be classified in any of the classes at the assumed threshold θ.

Such a construction of the kNN classifier enables proper classification even if near the considered i-th class point in the input space there are more prototypes appertaining to competitive (j-th) class. This is possible on condition that they are more distant from the biometric signature under consideration than the limit threshold value for j-th class and the adopted threshold θ.

4 Research Material and the Procedure of the Experiment

142 people (62 women and 80 men) took part in the research conducted in the Biaystok University of Technology. The participants in the research were 21.20 ± 1.14 years old, body weight 74.95 ± 17.0 kg and body height 174.26± 8.81cm. During the examinations the person walked at a free speed in his/her own sport shoes on the measurement path, in which 2 Kistler's force plates were hidden, working at the frequency of 1 kHz. The volunteers performed several walks (14-20), as a result of which over 2500 strides were registered. 1056 patterns obtained from 132 people (57 man and 75 women) made a learning set (prototype points). 1500 patterns obtained from all the people under examination were treated as a testing set. The people under examination were divided into two groups: the so-called users or genuine of the system (132 people), who, as assumed, were to receive access to the system and the so-called impostors (10 people, 176 strides), who were represented only in the testing set and use to test the resistance of the biometric system to attempts at cheating (attempts at access to the data by unauthorized people).

For the obtained time courses we determined distances between all prototypes and all patterns tested in accordance with formula (1). In this paper we conducted a classification in accordance with the modified kNN algorithm proposed before, whereas for comparison we employed the classical version of the k-nearest neighbors. In both cases we assumed k=5. In the case of the classical kNN algorithm, the situation where among k-nearest neighbors was equally numerous representation of 2 or more classes, was treated as FRR error.

Table 1. Female and male characteristics

	No. of subjects	Age	Body weight	Body height
Female	62	21.15±1.10	64.47±11.68	166.83±5.53
Male	80	21.24±1.17	83.08±16.04	180.03±6.15

5 Results and Disscusion

Percentage of wrong classification for the classical kNN algorithm amounted FRR 2.19% (29 cycles) and FAR 4.00% (53 strides wrongly classified). So good an output results from a very big difference in the distances determined with DTW. The distances between the prototypes and the courses appertaining to the testing set performed by the same person were usually lower by a rank than in the case of the distances to these prototype points of strides of other people. The analysis of the situations in which an FRR error occurred in the classical kNN demonstrated that in the case of 19 strides it is possible to obtain a correct classification provided the most popular method of solving 'draws' were applied. It is enough, while making decisions, to adopt a criterion of average distance of the point under consideration to the prototypes of particular classes. Obviously, the remaining 10 strides will be classified incorrectly, which will raise the value of the FAR error up to 4.76%. It is important to note that the value of FAR error for the classical kNN is relatively low. Especially if we compare it to the results of other authors' works basing on signals of the same or similar types. Thus, for example, in the paper [9] (selected features of GRF and kNN profiles) the error amounted 7% on the group of 15 people, and in [11] (UbiFloorII and MLP): 1% and 10 people respectively. Slightly better results than in this paper were obtained in [10], where for foot pressure patterns, in the best cases, the error was 0.6% to 7.1% (104 people; 520 strides) and in [8], where for GRF and so-called wavelet packet decomposition scheme the values of the error obtained were started from 0.5% (40 people). It is important to underscore that this paper is based on much richer research material, which obviously affects the obtained results. In the case of the proposed modified kNN algorithm the value of FAR errors for the users independently from the adopted threshold are considerably lower than for the classical version of kNN. The analysis of the situations where the classical kNN generated an error allowed for demonstrating in which cases the proposed kNN algorithm works better. One of such cases occurs when a correct class has less numerous representation than a competitive class among the selected k prototypes but the distances of some of these points in the competitive class are longer than acceptable (D>$\rho_{\theta i}$). This causes rejecting a sufficient number of prototypes of the competitive class and the correct classification. In this case the classical kNN algorithm gives a wrong answer. Another possibility may occur when a correct and competitive class (classes) have an equally numerous representation among the prototypes of the shortest distance but the average distance of these points of the correct class is longer than that of the competitive one. In the situation where distances of a higher number of cases in the competitive class are longer than the acceptable kNN algorithm provides a wrong classification, whereas the proposed one correct. Sometimes there occurs a situation where the proposed algorithm for the adopted threshold rejects all the prototypes of both the correct class and the competitive one. In this situation we obtain a FRR error. Generally speaking, it is important to state that the kNN algorithm modified for the needs of biometrics is quite conservative and such a situation occurs relatively frequently. The classifier selects the answer 'I don't know' forcing the user to repeat the trial. The value

Table 2. The results of the identification of the users and impostors using modified kNN

security level (thereshold)	modified kNN		
	users		impostors
	FRR	FAR	FAR
0.1	1.06%	3.25%	44.83%
0.2	1.44%	3.17%	41.38%
0.3	1.81%	2.87%	35.06%
0.4	2.27%	2.79%	29.89%
0.5	3.47%	2.57%	24.14%
0.6	4.53%	2.34%	16.67%
0.7	5.82%	1.96%	9.19%
0.8	8.23%	1.66%	7.47%
0.9	11.85%	1.06%	4.02%

of threshold $\rho_{\theta i}$ for particular people plays a very important role. A low value of the threshold increases the value of FRR error. Along with the increase in value of the threshold the modified kNN approaches in its work a classical kNN, which reduces FRR at the cost of higher FAR. The choice of the threshold value is a compromise between the desired flexibility of the system and its resistance. Thus, taking into consideration a threshold 0.7 we force a situation where it is slightly more often than every 20 person has to repeat an attempt at access to the system. However, this value secures only every 50th trial finishing with treating user X as user Y (FAR error 1.96%: 26 wrongly classified strides). It is important to note that in this case the number of wrong classifications is more than twice lower than in the case of the classical kNN algorithm. A certain anxiety is triggered by the results of the FAR error for intruders. At least 4% (for threshold 0.7 - 9%) the acceptance of the unauthorized people is a result definitely unsatisfactory. In fact, the proposed classifier even in this case works better than a classical kNN (where the value of FAR error would be 100%); however, it indicates a need for seeking for further algorithm modifications, which will protect the biometric system from access of unauthorized people without a dramatic rise of FRR error of users.

6 Conclusion

The results obtained, basing on measurements of ground reaction forces generated during a walk conducted on an enormous, in comparison to other authors researches, data base independently from the applied classifier demonstrate a high potential of human gait as a biometric. The presented modification of the kNN algorithm for the needs of biometrics, to a considerable degree, fulfills its task. It allows for a correct classification of also those cases which have no chances with the classical version of kNN. Also a weak point of the proposed method was demonstrated: a relatively low resistance to attempts at access to resources on the part of unauthorized people. The whole encourages to continue works in this direction, which, for example more rigorous criteria of threshold $\rho_{\theta i}$ selection, should make a good method of classification still better.

References

1. Balista, J.A., Soriano, M.N., Saloma, C.A.: Compact Time-independent Pattern Representation of Entire Human Gait Cycle for Tracking of Gait Irregularities. Pattern Recognition Letters 31, 20–27 (2010)
2. Cutting, J.E., Kozlowski, L.T.: Recognizing Friends by Their Walk: Gait Perception Without familiary Cues. Bulletin of the Psychonomic Society 9(5), 353–356 (1977)
3. Derlatka, M.: Human Gait Recognition Based on Signals from Two Force Plates. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2012, Part II. LNCS (LNAI), vol. 7268, pp. 251–258. Springer, Heidelberg (2012)
4. Gafurov, D., Bours, P., Snekkenes, E.: Used Authentication Based on Foot Motion. Signal, Image and Video Processing 5(4), 457–467 (2011)
5. Garcia-Pedrajas, N., Ortiz-Boyer, D.: Boosting K-nearest Classifier by Means of Input Space Projection. Expert Systems with Applications 36, 10570–10582 (2009)
6. Katiyar, R., Pathak, V.K., Arya, K.V.: A Study on Existing Gait Biometrics Approaches and Challenges. International Journal of Computer Science 10(1), 135–144 (2013)
7. Lin, Y.C., Yang, B.S., Lin, Y.T., Yang, Y.T.: Human recognition based on kinematics and kinetics of gait. Journal of Medical and Biological Engineering 31(4), 255–263 (2011)
8. Moustakidis, S.P., Theocharis, J.B., Giakas, G.: Feature Extraction Based on a Fuzzy Complementary Criterion for Gait Recognition Using GRF Signals. In: IEEE 17th Mediterranean Conference on Control & Automation, pp. 1456–1461 (2009)
9. Orr, R.J., Abowd, G.D.: The Smart Floor: a Mechanism for Natural User Identification and Tracking. In: Proc. of Conference on Human Factors in Computing Systems (2000)
10. Pataky, T.C., Mu, T., Bosch, K., Rosenbaum, D., Goulermas, J.Y.: Gait Recognition: Highly Unique Dynamic Plantar Pressure Patterns Among 104 Individuals. J. R. Soc. Interface 9(69), 790–800 (2011)
11. Yun, J.: User Identification Using Gait Patterns on UbiFloorII. Sensors 11, 2611–2639 (2011)
12. Zack, R.S., Tappert, C.C., Cha, S.-H.: Performance of a Long-Text-Input Keystroke Biometric Authentication System Using an Improved K-Nearest-Neighbor Classification Method. In: Fourth IEEE International Conference on Biometrics: Theory Applications and Systems (BTAS), pp. 1–6 (2010)

An Application of the Curvature Scale Space Shape Descriptor for Forensic Human Identification Based on Orthopantomograms

Dariusz Frejlichowski and Piotr Czapiewski

West Pomeranian University of Technology, Szczecin,
Faculty of Computer Science and Information Technology,
Zolnierska 52, 71-210, Szczecin, Poland
{dfrejlichowski,pczapiewski}@wi.zut.edu.pl

Abstract. This paper is devoted to the problem of forensic human identification based on the application of dental biometrics. For this purpose digital orthopantomogram images are used. The complete scheme of the approach is described. It starts with the image enhancement by means of the Laplacian pyramid decomposition. Secondly, the segmentation of an image into sub–images covering single teeth is performed. The process is based on determining the line between the upper and lower jaws by means of integral projections and, later, image intensity and types of teeth. For each obtained sub–image, covering a single tooth, its shape is extracted and the outer contour of this shape is traced. Thus, a shape descriptor can be applied in order to represent an object for the later identification. In this paper, the Curvature Scale Space algorithm is applied and investigated as one of the elements of the biometric identification approach based on digital orthopantomogram images.

Keywords: dental images, forensics, shape descriptors, Curvature Scale Space.

1 Introduction

Dental records are widely used in forensic human identification as one of the most efficient biometric modalities in this application [1]. They are often used in the process of establishing the identity of a person, together with DNA and fingerprints. The popularity of dental biometrics in judicial proceedings comes from their property of discrimination and the robustness to decomposition [2], where for example automatic face recognition would fail [3, 4]. It has led to several attempts for the development of an identification system, such as WinID that compares dental records previously codified by an expert. The second example, which is closer to the approach assumed in the works described in this paper, is the Automated Dental Identification System (ADIS). One of the stages in this system is the automatic extraction of dental biometrics [5]. Whereas WinID applies dental works as a basis for identification, ADIS utilizes teeth shapes extracted from radiograms for this purpose. The ADIS system is an important tool

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 67–76, 2013.

used in the work of the Dental Task Force (DTF), a special service created by the Federal Bureau of Investigation (FBI) [6].

The analysis of radiographic images is extremely significant for various medical and forensic purposes, and hence many general methods related to radiograms have been described [7–9]. However, the specifics of the dental image, and especially of orthopantomograms, must be taken into consideration in order to properly discern between teeth and represent their shape.

The automatic system for the identification of persons based on digital orthopantomograms is composed of image enhancement and segmentation, and feature extraction and recognition. In the first step one can apply, for example, the wavelet-transform [10, 11] or Laplacian pyramid [12, 13]. During the segmentation process the integral projections [15, 16] or active contour models [12] have been applied so far. The extraction of particular teeth shapes is performed using active shapes [14], line scanning [15] or watersheds [17]. The first two algorithms were developed specifically for the intraoral images. Hence, they fail when applied in pantomograms. This problem results from frequent occlusions. The third mentioned method performs better in this case. In the last stage any of the shape description and recognition approaches can be applied.

In this paper the whole algorithm is described. Firstly, the methods for image enhancement are presented. Next, the segmentation and localisation of the object of interest — single teeth — is provided. Both methods are discussed in the second section. Later, in the third section, the algorithm for the representation of the extracted contours is described. The fourth section discusses the experimental conditions and results. Finally, the last section concludes the paper.

2 Image Enhancement, Segmentation and the Extraction of Teeth Shapes

The image enhancement applied for the pantomograms starts with the Laplacian decomposition, proposed in [18]. The Laplacian pyramid's layers are derived through the subtraction of consecutive layers of a Gaussian pyramid. This process is represented by means of the following formula:

$$X_k = \downarrow(\bar{X}_{k-1}), \qquad L_k = X_{k-1} - \uparrow(X_k), \tag{1}$$

where X_0 is the original image, $\downarrow(X)$ and $\uparrow(X)$ represent the process of downsampling and upsampling the image by a factor of 2, $\bar{X}_k$ is a low-pass filtered image of X_k, and L_k is the successive layer of the Laplacian pyramid. The low-pass filtering is performed using the Gaussian filter.

For the Laplacian pyramid, a contrast equalisation function is used (as described in [19]):

$$f(x) = a(\frac{x}{|x|})|x|^p, \tag{2}$$

where "x is normalized to the range $[-1, 1]$ and the factor a is needed for rescaling the resulting image to the original dynamic range" [19]. The exponent p controls the degree of non-linearity. According to further modifications provided by [20], a new representation of the Laplacian pyramid, applied in our approach, can be formulated as:

$$\begin{cases} r(x) = G \cdot x \cdot (1 - \frac{|x|}{M})^p + x, & if\ |x| \leq M \\ r(x) = x, & elsewhere \end{cases}, \tag{3}$$

where M is the upper limit for linear enhancement and G is the constant gain.

In our previous experiments [13, 16, 17] the best results were obtained when using the following combination of methods: 1) the averaging of the two Laplacian pyramid layers next to last, 2) unsharp filter on the second layer, 3) contrast enhancement, based on the eq. 2–3. This approach was used at the stage of image pre–processing.

The second stage consists of the segmentation of an image. Firstly, the upper and lower jaws are separated by means of integral projections. Usually, the initial starting point for this process is established by the user. However, our goal is to fully automate this process. Hence, in the method described in this paper, this point is selected as the horizontal integral projection around the centre of the image with the lowest value. Taking into account the characteristics of a pantomogram (the separating line is not horizontal), only some of the pixels (20% of the image width), which lay closest to the frontal teeth gap selected earlier, are used for the derivation of projections. The area between the necks of two adjacent teeth is easy to locate on the pantomogram. Thus, the aforementioned line is applied for the location of the position of the neck for each tooth. Later, points lying on splines representing gaps between the necks of two adjacent teeth are selected. For the upper and lower jaw, an array of values is obtained, comprised of the intensity values of the points lying on the splines going through their respective dental pulps. The sharp spikes visible on the plot representing these values correspond to the dark regions surrounded by light intensities, what indicates a gap between teeth necks. For the matching process only small subsets (with the pre-assumed number of elements) of the values in the array are selected.

For the indication of a spike value corresponding to the gap between the necks of teeth for a processed tooth, lying on the position p_c, the Bayesian probability $P(x_i, p_c)$ calculated for each point on the curve (x_i) is used:

$$P(x_i, p_c) = I(x_i)G(x_i, p_c)D(x_i, p_c), \tag{4}$$

where:
I — the intensity of the range filtered and inverted original image in the point x_i,
G — a discrete Gaussian function with expected value equal to the horizontal position of the last detected gap, displaced left or right (depending on the current search direction) by the amount of pixels that is equal to the average width of the tooth at position p_c,

D — a function introduced to limit the number of pixels analysed in each iteration. It is equal to 1 for points, for which the horizontal distance from the previous detected gap is between 75% and 175% of the expected width of a tooth on position p_c, with regard to the current search direction, and equal to 0 elsewhere.

At each iteration, the current maximal value obtained using the above formula is added to the list of gap positions and it is used as a starting point for the next iteration. The iterations end at the border of an image or when the number of gaps for either the upper or lower jaw is equal to 8. Usually, the obtained gap locations are reliable indicators for the space between the teeth. However, sometimes the vertical line is insufficient for the distinction between two occluding teeth. Thus, additionally another point between them is localised by means of a greedy algorithm, that iteratively moves one pixel towards the radiogram's top or bottom, selects the pixel in horizontal neighbourhood with the highest intensity on the inverted, range–filtered image and uses it as the basis for the next iteration. The number of iterations is equal to the half length of an average tooth on a pantomogram. The last obtained point is taken as the second point applied for the separation. Using it, as well as the previously derived point, the segmentation line is established.

Later, the areas lying below the teeth roots are removed by means of the curve separating jaws. This line is shifted vertically until such an alignment is obtained, where the sum of pixels through which it passes is smaller than the surrounding result. That indicates the area between the teeth line and the cheekbone.

An exemplary pantomogram enhanced and segmented by means of the described approach is provided in Fig. 1.

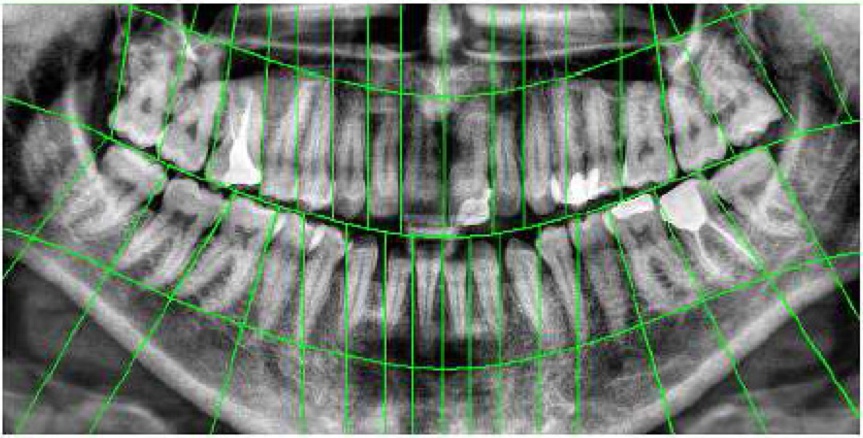


Fig. 1. An exemplary result of the image pre-processing and segmentation, described in this section

After the segmentation of the image, we can extract the shapes of particular teeth. Firstly, in order to lower the influence of noise and enlarge the areas with similar intensity, the opening operation on the image is performed. Later, the entropy filtering is applied. It detects edges of areas with similar colours. Then, the watershed algorithm is used in order to divide the image into small segments. The dimensions of the obtained areas depends on the size of a structuring element used in the morphological opening — the larger the element, the larger size of the segments.

For each segment the following features are obtained: the centroid, the normalised mean intensity, and the normalised vertical distance from the centroid to the curve separating the jaws. Next, the distance between obtained centroids is derived, and 50 of the closest segments to a processed segment are selected for the calculation of the average intensity. The distinction value of the i-th segment is derived using the equation:

$$D(i) = \sum_{j=1}^{N} \max(\bar{I}(i) - \bar{I}(j), 0), \tag{5}$$

where: $\bar{I}(i)$ and $\bar{I}(j)$ are the average intensities of i-th and j-th segments. These values help in determining whether a segment is brighter than its surrounding segments or not.

Later, a fitness function is calculated in order to select the segments belonging to the tooth. The values of this function depend on the type of the tooth. For the incissors only the distinction function and vertical distance from the jaws separating curve are used, with the weights of 0.7 and 0.3, respectively. For other types of teeth the average intensity is also included, with the weights of 0.4 (distinction function), 0.4 (average intensity) and 0.2 (vertical distance from the jaws separating curve).

When all fitness functions for segments are obtained, the process of selection of those belonging to a tooth begins. Basing on the performed experiments we could assume that a fitness function value has to be larger than 0.4 for incissors, and 0.5 for the canine and first premolar. For other types of teeth the threshold is established as the average intensity multiplied by 0.8. The selected regions are morphologically dilated, what results in the removal of borders that separate them. Finally, the contour of a tooth can be traced and extracted (see Fig. 2).

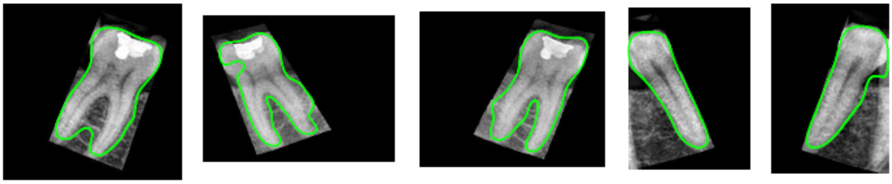


Fig. 2. Examples of the segmented teeth images

3 Representation of the Teeth Contours Using the Curvature Scale Space Transform

The extracted contours of teeth are described using the Curvature Scale Space (CSS) transform [21, 22]. The CSS representation is calculated using the convolution of a path–based parametric representation of a curve extracted from the boundary of processed shape with a Gaussian function of increasing variance σ^2. The zeros of curvatures of the convolved curves are extracted and combined in a scale space representation. Those values are derived during the evolution of the planar curve changed by the expanding Gaussian function.

The closed planar curve r is represented parametrically for axes x and y using the normalized arc length parameter u:

$$r(u) = \{x(u), y(u) | u \in [0, 1]\}. \tag{6}$$

The evolved curve is represented as Γ_σ:

$$\Gamma_\sigma(u) = \{\chi(u, \sigma), \psi(u, \sigma)\}, \tag{7}$$

where:

$$\chi(u, \sigma) = x(u) \otimes g(u, \sigma), \qquad \psi(u, \sigma) = y(u) \otimes g(u, \sigma), \tag{8}$$

and:
$\otimes$ — the convolution operator,
g — Gaussian function of width σ, calculated using the formula:

$$g(u, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-u^2/2\sigma^2}. \tag{9}$$

The curvature κ of Γ can be formulated as:

$$\kappa(u, \sigma) = \frac{\chi_u(u, \sigma) - \psi_{uu}(u, \sigma) - \chi_{uu}(u, \sigma) - \psi_u(u, \sigma)}{(\chi_u(u, \sigma)^2 + \psi_u(u, \sigma)^2)^{3/2}}, \tag{10}$$

where:

$$\chi_u(u, \sigma) = \frac{\partial}{\partial u}(x(u) \otimes g(u, \sigma)) = x(u) \otimes g_u(u, \sigma), \tag{11}$$

$$\chi_{uu}(u, \sigma) = \frac{\partial^2}{\partial^2 u}(x(u) \otimes g(u, \sigma)) = x(u) \otimes g_{uu}(u, \sigma), \tag{12}$$

$$\psi_u(u, \sigma) = y(u) \otimes g_u(u, \sigma), \tag{13}$$

$$\psi_{uu}(u, \sigma) = y(u) \otimes g_{uu}(u, \sigma). \tag{14}$$

Finally, the CSS image I_c gives in result a multi-scale representation of zero crossing points:

$$I_c = \{(u, \sigma) | \kappa(u, \sigma) = 0, u \in [0, 1], \sigma \geq 0\}. \tag{15}$$

The last stage of the descriptor's construction is the selection of contour maxima locations in CSS image I_c. They are used as a representation for a shape.

The similarity between two represented objects is calculated through matching between two sets of maxima, representing them. The method firstly finds any possible changes in orientation in one of the two shapes, then a circular shift is applied in order to compensate their influence. Later the Euclidean distance is calculated between the two sets, indicating the dissimilarity measure.

4 Discussion on the Experimental Conditions and Obtained Results

The approach described in the previous two sections was experimentally investigated using 60 digital pantomograms, belonging to 30 individuals. For each person two images were applied that were taken at various time points. One image was treated as a template pattern and the second as a test object.

After the pre–processing and segmentation of an image and the extraction of teeth shapes, each obtained contour was represented using the Curvature Scale Space transform. As can be seen on Fig. 3, the CSS shape descriptors obtained for different teeth differ significantly, which confirms the validity of their application in the discussed problem. For the derivation of the similarity between two descriptions the method discussed in the previous section was applied.

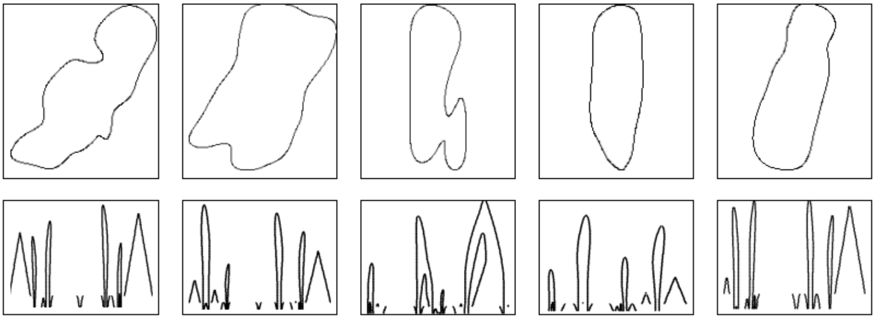


Fig. 3. Examples of teeth contours and obtained CSS descriptors

For further processes the location of each tooth was required, because the matching was performed on the level of particular tooth types — incisors extracted from pantomograms under recognition were matched with incisors from all base images, canines were compared with canines, and so on. The best recognition results were achieved when using the molars. Nevertheless, other tooth types should also be taken into account, since for certain people this type of tooth was not always present.

The indication process was performed separately for each of the four quarters of a pantomogram image. For each of them, the comparison started from the back of the jaw and continued towards the center of the image. If a tooth was present on a particular position in both the test and template sub–images, matching would be performed. The Euclidean distance indicated the template person for a single tooth. The template class with the largest number of indications was selected as the recognized person.

Using the described approach an experiment was performed. As a result of this experiment 20 of the 30 test objects were identified properly, which gives a recognition rate equal to 67%. Although this result may seem far from ideal it is mainly connected with the difficult character of the digital orthopantomogram images.

5 Conclusions

An approach for automatic people identification using pantomogram images was described in this paper. It consisted of four stages: image enhancement, image segmentation, tooth localisation, and recognition by means of tooth shapes for a particular person. The shapes were represented using the Curvature Scale Space transform and matched with the shapes stored in the template base. The largest number of correctly indicated teeth decided about the selection of the template class that was most similar to the test image.

During the experiment 60 pantomogram images acquired from 30 people were used — half of the images were selected as templates assigned to particular persons, and the other half was used as test objects. This means that two images obtained at various points in time were used for each person. The achieved identification rate was equal to 67%, what is not enough but nonetheless one has to keep in mind the particularly difficult character of the dental orthopantomogram images. Nevertheless, future works on this problem are planned. First, other shape representation techniques can be applied and experimentally investigated, including evaluation of potential hardware implementation [23]. Second, other methods for the pre–processing of the image can be analysed in order to emphasise the locations of particular teeth before the extraction of their shapes. Finally, other methods for making a final decision about the identified person could be developed and evaluated.

Acknowledgements. The authors of this paper wish to thank gratefully to R. Wanat, MSc for his significant help in developing and investigating the described approach.

References

1. Bowers, M.C.: Forensic Dental Evidence. Elsevier Academic Press (2004)
2. Lee, S.S., Choi, J.H., Yoon, C.L., Kim, C.Y., Shin, K.J.: The Diversity of Dental Patterns in Orthopantomography and its Significance in Human Identification. Journal of Forensic Science 49(4), 784–786 (2004)
3. Spaun, N.A.: Face Recognition in Forensic Science. In: Li, S.Z., Jain, A.K. (eds.) Handbook of Face Recognition, pp. 655–670. Springer (2011)
4. Shchegoleva, N.: Facial Surface Reconstruction in 3D Format. Journal of Theoretical and Applied Computer Science 6(4), 37–50 (2012)

5. Fahmy, G., et al.: Towards an Automated Dental Identification System (ADIS). In: Zhang, D., Jain, A.K. (eds.) ICBA 2004. LNCS, vol. 3072, pp. 789–796. Springer, Heidelberg (2004)
6. Nassar, D., Ammar, H.H.: A Prototype Automated Dental Identification System (ADIS). In: Proc. of the 2003 Annual National Conference on Digital Government Research, pp. 1–4. Digital Government Society of North America (2003)
7. Gut, P., Chmielewski, L., Kukolowicz, P., Dabrowski, A.: Edge-Based Robust Image Registration for Incomplete and Partly Erroneous Data. In: Skarbek, W. (ed.) CAIP 2001. LNCS, vol. 2124, pp. 309–316. Springer, Heidelberg (2001)
8. Bator, M., Chmielewski, L.J.: Finding Regions of Interest for Cancerous Masses Enhanced by Elimination of Linear Structures and Considerations on Detection Correctness Measures in Mammography. Pattern Analysis and Applications 12(4), 377–390 (2009)
9. Kukolowicz, P.F., Dabrowski, A., Gut, P., Chmielewski, L., Wieczorek, A., Kedzierawski, P.: Evaluation of Set-up Deviations During the Irradiation of Patients Suffering from Breast Cancer Treated With Two Different Techniques. Radiotherapy and Oncology 75(1), 22–27 (2005)
10. Lu, J., Healy Jr., D.M.: Contrast Enhancement of Medical Images Using Multiscale Edge Representation. Optical Engineering 33(7), 2151–2161 (1994)
11. Dippel, S., Stahl, M., Wiemker, R., Blaffert, T.: Multiscale Contrast Ehnahncement for Radiographies: Laplacian Pyramid Versus Fast Wavelet Transform. IEEE Trans. on Medical Imaging 21(4), 343–353 (2002)
12. Zhou, J., Abdel-Mottaleb, M.: A Content-Based System for Human Identification Based on Bitewing Dental X-ray Images. Pattern Recognition 38(11), 2132–2142 (2005)
13. Frejlichowski, D., Wanat, R.: Application of the Laplacian Pyramid Decomposition to the Enhancement of Digital Dental Radiographic Images for the Automatic Person Identification. In: Campilho, A., Kamel, M. (eds.) ICIAR 2010, Part II. LNCS, vol. 6112, pp. 151–160. Springer, Heidelberg (2010)
14. Chen, H., Jain, A.K.: Automatic Forensic Dental Identification. In: Jain, A.K., Flynn, P., Ross, A.A. (eds.) Handbook of Biometrics, pp. 231–251. Springer (2008)
15. Jain, A.K., Chen, H.: Matching of Dental X-ray Images for Human Identification. Pattern Recognition 37(7), 1519–1532 (2004)
16. Frejlichowski, D., Wanat, R.: Automatic Segmentation of Digital Orthopantomograms for Forensic Human Identification. In: Maino, G., Foresti, G.L. (eds.) ICIAP 2011, Part II. LNCS, vol. 6979, pp. 294–302. Springer, Heidelberg (2011)
17. Frejlichowski, D., Wanat, R.: Extraction of Teeth Shapes from Orthopantomograms for Forensic Human Identification. In: Real, P., Diaz-Pernil, D., Molina-Abril, H., Berciano, A., Kropatsch, W. (eds.) CAIP 2011, Part II. LNCS, vol. 6855, pp. 65–72. Springer, Heidelberg (2011)
18. Burt, P.J., Adelson, E.H.: The Laplacian Pyramid as a Compact Image Code. IEEE Trans. on Communications 31(4), 532–540 (1983)
19. Vuylsteke, P., Schoeters, E.: Image Processing in Computed Radiography. In: Proc. of International Symposium on Computerized Tomography for Industrial Applications and Image Processing in Radiology, pp. 87–101. DGZfP (1999)
20. Stahl, M., Aach, T., Buzug, T.M., Dippel, S., Neitzel, U.: Noise-Resistant Weak-Structure Enhancement for Digital Radiography. In: Hanson, K.M. (ed.) Medical Imaging 1999: Image Processing. SPIE, vol. 3661, pp. 1406–1417. SPIE (1999)

21. Mokhtarian, F.: Silhouette-Based Occluded Object Recognition Through Curvature Scale Space. Machine Vision and Applications 10(3), 87–97 (1997)
22. Roh, M.-C., Christmas, B., Kittler, J., Lee, S.-W.: Robust Player Gesture Spotting and Recognition in Low-Resolution Sports Video. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3954, pp. 347–358. Springer, Heidelberg (2006)
23. Frejlichowski, D.: Analysis of Possible System-Level Hardware Implementation of the Selected Shape Description Algorithms. Journal of Theoretical and Applied Computer Science 6(4), 51–58 (2012)

The Impact of Temporal Proximity between Samples on Eye Movement Biometric Identification

Paweł Kasprowski

Institute of Informatics
Silesian University of Technology
Gliwice, Poland
kasprowski@polsl.pl

Abstract. Eye movements identification is an interesting alternative to other biometric identification methods. It compiles both physiological and behavioral aspects and therefore it is difficult to forge. However, the main obstacle to popularize this methodology is lack of general recommendations considering eye movement biometrics experiments. Another problem is lack of commonly available databases of eye movements. Different authors present their methodologies using their own datasets of samples recorded with different devices and scenarios. It excludes possibility to compare different approaches. It is obvious that the way the samples were recorded influences the overall results. This work tries to investigate how one of the elements – temporal proximity between subsequent measurements – influences the identification results. A dataset of 2556 eye movement recordings collected for over 5 months was used as the basis of analyses. The main purpose of the paper is to identify the impact of sampling and classification scenarios on the overall identification results and to recommend scenarios for creation of future datasets.

Keywords: eye movement biometrics, behavioral biometrics, classification.

1 Introduction

The main problem of visual perception is that eyes register scene with uneven acuity. Only the part of the scene that falls on the fovea – region in the middle of the retina – is seen with correct sharpness. All other regions of retina are able to register only contours and fast movements. Therefore, eye movements are very important for correct recognition of objects in visual field. That is why eye movements are one of the fastest and the most accurate movements of a human being [8].

Eye movements may be divided into voluntary and involuntary. Voluntary eye movements are the effect of our will – we want to look at something. Involuntary eye movement is reflex action, automatic response to some stimulus, for instance sudden movement near the edge of vision. Both movements have physiological aspects but also depend on our previous knowledge or experience – having also important behavioral elements. That is why human identification using eye movements should be classified as behavioral biometrics [17][19].

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 77–87, 2013.

1.1 Human Identification Using Eye Movements

Eye movements are yet another possibility to perform human identification. The idea that eye movements may be used for human identification is about 10 years old [16][20]. There have been several publications showing that the method is promising, however it is still on the very early research stage because collecting eye movements' data is difficult and eye movement capturing devices (eye trackers) are still relatively expensive. That is why in most cases the datasets collected by researchers are not publicly available.

Most of the eye movements recording experiments have used a 'jumping point' pattern originally introduced in [16]. In such kind of experiment the stimulus is forcing eye movements - the examined individuals should follow the point on the screen with their eyes. Such a recording is easy to analyze, because it requires that fixations and saccades happen in specific moments. However, there are several interesting experiments with different scenarios, including faces observation [25] or text reading [11]. There is also an attempt to perform identification without any information about a stimulus [18].

The problem common for all biometric methods using behavioral traits is so called *learning effect* [13]. When using the same stimulus for several times the person familiarizes with it and eye movements tend to become automatic. It is for instance clearly visible for texts – eye movements of a person reading the text that she already knows are very different from eye movements of the person reading a text new for her [24]. Such kind of reading is therefore often called *skimming*.

The learning effect is especially visible for very short intervals. A person that completes the same task for several times in very short period tends to "learn" the task and the movements (eye movements in our example) are becoming similar to each other. The effect of similarity between subsequent experiments is stronger for shorter periods and becomes invisible for very long periods (because human body "forgets" the task).

In our paper we tried to investigate how the interval between subsequent experiments performed by the same person influences the identification rates. To our best knowledge it is the first paper that analyses that aspect of eye movements' biometrics, however the problem has been already introduced in [15].

1.2 Eye Movement Verification and Identification Competition

There are several methods to analyze eye movements, but until quite recently it has been difficult to compare them due to lack of publicly available datasets (like for fingerprints [5] or faces [22]).

The First Eye Movement Verification and Identification Competition (EMVIC) organized in 2012 as an official BTAS conference competition was the first opportunity to compare different approaches [15]. The aim of the competition was to correctly identify individuals on the basis of their eye movements. The organizers prepared four different datasets of eye movements collected with different stimuli and different eye trackers (denominated A, B, C and D).

All datasets were divided into two parts:

- Training set, containing labeled samples;
- Testing set, containing samples with hidden labels.

The aim of the competitors was to build their classification models using labeled samples and then try to use those models to classify unlabeled samples from the testing set. There were about 50 competitors with over 500 separate submissions.

The main problem of EMVIC was that the results were inconsistent for different datasets. For datasets A and B the identification accuracies were better than 90% and for datasets C and D the best accuracies approximated 60%. The question arose as to the reasons of such differences.

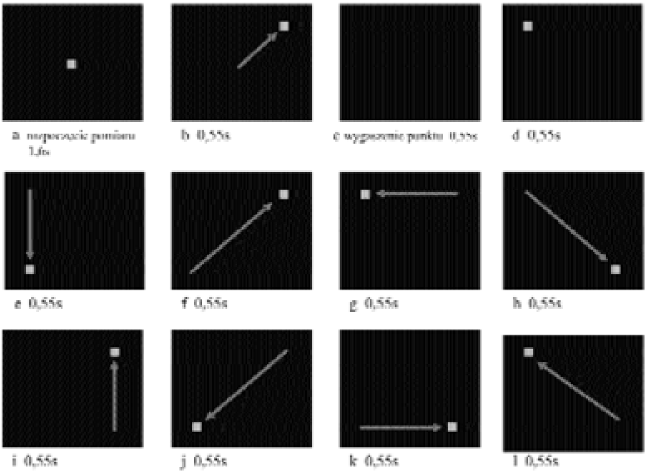
One of the obvious reasons was higher number of samples per person for datasets A and B. Another reason could be binocular data in A and B (which was already studied in [26] and [21]). In [15] authors suggested also that low quality of data in A and B could be the reason of better performance. In this paper we investigate another of the possible reasons: impact of proximity between consecutive samples.

As eye movements measurement has very important behavioral aspect, we may expect that measurements taken in short periods of time may share some common information that is unwanted for identification purposes. For instance it has been proven that the person's attitude may highly influence eye movements. If a person is angry or amused, eye movement patterns are different than for the same person in neutral state [7]. Similarly, it is possible to find out the level of tiredness by examining eyes reaction to salience regions of images [27][12][28].

Other obvious time-dependent factors which are not directly connected with human properties but may influence the measurement are e. g. lightning conditions or existence of devices which may interfere with eye tracker like cellular phones, computers etc.

1.3 Dataset

To check the impact of short term learning effect we decided to perform experiments that would check if temporal differences in the dataset might change the overall accuracy results. We used a dataset of 2556 eye samples collected for over 5 months. There were 61 different subjects under the test with uneven distribution of number of samples (from 4 to 129 samples per subject). The dataset was originally a part of dataset B from EMVIC. Samples were taken with 250Hz frequency using Ober2 eye tracker. It was a jumping point on 2x2 matrix used as stimulus. One experiment lasted for 8128 ms and the stimulus was exactly the same for every experiment (Fig. 1). Every sample consisted of 2048 measurements of horizontal position of the left eye and 2048 measurements of vertical position of the left eye giving 4096 attributes. Additionally every sample had two properties: timestamp of measurement in seconds (t) and subject id (id).



a rozpoczęcie pomiaru 1,6s

b 0,55s

c wygaszenie punktu 0,55s

d 0,55s

e 0,55s

f 0,55s

g 0,55s

h 0,55s

i 0,55s

j 0,55s

k 0,55s

l 0,55s

Fig. 1. Graphical representation of dataset [15]

2 Calculation of Time Interval Influence

To check the theory that time interval between samples influences the accuracy results several datasets were created with the same number of samples and the same distribution of samples per subjects. The only element that differed datasets was the minimal time interval between samples belonging to the same person.

2.1 Data Preparation

We started with creating a dataset for which a minimal time interval (MTI) between subsequent measurements of the same person was one week (604,800 seconds). We used original samples from the full dataset (D) and created a new dataset (W) by taking only samples fulfilling the interval condition. Algorithm filtering dataset D is shown below, where *t(s)* is the timestamp of measurement for sample *s*.

```
sort all samples in D by timestamp
foreach(id: ids from D)
  $t_{last}$ = 0
  foreach(s: samples with given id from D)
    If t(s)-$t_{last}$ > MTI
      add sample s to dataset W
      $t_{last}$ = t(s)
```

The new dataset W consisted of 222 samples belonging to 37 subjects. The average number of samples per person was 6 with minimal value equal to 4 and maximal value equal to 11.

The next step was creation of other datasets. The property that differed datasets was a minimal time interval (MTI) between subsequent measurements of the same person. Every dataset consisted of the same number of samples for each person and was created using a subset of original samples from dataset D. The algorithm to build datasets was almost identical to the presented above. An additional parameter was the way the samples were initially sorted. There were two possibilities: start from the beginning of the dataset (with samples recorded at the beginning of the experiment) or start from the end of the dataset (with samples recorded as the last samples during the whole experiment). Because we wanted to see if it influences results, we decided to use both methods and store samples in datasets denominated as F for the oldest samples and R for the newest.

The above procedure created two datasets ('F' and 'R') for every interval. There were seven different values of MTI used as presented in Table 1. The columns 'real minimal/maximal intervals' show actual intervals for subsequent trials found in datasets.

Table 1. Minimal time intervals used for experiments

index	MTI	real minimal interval	real maximal interval
0	0 (no minimal interval)	11s	13d, 21h, 31min, 26s
1	1 minute	61s	46d, 3min, 5s
2	10 minutes	13min, 25s	46d, 3min, 5s
3	1 hour	1h, 1s	57d, 19h, 44min, 45s
4	6 hours	6h, 10min, 11s	66d, 23h, 13min, 9s
5	1 day	24h, 33min, 42 s	66d, 23h, 13min, 9s
6	1 week	7d, 6s	66d, 23h, 13min, 9s

The main idea of the paper was that identification results were dependent on minimal time interval between samples. It was assumed that samples taken in shorter intervals have some additional (usually unwanted) time related information that could improve classification results. The datasets were examined using four different classic classification algorithms, namely:

- J48 (Java version of C45 algorithm [23]),
- Random Forest [3],
- Naïve Bayes
- SVM (using Sequential Minimal Optimization algorithm) [29].

Every dataset was validated using standard 10-fold cross-validation method. The result for every dataset-algorithm pair was then stored as accuracy value. Accuracy is the number of correctly classified samples to the overall number of samples. Because it was 37 different classes, probability of random guess was less than 3% and therefore accuracy seemed to be a good and sufficient measure.

It is very important to emphasize that all algorithms were used with standard parameters [10] without any optimizations towards results improvements. As the main purpose of the paper was to compare different datasets in the same environment (and not to obtain the best result), additional parameters tuning could introduce some biases. Nevertheless, the results are quite good as for identification (one-to-many) task.

2.2 Results

The results of the experiment are presented in Table 2.

Table 2. Accuracy of each classification method for every dataset

dataset	nb	j48	rf	smo
0F	22,97	20,27	28,38	52,25
0R	28,37	21,17	37,39	54,04
1F	18,46	13,51	18,02	31,53
1R	26,12	17,56	24,32	44,59
2F	17,11	11,26	23,87	33,78
2R	25,67	20,72	22,97	48,65
3F	17,56	12,16	22,97	34,23
3R	22,97	22,97	21,17	45,95
4F	18,46	13,96	18,47	32,43
4R	16,66	12,16	18,47	33,33
5F	18,01	13,51	20,27	31,08
5R	18,01	16,21	14,41	36,94
6F	12,16	12,61	15,31	29,73
6R	12,16	13,06	16,67	29,73

Datasets with F suffix were created by taking samples starting from the earliest while datasets with R suffix were created by taking samples from the last experiments (as it was described in the previous section). Every dataset had the same number of samples (222) and the same distribution of samples among 37 different subjects.

What can be seen clearly from the results is strong negative correlation between the interval and accuracy (-0.61). Because so called "forgetting curve" [9] is considered to be non-linear we also calculated correlation of accuracy with logarithm of the interval and obtained the correlation equal to -0.94. Fig 2 shows logarithmic regression of the average results with coefficient of determination (R^2) equal to 0.8637. For linear regression coefficient of determination was 0.8088 and for exponential regression was 0.8433 so logarithmic trend line was chosen as the best fitting option.

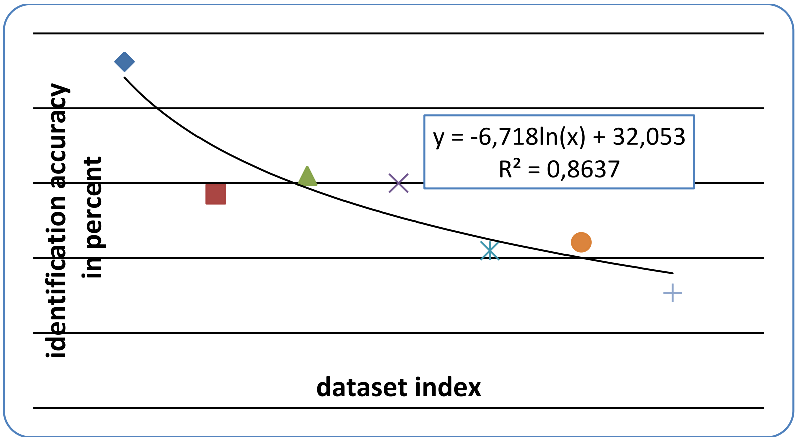


Fig. 2. Averaged accuracy results with logarithmic trend line

The accuracies of datasets classification were averaged for all datasets with the same interval and all classification methods. When calculating mutual significance of differences between these results it occurred that the only significant difference may be found between 0 and 1 minute interval (p=0.02). It means that memory effect is clearly visible only for very short intervals.

Another comparison was performed between samples of type 'F' (i.e. first samples of the specific person) and samples of type 'R' (i.e. last samples of the person). The hypothesis was that samples taken later - when a person is already familiarized with stimulus - will be more stable and therefore easier to classify. Indeed, average accuracy for 'R' datasets was better for every classification method. However, the differences were not significant, with the highest significance for SVM method (p=0.068).

To see if the results are stable for different signal conversions we repeated the same classification experiments on datasets converted using different algorithms previously used in eye movement biometric identification [1][4][11][14][15][16][18][21][25][20][13]. The results were similar, always showing negative correlation, however for some conversions the correlation was not strong.

Table 3. Correlation of classification accuracy and minimal time interval between samples for different signal conversions (*time* means correlation to time in seconds, *log* is correlation to logarithm of time in seconds)

applied conversion	time	log(time)
fourier spectrum	-0.42	-0.91
cepstrum	-0.7	-0.81
first derivate (velocity)	-0.22	-0.74
second derivate (acceleration)	-0.48	-0.92
direction (in radians)	-0.8	-0.84
wavelet transform (DWT)	-0.46	-0.86
high pass filter	-0.56	-0.93
low pass filter	-0.63	-0.94

3 Sessions Analyzes

The experiment presented in the previous section showed clearly that samples taken in short intervals should not be mixed in training and testing sets for classification.

On the other hand, closer look into the dataset that was used for experiments revealed that in most cases samples of the same person were recorded in series. It is natural and it is probably a common strategy for every biometric experiment, because with this scenario only one equipment setup is required to obtain several samples. However, as it was proven in the previous section, samples taken in the same session are not independent. Therefore, we decided to divide the original dataset into sessions and check how it influences classification results.

The *session* was defined as a set of samples taken from the same person with minimal interval between two samples less than 10 minutes. Preprocessing algorithm found 685 sessions in the dataset. The number of sessions per subject differed from 1 to 26 sessions with average number of sessions equal to 11. Every session consisted of one to eight samples (see Fig. 3).

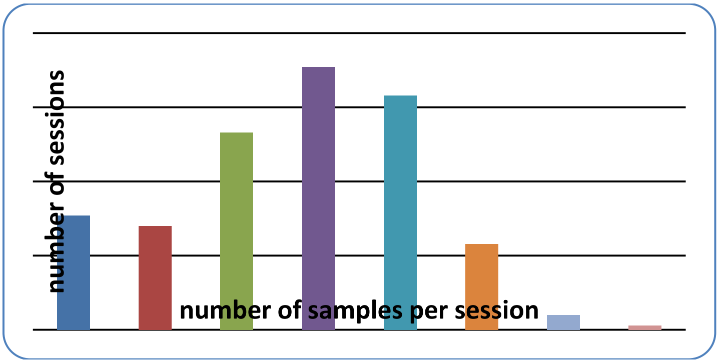


Fig. 3. Histogram of number of samples per session

To check identification results using 10-fold cross-validation and to obtain reliable results it was necessary to remove samples of subjects for which number of sessions was too low. Therefore samples of all subjects with less than 10 sessions were removed from dataset. The reduced dataset consisted of 2195 samples from 29 subjects divided into 567 sessions.

Because samples from the same session were considered to be dependent there were three classification experiments proposed:

- Experiment using only first samples from each session (referred as *first sample* in Table 4).
- Experiment using only last samples from each session (*last sample*).
- Experiment using all samples from dataset but with folding algorithm that doesn't divide samples from the same session to different folds (*all samples*).

Contrary to cross validations used in experiment described in Section 2, where folds were created by stratifying basing on number of samples per subject, this time stratification was done basing on sessions. It means that all samples from the same session had to be in the same fold.

Table 4. Accuracies of classification for three datasets

method	first sample	last sample	all samples
J48	23,27	21,02	29,33
NB	28,39	25,80	27,94
RF	34,41	29,52	40,58
SMO	50,09	52,12	64,83
average	**34,04**	**32,12**	**40,67**

Table 4 shows that there are no significant differences between datasets build from first and last samples from the session. However, when all samples from the session were taken, it significantly improved results. It must be remembered that the latter classification used much more samples both for training and testing (2195 versus 567). It shows that collecting samples in series is not generally a bad idea, but care must be taken how samples from the same session are used.

4 Conclusions

The results of analyzes presented in the paper clearly show that the data collecting scenario may significantly influence the overall results and classification possibilities for a dataset. Especially time related factors were carefully studied and impact of so called memory effect was analyzed.

All calculations used only eye movements datasets but it may be assumed that the conclusions could be extended to other behavioral biometric experiments.

Basing on our findings we advise that every behavioral biometric sample should be stored together with information about the exact timestamp when it was collected.

It is possible to collect more than one sample during one session with subject but these samples should never be mixed in training and testing set when evaluating performance. Additionally, we showed that using all samples collected during one session in the training set improves the overall performance of the system. Even if samples from the same session were considered dependent, multiplying the number of samples would give effect similar to bootstrap samples used in bagging algorithm [2].

References

1. Bednarik, R., Kinnunen, T., Mihaila, A., Fränti, P.: Eye-movements as a biometric. In: Kalviainen, H., Parkkinen, J., Kaarna, A. (eds.) SCIA 2005. LNCS, vol. 3540, pp. 780–789. Springer, Heidelberg (2005)

2. Breiman, L.: Bagging predictors. Machine Learning 24(2) (1996)
3. Breiman, L.: Random Forests. Machine Learning 45(1), 5–32 (2001)
4. Brzeski, R., Ober, J.: The biometrical system of the authentication realized on the ground of the movement of the eye, Techniki Komputerowe, Biuletyn Informacyjny IMM (2005)
5. Cappelli, R., Ferrara, M., Maltoni, D., Turroni, F.: Fingerprint Verification Competition at IJCB 2011 Proceedings of International Joint Conference of Biometrics (2011)
6. Chen, Y., Dass, S.C., Jain, A.K.: Localized iris image quality using 2-d wavelets. Advances in Biometrics (2005)
7. Deane, F., Henderson, R., Mahar, D., Saliba, A.: Theoretical examination of the effects of anxiety and electronic performance monitoring on behavioural biometric security systems. Interacting with Computers 7(4), 395–411 (1995)
8. Duchowski, A.: Eye Tracking Methodology. Theory and Practice. Springer, London (2003)
9. Ebbinghaus, H.: Memory: A contribution to experimental psychology. Teachers college, Columbia university (1913)
10. Hall, M., et al.: The WEKA data mining software: an update. ACM SIGKDD Explorations Newsletter 11(1), 10–18 (2009)
11. Holland, C., Komogortsev, O.V.: Biometric Identification via Eye Movement Scanpaths in Reading. In: IEEE International Joint Conference on Biometrics, IJCB (2011)
12. Ji, Q., Zhu, Z., Lan, P.: Real-time nonintrusive monitoring and prediction of driver fatigue. IEEE Transactions on Vehicular Technology 53(4), 1052–1068 (2004)
13. Kasprowski, P., Ober, J.: Enhancing eye movement based biometric identification method by using voting classifiers. In: SPIE Defence & Security Symposium, SPIE Proceedings, Orlando, Florida (2005)
14. Kasprowski, P.: Human identification using eye movements. Doctoral thesis. Silesian Unversity of Technology, Poland (2004)
15. Kasprowski, P., Komogortsev, O.V., Karpov, A.: First Eye Movement Verification and Identification Competition at BTAS 2012. IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems, BTAS (2012)
16. Kasprowski, P., Ober, J.: Eye Movements in Biometrics. In: Maltoni, D., Jain, A.K. (eds.) BioAW 2004. LNCS, vol. 3087, pp. 248–258. Springer, Heidelberg (2004)
17. Keesey, U.T.: Effects of involuntary eye movements on visual acuity. JOSA (1960)
18. Kinnunen, T., Sedlak, F., Bednarik, R.: Towards task-independent person authentication using eye movement signals. In: Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications. ACM, New York (2010)
19. Leigh, R.J., Zee, D.S.: The Neurology of Eye Movements. Oxford University Press (2006)
20. Maeder, A.J., Clinton, B.F.: A visual attention approach to personal identification (2003)
21. Nguyen, V.C., Vu, D., Lam, S., Tung, H.: Mel-frequency Cepstral Coefficients for Eye Movement Identification. In: IEEE International Conference on Tools with Artificial Intelligence, ICTAI (2012)
22. Ortega-Garcia, J., Fierrez-Aguilar, J., Simon, D., et al.: MCYT baseline corpus: a bimodal biometric database. IEE Proc.-Vis. Image Signal Process. 150(6) (2003)
23. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers (1993)
24. Rayner, K.: Eye movements in reading and information processing: 20 years of research. Psychological Bulletin 124(3), 372 (1998)
25. Rigas, I., Economou, G., Fotopoulos, S.: Biometric identification based on the eye movements and graph matching techniques. Pattern Recognition Letters 33(6), 786–792 (2012) ISSN 0167-8655

26. Rigas, I., Economou, G., Fotopoulos, S.: Human eye movements as a trait for biometrical identification. In: IEEE Fifth International Conference on Biometrics: Theory, Applications and Systems (BTAS), pp. 217–222 (2012)
27. Schleicher, R., et al.: Blinks and saccades as indicators of fatigue in sleepiness warnings: looking tired? Ergonomics 51(7), 982–1010 (2008)
28. Sodhi, M., Reimer, B., Llamazares, I.: Glance analysis of driver eye movements to evaluate distraction. Behavior Research Methods, Instruments, & Computers 34(4), 529–538 (2002)
29. Vapnik, V.N.: Statistical learning theory. Wiley Interscience (1998)

Biomedical Distributed Signal Processing and Analysis

Marek Penhaker, Vladimir Kasik, and Vaclav Snasel

VSB - Technical University of Ostrava, Faculty of Electrical Engineering and Computer Science, Ostrava, Czech Republic
{marek.penhaker,vladimir.kasik,vaclav.snasel}@vsb.cz

Abstract. The paper deal about the electrophysiology distributed data processing and analysis. The aim of the work was the distance computer environment for biomedical data processing services with effective data visualizing. The system combine end user interactive interaction with biomedical data was realized with the use of rapid calculating by FPGA technology. Combining of these technologies allows a vide range spectrum of users to quick access to proceed biomedical data. This article involved the possible topological proposal and implementation of that biomedical distributed signal processing system.

Keywords: Ultra Fast, Parallel, Signal Analysis, Web Services FPGA, Biomedical.

1 Signal Handling

Handling with biomedical data is not uniform and there are not any common standards till today. Many producers in the medical devices mainly contain their own data formats, what means in incompatible. Converting the data seems to be loss-making and between these formats is very limited. Nowadays way is to handling data through a web services interface that provides the user with many advantages. In that case, data storage using a standard XML format is often used and supported by many tools and programming languages.

In concern of medical practice, there is important that biomedical records are moderate from any web sites and networks without the specific programs installed. For this mind, a system of handling of biomedical data were developed, which is based on earlier work of the authors [1] - [4].

The system write on about web system that gives the opportunity to view of stored biomedical data and also allows you to store records in one of three predefined formats – EDF, DAT and XML. As part of this web system application is a viewer of stored records, which replaces the traditional programs and enables a view of recorded data directly on the website without need of additional programs. The system for storage and exchange biomedical data is based on ASP.NET 2.0 platform .NET Framework and Microsoft SQL Server.

For storage of biomedical data is used a data storage server. Every file uploaded to the server has its own unique name that is used as ID in the database. The basic functions of Web applications provide: display a list of all records stored in a database,

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 88–95, 2013.

add a new record and manage users. The Web application has its own biomedical data browser that allows graphical display of such an EEG records from XML file.

A significant part of the application is the application stored biomedical data, providing by display the 20-channels EEG records. It is also possible to modify the scale of the axis and to move the view in a direction of x axis.

Additionally the system were enlarged to evaluate the biomedical data with the detection of real time signals with ultra fast response and accuracy. The precise possibility of the application can be done by FPGA multiparallel system. This hardware is handling data with a data server, but can also be connected to the analog inputs for on-line processing of biomedical signals. [5]

Nowadays, complex applications are being developed, the scope of which exceeds the ability of one person. The development application is usually involved more programmers. In the work were again used part of their applications or other companies. The programmer thus can no longer rely only on himself, as it was in the days of the first operational system. The first tool for modular application development platform for Microsoft Windows is dynamic libraries. [6] - [8]

At the beginning of the 90th years were mostly created separate applications with very little ability to communicate. This deficiency was removed in the mid-90th years, when the company introduced technology Microsoft COM (Component Object Model). The great advantage of component technology is its language neutrality in binary form. For each component was defined interface that facilitates communication between the client and the relevant component. Over time, it became apparent that even the technology has its limitations. Nowadays, the modular architecture of increasingly more used. Components used are usually small and simple. The main disadvantage of COM components is that they hide their internal implementation. The only component that describes the appropriate interface. This prevents inheritance at the source. [9]

2 Problem Definition

Biomedical data representation on different devices was mostly done on local terminals with displaying of measured data using web services. Processing of these data is done either by post-processing after measurement and in minor cases on-line. On-line processing puts high demands on hardware and so this process is destined only for specialized work. Enabling data processing on any number of remote terminals for users working with internet ser currently represents significant progress in training of physician, cooperation and collaborations in real time with ultra fast and inexpensive analysis.

Distance processing of biomedical data was done by using web service terminals is solved several ways. The methods of implementation is described below having a native access in the installation of a powerful multiparalell processor with direct connection to server or database server, providing a web application platform and user cost-effective way of data evaluation.

The proposal of multiparalell ultrafast system for biomedical processing is the tasks that can be used in very large scale of integration composition of FPGA - Field Programmable Gate Array. With the FPGA technology are these features possible not only for off-line processing of saved data, but also the possibility of real time evaluation and data handling to distance physicians by web services. [10]

In cases where the digitized signal is processed in several steps, it is preferred to use a computer-operated pipelining. Typically, such calculations are solved by computer, however it does not perform these operations simultaneously but in sequential manner. Such a conventional approach requires higher CPU computing power. In addition the time required to calculate is difficult to predict and the whole situation has an impact on the safety of the calculation. The solution, which better reflects the computational structure for the signal processing is distributed architecture in both the system and the units inside the chip. Therefore, there is the FPGA technology utilized, in which the calculating phases are designed in separate processing modules. [10]

3 New Methods Description

Biomedical data processing requires specifically detection variety in stored measurements. The variety is evaluated qualitative and qualitative parameters and the time parameters. The proper evaluation in variety is usually not often easy, due the biomedical signals volatile on quasi periodic or stochastic characteristics and could also be changed by artificial or physiological artifacts.

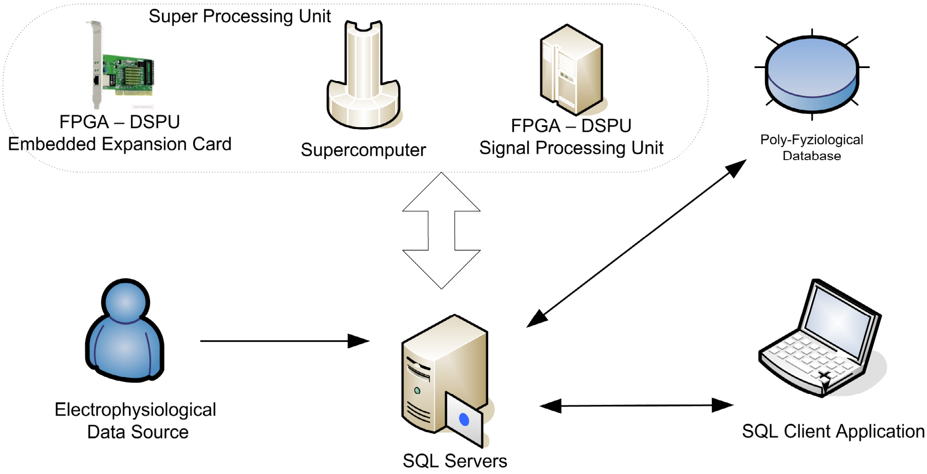


Fig. 1. Overall system structure for distributed signal processing

The way of ultra speed real time data processing should be connected with the computing demands of data handling. Fig. 1.

The multivariance mathematical transformations currently provide a very efficient and effective algorithm for detection and manipulation of biosignals, whatever is used for implementing many of the calculation estimate, especially at the local workstations. Using the multivariance mathematical transforms is easy thanks to the number of steps in one subscription by 2 x n, all the same for use on large real-time huge biomedical data it requires fully ultra fast parallel computing with high discharge. Implementation of this option can be currently done with just two possible architectural designs. First one is a huge supercomputer with many processors linked with tiny data pipelines and the other is a ultra fast parallel computing with FPGA technology, which is nevertheless similar to these with supercomputer ability. In this point of view, there are three alternatives to fast signal processing as shown in Fig. 2.

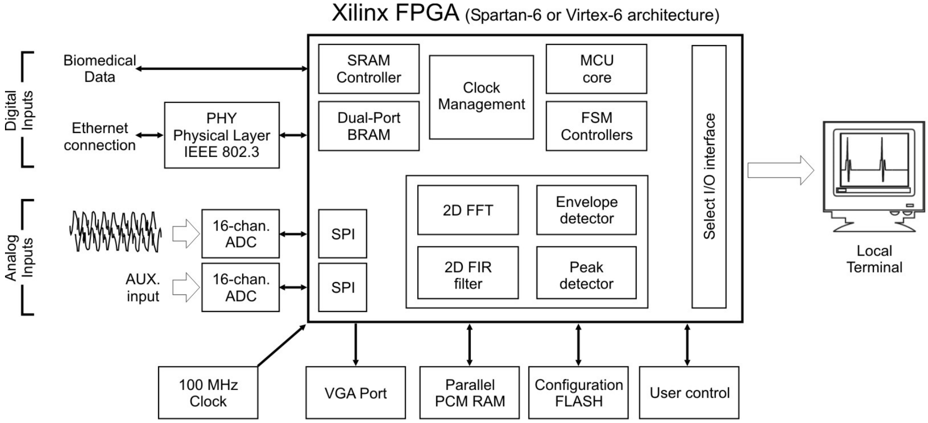


Fig. 2. FPGA – DSPU Detailed Structure

That drown describe the system of transmits saved biomedical data to the coprocessors unit for processing on request. Processed data are giving back after processing of data server, from where may be displayed on the web system. [11]

There are the three possibilities of co-processing units: a) FPGA – DSPU (Digital Signal Processing Unit) like Embedded Expansion Card b) FPGA – DSPU like standalone unit with an Ethernet connection and c) supercomputer like an alternative. Possibility and b of all options are established on FPGA programmable logic, which uses the same hardware and software accession in data housing. Hardware design is based realized by parallel implementation of some sub-algorithms in cooperation with the DSP units. Software design allows to process complex calculations and algorithms supporting MicroBlaze soft processor supplement Fig. 3.

In generally is the co-processing unit including FPGA is designed in such ways. The one solution is a PCI flaring card for managing data from server. Mentioned solution is effective and allows to make maximum signal transmission throughput server and the signal processing unit. The main drawback is necessary off a fixed location of the unit and modification of the data server configuration at the same time, which may not be advisable. Differ solution from co-processing unit is a standalone module for to be connected through a communication network. Anywhere the standard Ethernet network with TCP / IP protocol is applied. The advisable of the solution is not just freedom of data server positioning, and also a possibility to use this unit with more servers and / or clients on the network. [12] Network feature, withal, poses claims on the network communication module, containing requirements for proper safety and maintenance of data transmission in the network. The unit is equipped with a 2x16 channels for digitizing analog biomedical signals and their processing in real time. Data throughput is limited primarily by sampling rate of ADC, which is outside the chip, so it can be further customized according to application requirements. The Ethernet layer circuitry enables the web interface functions extension of the unit. This functionality is used for options parameterization of control units and also for data communication with co-processor unit.

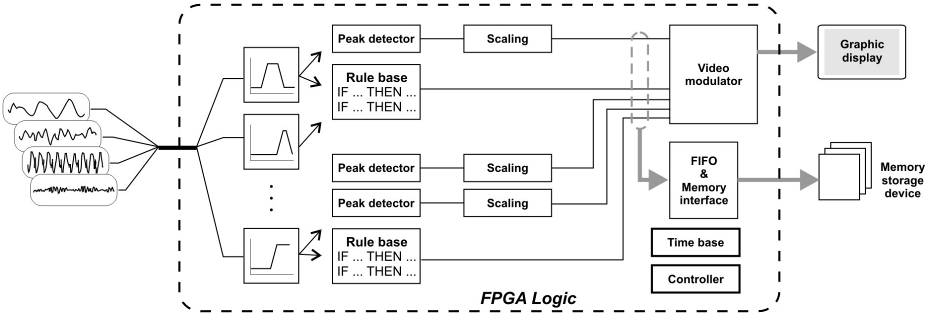


Fig. 3. Fundamental structure of DSP units connection with FPGA Device

An FPGA - DSPU Unit is a powerful computational tool for rapid evaluation of biomedical signals in real time. Individual calculation algorithms as frequency filters, peak detection, envelope detection and further, are implemented in hardware. Target architecture for implementation is Spartan-6 or Virtex-6 FPGA with soft-core MCU option. For fast data storage are available either fast FPGA Block RAM inside FPGA or off-the-chip fast static memory with parallel access. The advantage of using FPGA is also expanded diagnostic possibilities of digital circuits either in FPGA or in digital circuits around the chip. The logic inside the FPGA design also includes a VGA interface for real-time monitoring of calculated data or diagnostic variables of the system. [13]

In case of dataflow the FPGA – DSPU co-processing unit come second link between web server and SQL server. The operation of FPGA is driven by a web service that transmits commands to the unit. Partial digital biosignal processing algorithms on signal from SQL Server is realized in on real time. The encountered signals is processed by web services and offered to the authorized physician's web service. [13]

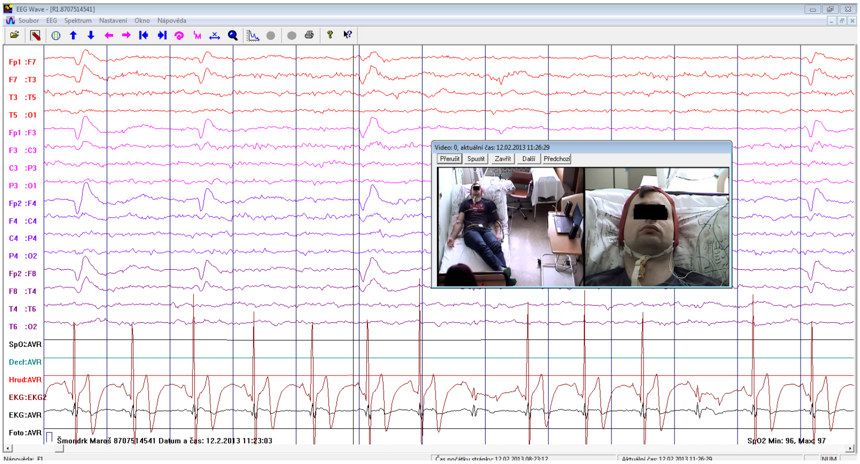


Fig. 4. Evaluation of medical image in real-time processing of input image gives a result in terms of reference points, whose position is further evaluated

The advance of the realized tests is the system for the EEG evaluation on patients and their correspondence to epileptic attack. To determine the EEG signal was used multivariance mathematic transform. Also to determine individual patient images in clinical practice were used as a method for digital image processing Fig. 4 line of epileptic attacks was evaluated by detection algorithm that was tested to locate any balanced points in the image. Situation of these points and their movements is based on subsequent in classification the patients signal manifestation Fig. 5.

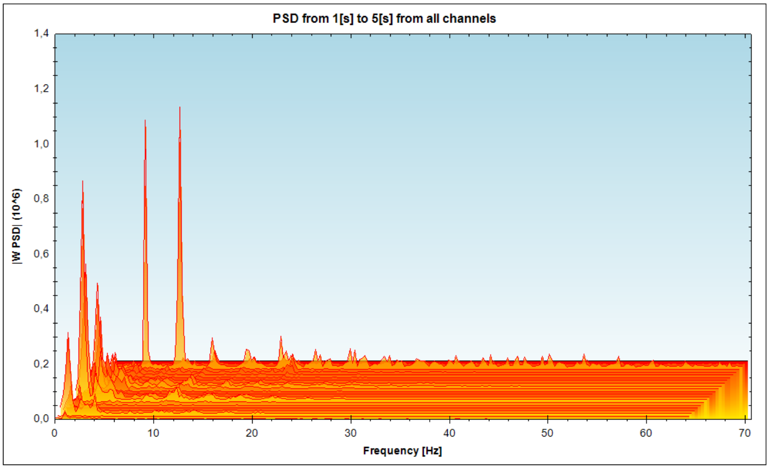


Fig. 5. Power spectral density by CSA for all channels in the selected time period from ultrafast capable system with multiparalel processing algorithms that detected the preliminary signs of epileptic attack in frequencies 7, 11 and 40Hz to 60 Hz

Application of epileptic attack recognized from EEG records were used multivariance mathematical transform methods implemented on an FPGA gantry. In the same time the fundamental parameters of signal were derived from EEG signal and mainly variety in magnitude and frequency characteristics is evaluated. Likewise, there was examine retardation in the similarity signal propagation within all channels between the first signs of attack in EEG and the corresponding video signals.

4 Conclusion

The introduced problem solution manipulation, detection of signals in real time with multi processing ultra fast processing by FPGA usage carries new possibilities for common workplace, but also for customer of web applications for access and visualization also with additional possibility of processing the far-out signals in real time. The introduced system was tested on a real polysomnographic video records and well verified by application of this system. There were instead traditional methods of epileptic seizure artifact detection determined.

The traditional ways of biosignal processing are very common, but there is a lot of information outside the traditional way of perception. There are hidden information in the measured biosignal especially in high frequencies which are not yet consternate on. We can access the information in the same way of as previous methods for low frequency analyzing, but at can take a lot of human-machine time with weak results. The proposed way of processing introduced in this article is based on multi parallel signal processing with ultra extremely high potential of recognition the even small changes of signal behavior.

Acknowledgment. The work and the contributions were supported by the project SP2013/35 “Biomedical engineering systems IX” and TACR TA01010632 “SCADA system for control and measurement of process in real time”. The paper has been elaborated in the framework of the IT4Innovations Centre of Excellence project, reg. no. CZ.1.05/1.1.00/02.0070 supported by Operational Programme 'Research and Development for Innovations' funded by Structural Funds of the European Union and state budget of the Czech Republic.

References

1. Krawiec, J., Penhaker, M., Krejcar, O., Novak, V., Bridzik, R.: System for Storage and Exchange of Electrophysiological Data. In: Proceedings of 5th International Conference on Systems, ICONS 2010, April 11-16, pp. 88–91. IEEE Conference Publishing Services, Menuires (2010), doi:10.1109/ICONS.2010.23
2. Carr, J.J., Brown, J.M.: Introduction to Biomedical Equipment Technology, 4th edn. Prentice Hall (2001) ISBN 0-13-010492-2
3. Acharya, U.R., Tamura, T., Ng, E.Y.K., Min, L.C., Suri, J.S. (eds.): Distributed Diagnosis and Home Healthcare. American Scientific Publishers (2010)
4. Gala, M., Babusiak, B., Novak, V.: Automatic creation of hypnogram. Communications 13(1), 47–51 (2011) ISSN 1335-4205

5. Babusiak, B., Gala, M.: The eye-blinking artifacts detection and elimination in the EEG record. In: IFAC Workshop on Programmable Devices and Embedded Systems (PDES 2009): Proceedings, pp. 254–259. Int. federat. automatic control, IFAC secretariat, schlossplatz 12, a-2361 Laxenburg, Austria (2009) ISBN 978-3-902661-41-8
6. Schalk, G., Mellinger, J.: Pactical Guide to Brain/Computer Interfacing with BCI 2000. Springer, Heidelberg (2010) 978-1-84996-091-5
7. Gala, M., Babusiak, B.: Sleep EEG Automatic Analysis. In: Ifac Workshop on Programmable Devices and Embedded Systems (PDES 2009): Proceedings, pp. 242–247. Int. federat. automatic control, IFAC secretariat, schlossplatz 12, a-2361 Laxenburg, Austria (2009) ISBN 978-3-902661-41-8
8. Kukucka, M., Krajcuskova, Z.: The Frequency and the Shape of Driving Signal Influence in Measurement of the Active Points. Advances in Electrical and Electronic Engineering 10(3), 181–186 (2012) ISSN 1336-1376
9. Kohlish, O., Schaefer, F.: Physiological changes during computer task: responses to mental load or to motor demands. Ergonomics 39(2), 213–224 (1996)
10. Valentová, H., Havlík, J.: Initial Analysis of the EEG Signal Processing Methods for Studying Correlations between Muscle and Brain Activity. In: Khuri, S., Lhotská, L., Pisanti, N. (eds.) ITBAM 2010. LNCS, vol. 6266, pp. 220–225. Springer, Heidelberg (2010)
11. Krejcar, O., Janckulik, D., Motalova, L.: Complex Biomedical System with Biotelemetric Monitoring of Life Functions. In: Proceedings of the IEEE Eurocon 2009, St. Petersburg, Russia, May 18-23, pp. 138–141 (2009), doi:10.1109/EURCON.2009.5167618, ISBN 978-1-4244-3861-7; Hughes, J.W., Stoney, C.M.: Depressed mood is related to high-frequency heart rate variability during stressors. Psychosomatic Medicine 62, 796–803 (2000)
12. Accortt, E.E., Allen, J.J.B.: Frontal EEG asymmetry and premenstrua dysphoric symptomatology. Journal of Abnormal Psychology 115(1), 179–184 (2006)
13. Krejcar, O., Janckulik, D., Motalova, L.: Complex Biomedical System with Mobile Clients. In: Dössel, O., Schlegel, W.C. (eds.) WC 2009. IFMBE Proceedings, vol. 25(5), pp. 141–144. Springer, Heidelberg (2009)

Effect of Slice Thickness on Texture-Based Classification of Liver Dynamic CT Scans

Dorota Duda[1], Marek Kretowski[1], and Johanne Bezy-Wendling[2,3]

[1] Faculty of Computer Science, Bialystok University of Technology
Wiejska 45a, 15-351 Białystok, Poland
[2] INSERM, U1099, Rennes, F-35000, France
[3] University of Rennes 1, LTSI, Rennes, F-35000, France
{d.duda,m.kretowski}@pb.edu.pl

Abstract. This paper assesses the impact of slice thickness on texture parameters. Experiments are performed on liver dynamic CT scans, with two slice thicknesses. Three acquisition moments are considered: without contrast, in arterial and in portal phase. In total, 155 texture parameters, extracted with 9 methods, are tested. Classification of normal and cirrhotic liver is performed using a boosting algorithm. Experiments reveal that slice thickness does not considerably influence the stability of the parameters. They also enable to assess the rate of parameter dependency on slice thickness. Finally, they show that applying different slice thicknesses for training and testing the CAD system requires slice thickness-independent parameters.

Keywords: texture analysis, classification, dynamic CT, liver, slice thickness, stability of the parameters, tissue characterization, CAD.

1 Introduction

Various medical imaging techniques are available for acquiring a diagnostic information, e.g., Computed Tomography (CT), Positron Emission Tomography (PET), ultrasound, Medical Resonance Imaging (MRI), Single Photon Emission Computed Tomography (SPECT). With their rapid development, the image quality has been significantly improved. The images acquired over the years are characterized by increasing spatial and grey level resolution, and decreasing slice thickness. The number of images obtained within a single examination increases. The appropriate interpretation of an image information is thus a very complex task, but decisive for a correct diagnosis and treatment proposal.

Due to the fact, that the men are not able to identify such huge amount of information stored in the images, the (semi)automatic Computer Aided Diagnosis (CAD) systems become of a rapidly growing interest. Combining different image analysis techniques (like texture analysis [1] or segmentation) and classification algorithms, they appear to be a very promising diagnostic tool.

In most of the CAD systems based on imaging data two main stages of work can be distinguished. The first one, called *training*, is a preparation of the system

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 96–107, 2013.

for the recognition of several classes of tissue, representing different pathologies. The second stage is the applicationof the system for the (semi)automatic recognition of new cases, not yet diagnosed. Due to the constant changes of image acquisition protocols, the question arises whether the protocols for new recognized images can be different from those that were used during the system training. The question seems all the more important given the fact that there is still no universal consensus on image acquisition protocols, so the images obtained from different machines could be of different properties and qualities.

The aim of our study is to examine the effect of slice thickness on texture parameters characterizing hepatic tissue on dynamic (contrast-enhanced) CT images. Our choice was dictated by the fact that the image database that we have been creating for about 10 years includes images of several slice thicknesses: from the oldest ones, of 10 mm, to the most recent ones, of 1.3 mm. So far, we have not found any research concerning the impact of slice thickness on texture-based classification of liver CT images with different contrast product concentrations.

In this study, we first assess the influence of slice thickness on parameter stability. Secondly, we study the possibility of tissue differentiation, with the most known parameters and different combinations of slice thicknesses used for system training and testing. Then, the parameter dependency on slice thickness is evaluated. Finally, the classification of two types of liver tissue, characterized by parameters which are least dependent on slice thickness is performed.

The next section includes a short description of related works. In Sect. 3 the system for classification of multiphasic textures is presented. Then, the methods for assessing the effect of slice thickness on the system performance are proposed. An experimental validation is described in Sect. 4. Conclusions and future work are presented in the last section.

2 Related Work

The effect of image acquisition protocols on CAD system performances has already been investigated in several studies.

In [2] the influence of slice thickness on CT-based texture parameters for cancellous calf bone was studied. Four different slice thicknesses were considered.

The work [3] analyzed the sensitivity of texture features of five different categories to variations in the number of acquisitions, repetition time, echo time, and sampling bandwidth at different spatial resolutions of T2-weighted MR images.

In [4] the effect of acquisition parameters, such as tube currents and voltages (10 different combinations), on texture was assessed on CT images of a cylindric phantom filled with water. Eight different texture parameters were analyzed.

The effect of slice thickness on brain MRI texture classification was studied in [5]. In the work, thick slices were simulated on the basis of thin ones.

The first three of the cited studies showed that changes in all analyzed acquisition parameters could influence the texture parameter values, and thus – the texture discrimination. Only in the last work, the two considered tissue classes were separable even if slice thicknesses differed between training and testing sets.

3 Texture-Based Classification of Hepatic Tissues

3.1 Two Stages of Work: Training and Aiding a Diagnosis

The system we develop for texture-based classification of multiphasic liver CT scans also works according to previously described two-stage model. Since typical CT exams of abdominal organs often consider three acquisition moments, which are related to the contrast product propagation in hepatic vessels, we proposed to analyze triplets of images [6], [7]. The first of the three simultaneously analyzed images corresponds to acquisition without a contrast. The second and the third are taken after its injection, at arterial and portal phases of contrast propagation.

The main steps of system training are shown in Fig. 1. First, a database of image triplets (preprocessed, if needed) is created. Then a Region of Interest (ROI) is drawn at the same position on each of the three corresponding images. Each ROI is characterized by the same vector of texture parameters. Three parameter vectors are thus created, each of them characterizes a texture at different acquisition moment. In the next step, parameters from those three vectors are concatenated, in order to make one "complex" parameter vector, describing a triplet of textures. The label representing a pathology is associated with each complex vector. At this moment, a set of labeled complex vectors could be subjected to a feature selection. Finally, it is used for the construction of the classifiers. Then the second stage of the system work can take place.

In order to recognize a new observation (see Fig. 2), a triplet of images is necessary. The three simultaneously analyzed images (in no contrast, arterial and portal phase) are subjected to the same preprocessing as it was in the training stage. Then three ROIs are drawn – one ROI on each of the three images. The triplet of ROIs is characterized in the same way as in the training stage. Finally, a complex vector of concatenated parameters (corresponding simultaneously to three acquisition moments) is classified. The tissue class that is attributed to this vector, is one of the classes considered in the training stage.

During the system design, many aspects must be investigated to ensure the best possible tissue recognition. One of them is the choice of the most relevant texture parameters. Such a choice must consider the variability of image acquisition settings that could result in different image properties, like those depending on slice thickness. Some ideas for adapting the system for working with images of different slice thicknesses are presented in the next part of our study.

3.2 Parameter Stability

The estimation of parameter stability could help to decide if the parameter is reliable for proper tissue characterization. Parameters sensitive to small changes in ROI size, or small ROI displacements, should be excluded from further analyses.

We adopt the approach considered in [8] in order to assess how the parameter changes over the different ROI locations or sizes. As a measure of its changeability, the classical coefficient of variation (CV, the ratio of the standard deviation to the mean) is used. Stable parameters are characterized by low CV values, and

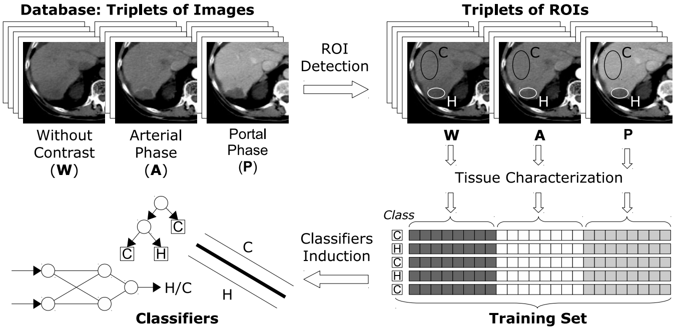


Fig. 1. System for texture-based classification of liver tissues: Training

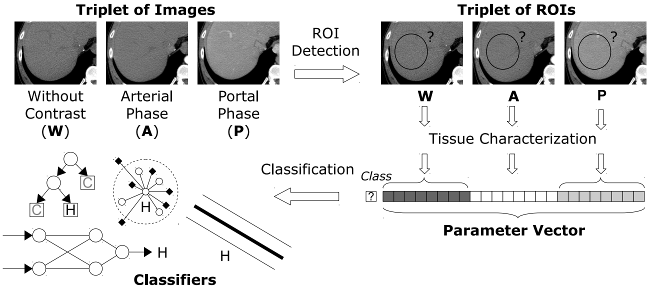


Fig. 2. System for texture-based classification of liver tissues: Aiding a diagnosis

the more unstable is the parameter, the greater is its CV. In our work, the CV of each texture parameter will be calculated for different slice thicknesses.

We propose to apply the following approaches: *Displace* and *Size Changing*. In the *Displace* approach, the initial ROI dimensions are first slightly decreased, then the reduced ROI is displaced in order to take all the possible positions inside its initial boundaries. On the basis of each new ROI location, the value of a parameter is calculated. With the *Size Changing* approach, the initial ROI size is successively reduced, pixel by pixel, by moving each time one of the subsequent ROI vertices. For each of thus obtained ROIs a parameter value is calculated. In both cases, a set of several parameter values, obtained for different ROI locations (*Displace*) or sizes (*Size Changing*) serves to calculate a CV.

3.3 Effect of Slice Thickness on Classification Accuracy

In order to evaluate the effect of slice thickness on the classification accuracy, we propose to perform several experiments, each time considering a different combination of slice thicknesses for training of classifier and for its test. If the image of a particular slice thickness is included in a training set, its counterpart (of the same slice position in a patient's body) of another slice thickness should not be included in the test set.

3.4 Effect of Slice Thickness on Parameter Values

To access the parameter dependency of the slice thickness, we propose a similar approach that we apply for the assessment of the parameter stability. For a given parameter, its dependency on slice thickness will be measured by a "variation" between several parameter values measured by the classical coefficient of variation. These values will be obtained for the same ROI position on images of the same acquisition moment, but characterized by different slice thicknesses.

In order to validate the usefulness of the slice thickness-independent parameters, the experiments on images with different slice thicknesses (different for a training and for a testing stage) will be performed.

4 Experiments

4.1 Database Description

The images, from 29 patients, were gathered at the Department of Radiology of the Pontchaillou University Hospital in Rennes, France. They were acquired with *LightSpeed16* device (*GE Medical Systems*). For each patient, three scan series were performed: without contrast, at arterial and at portal phase. The contrast material, of 100 ml, was injected at 4 ml/s, in an arm vein. The arterial phase acquisitions started about 20 seconds after the contrast product injection, the portal phase acquisitions started from 30 to 40 seconds later. For each of the three scan series, two "versions", corresponding to slice thicknesses of 5 mm and of 1.3 mm, were available. Each thick-slice image had its equivalent in thin-slice one, with the same slice position in the patient's body. All images had the size of 512×512 pixels. They were recorded in DICOM format, with 4096 gray levels. Since only the range of 248 gray levels sufficed for describing the pixels in the considered ROIs, the images were converted to a 8-bit BMP format.

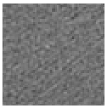
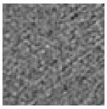
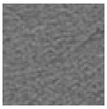
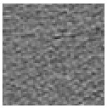
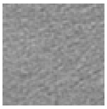
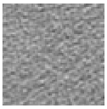
The 303 pairs of images (of a thick slice and of a corresponding thin slice) were taken into account for each of the three acquisition moments. Two classes of liver tissue were represented: cirrhotic and healthy liver (171 and 132 pairs of image triplets, respectively). A square ROI of 60×60 pixels was drawn at the same location on each of the $2 \cdot 3 = 6$ considered simultaneously images. An example of six corresponding ROIs is given in Table 1.

4.2 Texture Parameters Chosen for Evaluation

In total, 155 texture parameters, extracted with 9 methods, were tested (see Table 2). They are based on: First Order Statistics (FO), Gradients (GB), Co-Occurrence Matrices (COM) [9], Run Length Matrices (RLM) [10], Gray Level Difference Matrices (GLDM) [11], Laws Texture Energy (LTE) [12], Fractals (FB) [13], Texture Feature Numbers (TFN) [14], and Autocorrelation (AC) [15].

When applying the COM, GLDM, and RLM methods, the number of gray levels was reduced from 256, used initially, to 64. The CO Matrices and the GLD Matrices were constructed separately for 4 standard directions (0°, 45°, 90°, 135°) and

Table 1. Six ROIs at the same slice position

Acquisition:	**W**ithout Contrast		**A**rterial Phase		**P**ortal Phase	
ROI example:						
slice thickness:	5 mm	1.3 mm	5 mm	1.3 mm	5 mm	1.3 mm

for 5 different distances between the pixel pairs, going from 1 to 5. From each of 20 thus obtained matrices, the same parameters were calculated, 11 parameters by the COM method and 5 parameters by GLDM. The RL Matrices were also constructed for 4 standard directions, each of them served to calculate 8 parameters. For the three aforementioned methods, an averaging of the same parameter corresponding to 4 different directions was done. As the result, the following parameter sets were obtained: COM_{55} (11·5 parameters), GLDM_{25} (5·5 parameters), and RLM_8. The sets COM_{11} and GLDM_5 are the result of averaging of 20 parameter values, calculated for 4 directions and for 5 distances.

The normalized autocorrelation coefficients (AC) and the two TFN parameters (among 7) were calculated separately for 5 different pixel distances, from 1 to 5. They were included, respectively, in the sets AC_5, and TFN_{15} (with 5 other distance-independent TFN parameters). In the set TFN_7, the values of 2 parameters calculated separately for the 5 distances were averaged.

The LTE method provided two parameter sets: LTE_{14} and LTE_5. The first one, composed of 14 parameters, was obtained by the application of 24 filtering masks of size 5×5: 4 symmetric, and 10 pairs of asymmetric ones, each pair consisted of a mask and its transposition. The second set was composed of 5 parameters, corresponding to the application of 3×3 masks, 2 symmetric and 3 pairs of asymmetric ones. The sum of elements of each convolution matrix was zero. For each pair of asymmetric masks, the resulting images were added. Images obtained by the application of symmetric masks were multiplied by two. Finally, the entropies of thus obtained images served as the texture parameters.

The FB method is based on the fractional Brownian motion model [16] and considers 4 pixel distances (1, 2, 3, 4). It provides a set of two parameters: FB_2.

4.3 Effect of Slice Thickness on Parameter Stability

In this experiment, two approaches were applied for the CV calculation: *Displace* and *Size Changing*. The coefficient of variation was calculated on the basis of 9 parameter values. In the *Displace* approach, the ROI was reduced to a 58×58 square in order to take the 9 possible positions inside its initial boundaries. With the *Size Changing* one, the reduced ROI sizes were going from 60×60 to 52×52.

The experiment was performed separately for 12 texture sets, corresponding to all combinations of: 2 slice thicknesses, 2 tissue classes, 3 acquisition moments.

Table 2. Texture parameters chosen for evaluation. In bold – parameters which later turned out to be stable for both slice thicknesses.

Set	Parameter Names
AC$_5$	**(*d*)Autocorr**, where $d = 1, 2, 3, 4, 5$
COM$_{55}$	**(*d*)InvDiffMom**, **(*d*)SumAvg**, **(*d*)SumEntr**, **(*d*)DiffEntr**, **(*d*)Entr**, (*d*)AngSecMom, (*d*)SumVar, (*d*)DiffVar, (*d*)DiffAvg, (*d*)Contrast, (*d*)Corr, where $d = 1, 2, 3, 4, 5$
COM$_{11}$	**InvDiffMom**, **SumAvg**, **SumEntr**, **DiffEntr**, **Entr**, AngSecMom, SumVar, DiffVar, DiffAvg, Contrast, Corr
FB$_2$	**FractalDim**, FractalArea
FO$_4$	**Avg**, Var, Skewness, Kurtosis
GB$_4$	**GradAvg**, GradVar, GradSkewness, GradKurtosis
GLDM$_{25}$	**(*d*)DAvg**, **(*d*)DEntr**, **(*d*)DAngSecMom**, **(*d*)DInvDiffMom**, (*d*)DContrast, where $d = 1, 2, 3, 4, 5$
GLDM$_5$	**DAvg**, **DEntr**, **DAngSecMom**, **DInvDiffMom**, DContrast
LTE$_{14}$	**E5L5**, **S5L5**, **W5L5**, **R5L5**, **S5E5**, **W5E5**, **R5E5**, **W5S5**, **R5S5**, **R5W5**, **E5E5**, **S5S5**, **W5W5**, **R5R5**
LTE$_5$	**E3L3**, **S3L3**, **S3E3**, **E3E3**, **S3S3**
RLM$_8$	**ShortEmp**, **LongEmp**, **Fraction**, **HighGLREmp**, **RLEntr**, GLNonUni, RLNonUni, LowGLREmp
TFN$_{15}$	**MeanConv**, **(*d*)CodeEntr**, Coarse, Hom, CodeVar, ResSim, (*d*)CodeSim, where $d = 1, 2, 3, 4, 5$
TFN$_7$	**MeanConv**, **CodeEntr**, Coarse, Hom, CodeVar, ResSim, CodeSim

Table 3. Maximum parameter CV value (among 12 values, corresponding to: 2 approaches, 2 classes and 3 acquisition moments), obtained for 2 different slice thicknesses

Parameter	**5 mm**	**1.3 mm**
(1)Autocorr	0.0004	0.0007
FractalDim	0.0012	0.0009
(5)SumAvg	0.0016	0.0017
(1)SumAvg	0.0017	0.0018
Avg	0.0018	0.0018
E3L3	0.0023	0.0021
DiffEntr	0.0026	0.0019
S5E5	0.0034	0.0031
HighGLREmp	0.0035	0.0037
(4)CodeEntr	0.0036	0.0027
S3E3	0.0038	0.0032
(1)InvDiffMom	0.0039	0.0048
(3)AngSecMom	0.0047	0.0046
GradAvg	0.0054	0.0052
MeanConv	0.0058	0.0062
DAvg	0.0063	0.0057

For each parameter, the CV values obtained from the same texture set were averaged. We observed that regardless of the approach, the three corresponding averaged CV values, obtained for three acquisition moments, did not differ significantly. Nor did they differ between 2 tissue classes. The *Displace* approach almost always produced slightly lower CVs than the *Size Changing* approach.

Table 3 shows the averaged CV values obtained for the two different slice thicknesses. Each value is the maximum one of the 2·2·3=12 values, obtained for each combination of the approaches, classes and acquisition moments. Since the presentation of all the results occupies too much space, we present the results for 16 selected parameters. However, the CVs calculated on all 155 parameters yield the same conclusions as those obtained from the presented subset.

We can conclude, that the parameter stability, expressed by its coefficient of variation, is not considerably influenced by the slice thickness. Regardless of the extraction method, the CVs corresponding to thick and thin slices were similar. None of the thicknesses had proven to ensure better parameter stabilities.

For further experiments we decided to use the most stable parameters, for which the coefficient of variation does not exceed 0.01. The sets of parameters satisfying this condition were nearly identical for the two slice thicknesses. Only 4 parameters turned out relatively stable with only one of the thicknesses: CodeVar (rejected for thin slices, with CV=0.0102), Contrast, DiffVar, and Hom (rejected for thick slices, with CV equal to 0.0144, 0.0195, and 0.1721, respectively). In total, 93 parameters were accepted for both slice thicknesses (see bolded parameters in Table 2), and the 62 parameters were rejected.

4.4 Effect of Slice Thickness on Classification Accuracy

In this step, we considered the following combinations of slice thicknesses for training and for testing of the classifier:

- ”T/T”: training and testing performed on only thick slices (of 5 mm),
- ”t/t”: training and testing performed on only thin slices (of 1.3 mm),
- ”T/t”: training on only thick slices, testing on only thin slices,
- ”t/T”: training on only thin slices, testing on only thick slices,
- ”mix”: both thick and thin slices in a training and a testing set.

For each combination, the whole set of 303 pairs of image triplets (thick and thin version) was randomly divided into two subsets: a training set (202 pairs) and a testing set (101 pairs). For each subset, the original proportion between tissue classes was preserved. The experiment was repeated 10 times.

Table 4 presents the classification results obtained for the 5 combinations of slice thicknesses used for training and testing. Only the most efficient stable parameter sets are taken into account. Classification was performed with the Weka software [17], using Ensemble of Classifiers with adaptive boosting voting scheme [18] (*AdaBoostM1*) and a C4.5 tree [19] as the underlying algorithm.

We can observe that using the same slice thickness, both for training and for testing the classifier, results in a quite high classification accuracy. The best results are more frequent for thick slice thicknesses (”T/T” possibility). In this case, the maximal classification accuracy (86.73%) is obtained for the parameter sets COM_{55} and RLM_8. Just below the best results are those obtained when only thin slices are considered (”t/t”): 85.15% with the RLM_8 set. Slightly inferior results are observed when both slice thicknesses are considered for training and for testing (”mix”). The best of them is 78.72%, with the set COM_{55}.

Table 4. Classification accuracy obtained by Ensemble of Classifiers for 5 combinations of slice thicknesses, used for training and testing. Each line corresponds to a different set of texture parameters. Only stable parameters are considered.

Set	**T/T**	**t/t**	**T/t**	**t/T**	**mix**
COM_{55}	86.73±2.48	84.75±2.73	63.07±2.24	56.64±5.52	78.72±4.81
COM_{11}	84.16±3.33	82.97±3.29	64.16±2.08	53.77±5.57	77.13±4.26
GLDM_{25}	67.03±2.47	69.81±6.40	56.84±1.76	48.02±2.61	57.93±1.94
GLDM_{4}	63.47±3.51	61.09±4.55	56.94±0.70	49.21±6.32	56.84±0.96
LTE_{14}	78.02±3.45	78.62±2.65	58.42±2.95	44.75±8.87	69.81±3.62
LTE_{5}	74.46±3.67	74.46±4.89	59.71±7.61	49.50±4.48	63.27±5.25
RLM_{8}	86.73±4.43	85.15±1.81	63.96±3.65	58.82±6.65	78.62±3.65

Table 5. Ranking of parameters by their dependency on the slice thickness

Rank	Parameter	Variation	Rank	Parameter	Variation
1	Avg	0.0024	21	(2)CodeEntr	0.0515
2	(5)SumAvg	0.0024	22	(3)CodeEntr	0.0515
3	(4)SumAvg	0.0024	23	RLEntr	0.0658
4	SumAvg	0.0024	24	E3E3	0.0750
5	(3)SumAvg	0.0024	25	E3L3	0.0787
6	(2)SumAvg	0.0024	26	E5E5	0.0803
7	(1)SumAvg	0.0025	27	S5S5	0.0811
8	FractalDim	0.0027	28	S5L5	0.0826
9	(1)Autocorr	0.0104	29	E5L5	0.0847
10	(4)Autocorr	0.0144	30	S5E5	0.0884
11	(5)Autocorr	0.0145	31	(1)SumEntr	0.0905
12	(3)Autocorr	0.0147	32	SumEntr	0.0931
13	(2)Autocorr	0.0155	33	(2)SumEntr	0.0932
14	HighGLREmp	0.0157	34	S3L3	0.0935
15	ShortEmp	0.0296	35	(3)SumEntr	0.0937
16	Fraction	0.0394	36	(4)SumEntr	0.0939
17	(1)CodeEntr	0.0511	37	W5E5	0.0941
18	CodeEntr	0.0514	38	(5)SumEntr	0.0945
19	(5)CodeEntr	0.0514	39	W5L5	0.0953
20	(4)CodeEntr	0.0515	40	S3E3	0.1016

Unsatisfactory results are obtained when the slice thicknesses used for training and testing are different. Most of them do not exceed 60%. The best results for "T/t" and "t/T" combinations are: 64.16% and 58.82%, respectively.

We can conclude that, one should be careful using the system to aid the diagnosis, basing on images of a slice thickness different from those used for training. However, we suppose that, in this case, the use of the texture parameters independent of the slice thickness could improve the system performance. The search for such parameters will be the subject of our next experiment.

4.5 Effect of Slice Thickness on Parameter Values

In this experiment, we considered only those parameters that have been found to be stable for both slice thicknesses: 5 mm and 1.3 mm (see bolded parameters in Table 2). The parameter dependency from the slice thickness was expressed by the coefficient of variation of its two values, obtained from two corresponding ROIs (drawn on the images of a 5 mm and of a 1.3 mm slice thickness).

In total, 303 pairs of images were considered for each of 3 acquisition moments. For each parameter, the 3·303 values of the coefficient of variation were averaged. The ranking of parameters, according to the average coefficient of variation, was performed. Parameters with the lowest average coefficient are considered as least dependent on slice thickness. Table 5 presents the ranking of the first 40 parameters which are least dependent on the slice thickness.

Since we do not know how many parameters are acceptable and sufficient for the best possible tissue recognition, when different slice thicknesses are considered, we will test several sets of the first parameters from the ranking.

4.6 Classification of Textures Characterized by Parameters the Least Dependent on Slice Thickness

In our final experiment, 7 parameter sets were tested: First-8, First-13, First-16, First-23, First-30, and First-39. They were composed, respectively, of the first 8, 13, 16, 23, 30, and 39 parameters from the ranking presented in Table 5. Table 6 presents the classification accuracy obtained for those 7 sets by Ensemble of Classifiers (with the same settings as in the previous classification experiment).

We can conclude that the classification accuracy, obtained for the 5 considered combinations of slice thicknesses are similar only when the First-8 parameter set is used. This set provides the classification accuracy ranging from 84.06%, for the ”t/T” combination, to 87.13%, for the ”mix” one. We can also notice that the application of the First-8 set guaranties the best classification results for the combinations, for which the slice thickness used for testing is different from

Table 6. Classification accuracy obtained by Ensemble of Classifiers for 5 combinations of slice thicknesses, used for training and testing. Each line corresponds to a different set of stable texture parameters, the least dependent of the slice thickness.

Set	**T/T**	**t/t**	**T/t**	**t/T**	**mix**
First-8	85.94±4.12	86.64±2.77	85.35±4.78	84.06±4.08	87.13±3.90
First-13	87.43±2.64	87.53±1.76	67.03±3.59	75.94±3.62	82.58±3.10
First-16	87.82±2.84	87.63±3.17	63.47±2.11	73.96±6.42	85.05±3.21
First-23	89.51±3.50	88.42±2.76	62.38±3.10	74.46±4.82	86.24±3.38
First-30	91.39±3.10	88.52±6.00	64.06±2.92	71.09±5.00	88.22±2.49
First-39	90.79±3.93	92.48±2.10	62.78±1.88	66.74±6.78	88.32±2.71

that used for training (85.35% for "T/t" and 84.06% for "t/T"). For the first of these combinations, the best result is now 21.39% better than the best one obtained in the previous classification experiment, which did not consider the slice thickness-independent parameters (see Tab. 4). For the second one – the best classification accuracy augmented of 25.24%. Nevertheless, this accuracy still remain of about 6%-8% worse than the best results obtained now for the one-thickness cases ("T/T", "t/t").

In general, for the combinations "T/t" and "t/T", the more numerous is the set of parameters, the worse is the classification accuracy. Contrarily, for the one-thickness cases, the classification is generally better for more numerous sets.

With the "mix" combination, the application of the First-8 set leads to obtaining quite good classification accuracy, but not the best. In this case, increasing the number of parameters first results in lowering the classification accuracy (82.58% for the First-13 parameter set), then leads to its successive increase. Finally, the best classification result, for "mix" case (88.32%) is obtained by the most numerous parameter set, First-39, and is 9.60% better than the best accuracy observed, for "mix" case, in the previous classification experiment.

5 Conclusions and Future Work

The experiments allowed us to: (*i*) analyze the influence of the slice thickness on parameter stability, (*ii*) assess the parameter dependency on slice thickness, (*iii*) find parameters which are the least dependent on slice thickness, (*iv*) evaluate the classification accuracy obtained by parameters which are less and more dependent on slice thickness, when different slice thicknesses were simultaneously considered. Several conclusions can be formulated to sum up our study.

First, the parameter stability does not considerably depend on slice thickness. The sets of parameters recognized as stable were nearly identical for the two slice thicknesses. Second, one should be particularly careful when applying a CAD system, when recognized images are of the slice thickness different from the one used for the system training. With quite popular parameters, obtained by the COM or RLM methods, a satisfactory tissue recognition is not possible. In this case, it would be safer to use the slice thickness-independent parameters.

In the future, we plane to perform the experiments with more slice thicknesses. Due to the fact that acquiring the series of images of many different slice thicknesses is practically impossible within a single patient study, we plane to use a phantom. It will enable to assess not only the effect of the slice thickness in the texture-based classification, but also the effect of several other parameters, like image resolution, or the time elapsed from the contrast product injection.

Acknowledgments. We thank Dr D. Olivie for his precious help. This work was supported by the grant S/WI/2/2013 from Bialystok University of Technology.

References

1. Haralick, R.M.: Statistical and structural approaches to texture. Proc. of the IEEE 67(5), 786–804 (1979)
2. Guggenbuhl, P., Chappard, D., Garreau, M., Bansard, J.Y., Chales, G., Rolland, Y.: Reproducibility of CT-based bone texture parameters of cancellous calf bone samples: Influence of slice thickness. Eur. J. Radiol. 67(3), 514–520 (2008)
3. Mayerhoefer, M.E., Szomolanyi, P., Jirak, D., Materka, A., Trattnig, S.: Effects of MRI acquisition parameter variations and protocol heterogeneity on the results of texture analysis and pattern discrimination: an application-oriented study. Med. Phys. 36(4), 1236–1243 (2009)
4. Miles, K.A., Ganeshan, B., Griffiths, M.R., Young, R.C., Chatwin, C.R.: Colorectal cancer: texture analysis of portal phase hepatic CT images as a potential marker of survival. Radiology 250(2), 444–452 (2009)
5. Savio, S.J., Harrison, L.C., Luukkaala, T., Heinonen, T., Dastidar, P., Soimakallio, S., Eskola, H.J.: Effect of slice thickness on brain magnetic resonance image texture analysis. Biomed. Eng. Online 9(60) (2010)
6. Duda, D., Krętowski, M., Bézy-Wendling, J.: Texture-based classification of hepatic primary tumors in multiphase CT. In: Barillot, C., Haynor, D.R., Hellier, P. (eds.) MICCAI 2004, Part II. LNCS, vol. 3217, pp. 1050–1051. Springer, Heidelberg (2004)
7. Duda, D., Kretowski, M., Bezy-Wendling, J.: Texture characterization for Hepatic Tumor Recognition in Multiphase CT. Biocybern. Biomed. Eng. 26(4), 15–24 (2006)
8. Lefebvre, F., Meunier, M., Thibault, F., Laugier, P., Berger, G.: Computerized ultrasound B-scan characterization of breast nodules. Ultrasound Med. Biol. 26(9), 1421–1428 (2000)
9. Haralick, R.M., Shanmugam, K., Dinstein, I.: Textural features for image classification. IEEE Trans. Syst., Man Cybern. 3(6), 610–621 (1973)
10. Galloway, M.M.: Texture analysis using gray level run lengths. Comp. Graph. and Im. Proc. 4(2), 172–179 (1975)
11. Weszka, J.S., Dyer, C.R., Rosenfeld, A.: A comparative study of texture measures for terrain classification. IEEE Trans. Syst., Man Cybern. 6(4), 269–285 (1976)
12. Laws, K.I.: Textured image segmentation. PhD thesis, University of Southern California (1980)
13. Chen, C., Daponte, J.S., Fox, M.D.: Fractal feature analysis and classification in medical imaging. IEEE Trans. Med. Imag. 8(2), 133–142 (1989)
14. Horng, M.H., Sun, Y.N., Lin, X.Z.: Texture feature coding method for classification of liver sonography. In: Buxton, B., Cipolla, R. (eds.) ECCV 1996, Part I. LNCS, vol. 1064, pp. 209–218. Springer, Heidelberg (1996)
15. Gonzalez, R.C., Woods, R.E.: Digital Image Processing, 2nd edn. Addison-Wesley, Reading (2002)
16. Chen, E.L., Chung, P.C., Chen, C.L., Tsai, H.M., Chang, C.I.: An automatic diagnostic system for CT liver image classification. IEEE Trans. Biomed. Eng. 45(6), 783–794 (1998)
17. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. SIGKDD Explorations 11(1), 10–18 (2009)
18. Freund, Y., Shapire, R.: A decision-theoretic generalization of online learning and an application to boosting. J. Comput. Syst. Sci. 55(1), 119–139 (1997)
19. Quinlan, J.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Francisco (1993)

Independent Component Analysis for EEG Data Preprocessing - Algorithms Comparison

Izabela Rejer and Paweł Górski

West Pomeranian University of Technology, Szczecin,
Faculty of Computer Science,
Zolnierska 49, 71-210 Szczecin
{irejer,pagorski}@wi.zut.edu.pl

Abstract. Some scientific papers report that when Independent Component Analysis (ICA) is applied in the preprocessing step of designing a brain computer interface, the quality of this interface increases. At the same time, however, these papers do not provide information about the exact gain in classification precision obtained after applying different ICA algorithms. The aim of this paper is to compare three algorithms for Independent Component Analysis applied in the process of creating a brain computer interface in order to find out whether the choice of a specific ICA algorithm has an influence on the final classification precision of this interface. The comparison will be carried out with a set submitted to the second BCI Competition.

Keywords: Brain Computer Interface, BCI, EEG, preprocessing, ICA, Independent Component Analysis.

1 Introduction

A BCI (Brain Computer Interface) is defined as a communication system in which messages or commands that a user sends to the external world do not pass through the brain's normal output pathways of peripheral nerves and muscles [1]. Although, at first brain computer interfaces were dedicated mainly to people in locked-in state or in completely locked-in state, nowadays, the range of their potential recipients is much wider. They are tested and used by the army, they are used in the entertainment industry and there are also some attempts to use them to control limb prosthesis.

A BCI system is composed of seven parts, followed one after the other in a closed-loop. Succeeding parts of the BCI system are responsible for: measuring brain activity, preprocessing of the acquired signals, describing the signals by a few relevant features (feature extraction), selecting the most relevant features (feature selection), assigning a class to a set of selected features (classification), executing a command assigned to the chosen class and providing feedback to the user informing him about the mental state recognized by the BCI system. Although, all steps of the loop have to be carefully designed in order to build a successful BCI system, four of them play the major role: preprocessing [2-4], feature extraction [5-6], feature selection [7-9] and

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 108–119, 2013.

classification [10-12]. This paper deals with the first of these four main steps, it is with the preprocessing step. The aim of this step is to transform the acquired brain signals to its purer form by eliminating artifacts, reducing the ongoing brain activity, enhancing the primary components. One of the methods used in the preprocessing step is Independent Component Analysis (ICA).

Independent Component Analysis is a method for transforming a set of mixed signals into a set of independent components. It has been reported in some publications that this method allows to detect artifacts in originally recorded signals [13]. Due to this, ICA seems to be a good choice for preprocessing of electroencephalographic data (EEG) used as control signals in brain computer interfaces (BCI). After applying ICA on a set of EEG data, some components should reflect original data sources and one or more components should reflect artifacts. In the process of feature selection, which is a further step of signal processing leading to classification, features calculated for the "artifact components" should be discarded from the set of features while features calculated for the components essential for the classification precision should be preserved.

One can ask why features calculated for the "artifact components" will be discarded from the feature set. The answer is quite straightforward. There are two main types of artifacts activity. Some artifacts, such as eye artifact, appear rhythmically during each trial, while others, such as unexpected body movements, appear in unexpected moments along the whole experiment. Since artifacts from both groups do not dependent on the class coded in the recorded signals, they do not enhance the classification precision. Hence, when the classification precision is a measure used in the feature selection process, features calculated for the components reflecting artifacts should be discarded from the feature set.

There are a lot of scientific papers describing applications of ICA in BCI research [13-15]. In most of them there is underlined that by applying ICA for data preprocessing, the classification precision increases. However, these papers do not specify the exact profits (in terms of classification accuracy) gained by using different algorithms for calculating independent components. Does it mean that the choice of the ICA algorithm is of no importance for the classification precision?

The aim of this paper is to examine whether ICA transformation, performed according to three most popular algorithms, has a major or only minor influence on the classification precision in different experimental settings. Hence, three questions are posed in the paper:

- Is the classification precision essentially higher when ICA transformation is performed?
- Has a choice of a specific algorithm for performing ICA an influence on the classification precision?
- Is there any correlation between the classification precision (after performing ICA) and the number of input features used in the classification process?

In order to answer these questions, a set of experiments was planned and performed. In all experiments a data set from the second BCI Competition was used. The results of the experiments, together with a short discussion, are presented in the paper.

2 Independent Component Analysis

The problem of a blind source separation (BSS) consist in finding a matrix W such that the linear transformation will allow to recover the source signals from a set of mixed signals [16-17]. The term 'blind' means that no prior information about the source signals or the mixing process is available [16].

Independent Component Analysis (ICA) is one of the most popular BSS method. ICA problem can be stated as follows. Let's assume that there are n linear mixtures $x_1, \dots x_n$ of n independent components. Vector x (observed signals) can be written as:

$$\mathrm{x} = \mathrm{As} \tag{1}$$

where A represents a mixing matrix with the size of $n \times n$, and s is the vector of independent components. The aim of ICA is to find a matrix W (i.e. an inverse of the matrix A) to reverse the mixing effect. Then, after computing the matrix W, we can obtain the independent components by [18-19]:

$$y = wX \cong s \tag{2}$$

Most of the popular ICA algorithms put some constraints on the mixed signals. First of them is a statistical independence between source signals s; second, a non-Gaussian distribution of the source signals and the third - the equality of the number of source signals and the number of mixture signals. While two first constrains are main assumptions utilized by many algorithms, the third one is introduced only to decrease the algorithm complexity (it causes that the mixing matrix is square). Furthermore, it is assumed that each source signal has the unit variance $E\{s_i^2\} = 1$. To hold this assumption, the matrix of the source signals is whitened before the ICA calculation [18-19]. One more assumption, introduced only to simplify the algorithm, is that all mixture signals are centered.

As was mentioned earlier, ICA does not require any prior information about the source signals. Instead, ICA algorithms utilize the concept of statistical independency of the mixed signals. According to the formal definition, the variables a and b are said to be independent if information about the value a does not give any information about the value b and vice versa [17], [19]. Technically, independence can be defined in terms of the probability density function (pdf) [18]:

$$f(x_1, x_2, \dots, x_m) = f_1(x_1) f_2(x_2) \dots f_m(x_m) \tag{3}$$

where $x_1, x_2, \dots, x_m$ are random variables.

There are two main approaches to measuring independence: maximization of non-Gaussianity and minimization of mutual information. Most of the existing ICA algorithms are based on one of them. When the first approach is applied, the task for the algorithm is to modify the components in such a way to obtain the source signals of strong non-Gaussian distribution (the assumption is: the stronger non-Gaussianity, the stronger independence [18]). In other words, the distributions of the mixture signals have to be more Gaussian than the source signals. This approach utilizes various measures of non-Gaussianity, like: kurtosis, negentropy, approximations of negentropy and others [19].

Mutual information, utilized in the second approach, informs how much information about the variable a can be gained from the information about the variable b. Since smaller value of mutual information means that more information about a given system is stored in the variables [18], ICA algorithms based on mutual information approach minimize the mutual information of the system outputs [19].

2.1 FastICA - Deflation Approach

The FastICA algorithm, proposed by Hyvärinen and Oja, is an iterative method to find local maxima of a defined cost function [18-19], [3]. The purpose of this algorithm is to find the matrix of weights w such that the projection $(w^T x)$ maximizes non-Gaussianity [3], [19]. As a measure for non-Gaussianity, simple estimation of negentropy based on the maximum entropy principle is used [18-19]:

$$J(y) \propto [E\{G(y)\} - E\{G(v)\}]^2 \tag{4}$$

where: y – standardized non-Gaussian random variable, v – standardized random variable with Gaussian distribution, $G(.)$ - any non-quadratic function.

There are two classes of FastICA algorithms, the deflation algorithms (called also one-unit algorithms) and the symmetric algorithms [20]. In the deflation approach, the independent components (ICs) are extracted sequentially, one by one. The algorithm can be summarized as follows [19], [21]:

1. Choose an initial vector w (e.g. random)
2. Do steps 3-6
3. $w^+ = E\{xg(w^+x)\} - E\{g'(w^Tx)\}w$
4. $w = \frac{w^+}{||w^+||}$
5. Do the Gram-Schmidt orthogonalization:

$$w_{p+1} = w_{p+1} - \sum_{j=1}^{p} w_{p+1}^T \, w_j w_j$$

$$w_{p+1} = \frac{w_{p+1}}{\sqrt{w_{p+1}^T w_{p+1}}}$$

6. Stop if not converged

Gram-Schmidt procedure, used in the algorithm, prevents different vectors from matrix w from converging to the same maxima [19]. The order, in which the independent components are extracted, depends on the initial value of w.

2.2 FastICA - Symmetric Approach

The only difference between deflation approach and symmetric approach is the procedure of weights calculation. While in deflation approach vectors of weights are calculated one by one, in symmetric approach the estimation of all components (all weights vectors) proceeds in parallel [19-20]. Instead of Gram-Schmidt procedure, the following formula is used in the orthogonalization step:

$$w = (ww^T)^{-1/2}w \tag{5}$$

where w is the matrix of weights vectors $(w_1, \dots, w_n)^T$. The square root of ww^T is obtained from the eigenvalue decomposition of $ww^T = QDQ^T$ as [22]:

$$(ww^T)^{-1/2} = QD^{-1/2}Q^T \tag{6}$$

where Q is the matrix of eigenvectors and D is the diagonal matrix of eigenvalues.

The algorithm is performed until the stop condition (e.g. given by 7 [20]) is met:

$$1 - \min\left(abs\left(diag(w^T w_{old})\right)\right) < \varepsilon \tag{7}$$

where ε is a chosen constant.

2.3 Infomax

Infomax algorithm is based on the general optimization principle for neural networks and other processing systems described by Linsker in 1987 [23]. In general this principle says that a function that maps a set of input values *a* to a set of output values *b* should be chosen or learned so as to maximize the average Shannon mutual information between *a* and *b*. The ICA algorithm utilizing this principle was first proposed in 1995 by Bell and Sejnowski [24] and then in 1997 optimized by Amari [19], [21].

Infomax algorithm for calculating independent components is based on the maximization of the output entropy of a neural network with non-linear outputs [19]. The most essential parameter of this algorithm is a learning rate which does not need to be constant over time and which should give a good compromise between speed of learning and estimation precision [19], [25]. The weights of this neural network are updated according to the following formula [18], [21], [26]:

$$w_{k+1} = w_k + \mu_k[I - 2g(y_k)y_k^T]w_k \tag{8}$$

where: y – matrix of source estimation (y=Wx); k – number of iteration; I – the identity matrix; μ_k – learning rate which may depend on k; $g(.)$ – a nonlinear function.

Mostly a classic logistic function is used as a nonlinear function g [26]:

$$g(y) = \frac{1}{1+e^{-y}}\,, \tag{9}$$

however, sometimes also its extended version is applied:

$$g(y) = y \pm \tanh(y) \tag{10}$$

Using (9), the Infomax algorithm can be summarized as follows [21]:

```
1. x = perm(sources);
2. $y = w \times x$
3. $g = \frac{1}{1+e^{-y_k}}$
4. $gu = g \times y^T$
5. $gu = I - 2 \times gu$
6. $w_{k+1} = w_k + \mu_k \times gu \times w_k$
```

where $perm$ is random permutation.

3 Experimental Settings

The comparison of ICA algorithms described in Section 2 was carried out with a data set submitted to the second BCI Competition (data set III – motor imaginary) by Department of Medical Informatics, Institute for Biomedical Engineering, Graz University of Technology [27]. The data set was recorded from a normal subject (female, 25y) whose task was to control the movements of a feedback bar by means of imagery movements of the left and right hand. Cues informing about the direction in which the feedback bar should be moved were displayed on a screen in the form of the left and right arrows. The order of left and right cues was random. The experiment consisted of 280 trials, each trial lasted 9 seconds. The first 2s was quiet, at t=2s an acoustic stimulus was generated and a cross "+" was displayed for 1s; then at t=3s, an arrow (left or right) was displayed as a cue. The EEG signals were measured over three bipolar EEG channels (C3, Cz and C4), sampled with 128Hz and preliminary filtered between 0.5 and 30Hz. The whole data set, containing data from 280 trials, was then divided into two equal subsets – the first one was intended for classifier training and the second intended for external classifier test. Since only data from the first subset was published with target values (1 - left hand, 2- right hand), only this subset could be used in the research.

In the preprocessing step, the data from the original data set was transformed according to the algorithms described in Section 2. After performing this step, three different sets of signals were obtained:

1. Set of components obtained with FastICA - deflation approach algorithm.
2. Set of components obtained with FastICA - symmetric approach algorithm.
3. Set of components obtained with Infomax algorithm.

All these sets of components, together with the fourth set, composed of original signals from the channels C3, Cz and C4 were used in the experiments.

The data from each set of components, was transformed to a set of frequency band power features. The signal power was calculated separately for:

1. 12 frequency bands: alpha band (8-13Hz) and five sub-bands of alpha band (8-9Hz; 9-10Hz; 10-11Hz; 11-12Hz; 12-13Hz); beta band (13-30Hz) and also five sub-bands of beta band (13-17Hz; 17-20Hz; 20-23Hz; 23-26Hz; 26-30Hz),
2. each of 7 seconds of the trial (data from the first and second seconds of the recordings of each trial were discarded because they covered the period before the clue presentation),
3. each of 3 canals (C3, Cz, C4).

In this way 252 band power features were obtained per each of four sets of components. Taking into account a very small number of trials equal to 140, the number of features had to be significantly reduced before the classification step. In fact with 140 trials, no more than several features should be used without the threat of overfitting. According to Raudys and Jain, at least 10 times more training data per class than the features should be gathered to train the classifier correctly [28].

In order to reduce the number of features, a genetic algorithm, described in details in [8] was used. Some basic features of this algorithm are as follows:

1. An individual is composed of the number of genes equal to the critical number of features, given by the user or calculated automatically in terms of number of observations, number of classes and classifier type.
2. Each gene can take an integer value from the interval {0,1...*F*}, where *F* denotes the dimension of the feature set.
3. The basic genetic operation in the algorithm is a very aggressive mutation. The aggressive mutation means that not only each individual in the population is mutated, but also each gene of each individual.
4. Since, after the mutation a lot of new individuals is born, the selection step is performed after the reproduction step. The selection is made from the population composed of parent individuals and their mutated children.
5. The fitness function is pure classifier accuracy.

As it was stated above, a fitness function of the algorithm used for feature selection was a classification accuracy. A classifier was built per each individual of each generation. Input features introduced to each classifier were encoded in succeeding genes of the evaluated individual.

A linear SVM method was used in the classification process. The classification threshold was set to 0.5 and hence, all classifier results greater than 0.5 were classified as class "2" (right hand) and results smaller or equal to 0.5 were classified as class "1" (left hand). The classifiers accuracy was tested with 10-fold cross-validation. The final accuracy measure of a given feature set was the mean value calculated on the basis of classification accuracy obtained for all validation sets. The accuracy of one validation set was calculated according to the following equation:

$$A_k = \frac{R_k}{U_k} \tag{11}$$

where: A_k - accuracy of k validation subset (k=1...10), R_k - number of properly classified cases from k validation subset, U_k - number of all cases in k validation subset.

4 Results

The experiments were performed with the application prepared for the Matlab 7.12.0 environment. All analyzed preprocessing algorithms, it is: FastICA-deflation, FastICA-symmetric and Infomax, were implemented according to the general schemes given in Section 2. The learning rate in Infomax algorithm was variable in time. In three first iterations it was equal to: 0.01, 0.001 and 0.0001, respectively and in the remaining 500 iterations it was equal to 0.005.

The main aim of the experiments described in the paper was to find out whether the preprocessing with ICA has an essential influence on the classification accuracy. In order to answer this question, the classification accuracy calculated over raw signals and signals preprocessed with the analyzed ICA algorithms had to be compared. The comparison was carried out for different number of input features introduced to the classifier (from one to six). To perform the analysis, six genetic algorithms were prepared. The first one processed individuals composed of only one gene, second processed individuals composed of two genes and so forth up to the last one which processed individuals composed of six genes. Since a genetic algorithm is a heuristic optimization method which gives only sub-optimal solutions, the algorithm was run five times for each from the given six settings. Each from these 120 algorithms (six features, 4 sets of input signals, 5 runs) processed data by 30 generations. After each generation of each genetic algorithm, the individual of the highest value of the fitness function was stored in a table. At the end of the experiments, the results of the best individuals were averaged separately for each set of signals and for each number of input features. The average values of fitness function (classification accuracy) of the best individuals, together with standard deviations are presented in Table 1.

Table 1. The average values of fitness function (classification accuracy) and standard deviation calculated over the best individuals obtained for different sets of signals and different numbers of input features

	Raw data		Deflation		Symmetric		Infomax	
No. of features	Accuracy [%]	std	Accuracy [%]	std	Accuracy [%]	std	Accuracy [%]	std
1	76.71	0.64	75.14	0.53	75.00	0.45	75.86	0.83
2	82.57	2.7	82.71	1.31	82.43	1.32	82.29	0.29
3	89.00	0.81	88.43	0.53	87.86	1.36	88.57	1.43
4	91.00	1.08	90.86	0.53	90.71	0.9	91.71	0.73
5	92.71	1.06	92.14	1.01	92.57	0.35	92.43	0.57
6	93.14	1.08	92.29	1.53	93.86	0.73	94.57	0.73

A much more compact comparison of results obtained with ICA algorithms and results obtained with raw signals is given in Fig. 1. Each sub-figure of Fig. 1 presents the comparison of the results obtained with raw signals with the results obtained with signals preprocessed by one of the analyzed ICA algorithms (Fig. 1a - FastICA-deflation, Fig. 1b - FastICA-symmetric, Fig. 1c - Infomax).

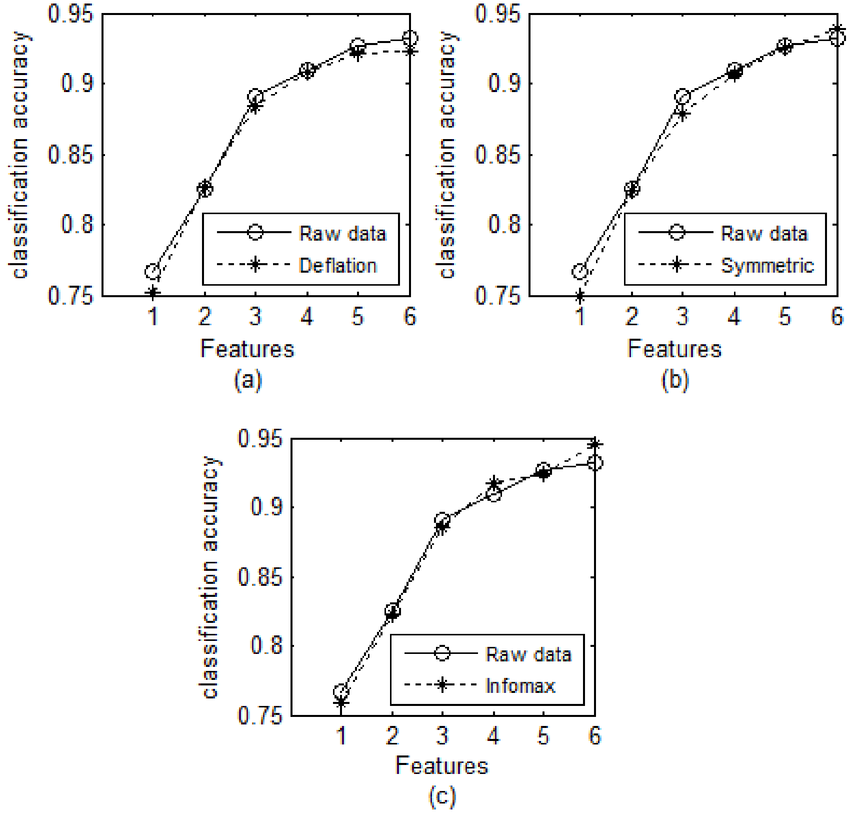


Fig. 1. The average classification accuracy calculated over five runs for each of the six genetic algorithms. Each sub-figure presents the results obtained with raw signals and the results obtained with signals preprocessed by one of the analyzed ICA algorithms (Fig. 1a - FastICA-deflation, Fig. 1b - FastICA-symmetric, Fig. 1c - Infomax).

5 Discussion

In Section 1 three questions regarding the influence of ICA transformation on the classification precision were posed. To answer the first question, about the overall increase in the classification accuracy, the average value of the classification accuracy obtained with all three ICA algorithms was calculated. The result was surprising because the average classification precision calculated for signals preprocessed with ICA algorithms was equal to 87.2% which was slightly lower than the average classification precision calculated for raw signals (87.5%).

In order to answer the second question about the influence of the choice of a specific ICA algorithm on the classification precision, the comparison between algorithms had to be done. This time the results were more consistent with the theory and other analysis. The highest average classification precision for all six genetic algorithms was obtained with Infomax algorithm (87.6%), a slightly smaller precision was obtained with FastICA-symmetric algorithm (87.1) and the smallest precision was obtained with FastICA-deflation (86.9%) (Fig. 2). Looking closer at the average precisions of all three algorithms, one can easily notice that the differences between the performance of ICA algorithms are very small, in fact they are so small that they are not regarded as significant by any reasonable statistical test. Therefore, it is difficult to state that one algorithm is better than the others only on the basis of the classification precision. However, when standard deviations of results are taken into consideration, the conclusion is quite different. The standard deviation of the results gathered in Tab. 1 was on average by 33% lower in case of Infomax (0.8%) than in case of the raw signals (1.2%). This means that the Infomax algorithm gave more stable results and that these results are more reliable than results calculated over raw signals. Also the uncertainty of the classification precision calculated over signals preprocessed with two other ICA algorithms was significantly smaller than in case of raw data (std of FastICA-deflation - 0.9% and std of FastICA-symmetric - 0.9%). Hence, taking into account the standard deviation of results a conclusion of a practical usefulness of ICA transformation should be drawn.

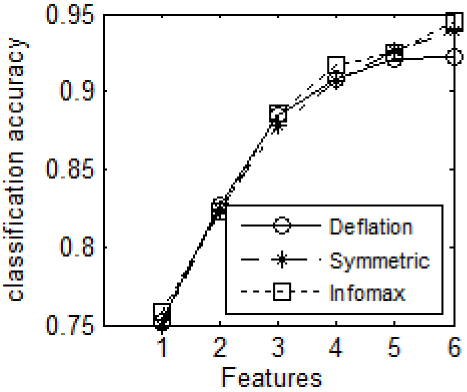


Fig. 2. The comparison of classification accuracy calculated over signals preprocessed with the analyzed ICA algorithms for different numbers of input features

And finally, the third question posed in Section 1 was about a correlation between difference in the classification precision obtained with signals preprocessed with analysed ICA algorithms and the number of input features used in the classification process. This question is difficult to address because of very small, in fact non-significant differences in classification accuracy obtained over signals preprocessed with all three algorithms at each step of the experiment. On average Infomax gave better results but for different number of input features different algorithms exhibits slightly better performance: FastICA-deflation gave the best result for two features, FastICA-symmetric for five features and Infomax for one, three, four and six features.

6 Conclusion

The classification accuracy is a very important factor of a successful BCI because wrong classification results in executing improper action/command. Misclassification is not allowed, especially when the BCI is used as a system controlling actions of real devices.

The overall goal of this paper was to determine whether the preprocessing with ICA results in increasing the classification accuracy. Since the classification results were very similar with preprocessing and without it, it is very difficult to answer this question decidedly. Undoubtedly, the application of ICA increases the complexity of the whole BCI system so, on the one hand, taking into account only the lack of improvements in the accuracy, the answer should be "no". However, on the other hand, after applying ICA the uncertainty of classification results decreased rapidly which is a very important fact pro ICA application. Hence, before the question posed in the paper will be answer definitely, more experiments have to be done, experiments which will allow to find out which components survive the selection process, whether these components are the same in all ICA algorithms and lastly what is their distribution over the head model.

References

1. Wolpaw, J.R., Birbaumer, N., McFarland, D.J., Pfurtscheller, G., Vaughan, T.M.: Brain–computer interfaces for communication and control. Clinical Neurophysiology 113, 767–791 (2002)
2. Wang, Y., Shangkai, G., Xiaorong, G.: Common Spatial Pattern Method for Channel Selection in Motor Imagery Based Brain-Computer Interface. In: Proceedings of the 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference (2005)
3. Oja, E., Yuan, Z.: The FastICA Algorithm Revisited: Convergence Analysis. IEEE Transaction on Neural Networks 17(6), 1370–1381 (2006)
4. Augustyniak, P.: Adaptive Wavelet Discrimination of Muscular Noise in the ECG. In: Proc. Computers in Cardiology (IEEE-EMB), vol. 33, pp. 481–484 (2006)
5. Yang, R., Song, A., Xu, B.: Feature Extraction of Motor Imagery EEG Based on Wavelet Transform and Higher-Order Statistics. International Journal of Wavelets, Multiresolution and Information Processing 8(3), 373–384 (2010)
6. Müller-Putz, G.R., Kaiser, V., Solis-Escalante, T., Pfurtscheller, G.: Fast set-up asynchronous brain-switch based on detection of foot motor imagery in 1-channel EEG. Medical & Biological Engineering & Computing 48(3), 229–233 (2010)
7. Koprinska, I.: Feature Selection for Brain-Computer Interfaces. In: Theeramunkong, T., Nattee, C., Adeodato, P.J.L., Chawla, N., Christen, P., Lenca, P., Poon, J., Williams, G. (eds.) PAKDD Workshops 2009. LNCS (LNAI), vol. 5669, pp. 106–117. Springer, Heidelberg (2010)
8. Rejer, I.: Genetic Algorithms in EEG Feature Selection for the Classification of Movements of the Left and Right Hand. In: Burduk, R., Jackowski, K., Kurzynski, M., Wozniak, M., Zolnierek, A. (eds.) CORES 2013. AISC, vol. 226, pp. 579–589. Springer, Heidelberg (2013)

9. Burduk, R.: Recognition Task with Feature Selection and Weighted Majority Voting Based on Interval-Valued Fuzzy Sets. In: Nguyen, N.-T., Hoang, K., Jedrzejowicz, P. (eds.) ICCCI 2012, Part I. LNCS, vol. 7653, pp. 204–209. Springer, Heidelberg (2012)
10. Pfurtscheller, G., Neuper, C., Schlögl, A., Lugger, K.: Separability of EEG Singals Recorded During Right and Left Motor Imagery Using Adaptive Autoregressive Parameters. IEEE Trans. on Rehabilitation Engineering 6(3) (September 1998)
11. Bobrowski, L., Topczewska, M.: Separable Linearization of Learning Sets by Ranked Layer of Radial Binary Classifiers. In: Burduk, R., Jackowski, K., Kurzynski, M., Wozniak, M., Zolnierek, A. (eds.) CORES 2013. AISC, vol. 226, pp. 131–140. Springer, Heidelberg (2013)
12. Wozniak, M., Krawczyk, B.: Combined classifier based on feature space partitioning. Applied Mathematics and Computer Science 22(4), 855–866 (2012)
13. Chen, X., Wang, L., Xu, Y.: A symmetric orthogonal FastICA algorithm and applications in EEG. In: Fifth International Conference on Natural Computation, ICNC 2009, vol. 2 (2009)
14. Peterson, D.A., Knight, J.N., Kirby, M.J., Anderson, C.W., Thaut, M.H.: Feature Selection and Blind Source Separation in an EEG-Based Brain-Computer Interface. EURASIP Journal on Applied Signal Processing 19, 3128–3140 (2005)
15. Boord, P., Craig, A., Tran, Y., Nguyen, H.: Discrimination of left and right leg motor imagery for brain–computer interfaces. Medical & Biological Engineering & Computing 48(4), 343–350 (2010)
16. Park, H.M., Oh, S.H., Lee, S.Y.: A modified infomax algorithm for blind signal separation. Science Direct. Neurocomputing 70, 229–240 (2006)
17. Stone, J.V.: Independent component analysis: an introduction. TRENDS in Cognitive Sciences 6(2), 59–64 (2002)
18. Langlois, D., Chartier, S., Gosselin, D.: An Introduction to Independent Component Analysis: InfoMax and FastICA algorithms. Tutorials in Quantitative Methods for Psychology 6, 31–38 (2010)
19. Hyvärinen, A., Oja, E.: Independent Component Analysis: Algorithms and Applications. Neural Networks 13(4-5), 411–430 (2000)
20. Tichavský, P., Koldovský, Z., Oja, E.: Performance Analysis of the FastICA Algorithm and Cramér–Rao Bounds for Linear Independent Component Analysis. IEEE Transactions on Signal Processing 54(4), 1189–1203 (2006)
21. Naik, G., Kumar, D.: An Overview of Independent Component Analysis and Its Applications. Informatykica, 63–81 (2011)
22. Fan, C., Wang, B., Ju, H.: A New FastICA Algorithm with Symmetric Orthogonalization, pp. 2058–2061. IEEE (2006)
23. Linsker, R.: Self-Organization in a Perceptual Network. Computer 21, 105–117 (1988)
24. Bell, A.J., Sejnowski, T.J.: An information-maximisation approach to blind separation and blind deconvolution. Journal Neural Computation 7, 1129–1159 (1995)
25. Raimondo, F., Kamienkowski, J.E., Sigman, M., Slezak, D.F.: CUDAICA: GPU Optimization of Infomax-ICA EEG Analysis. Computational Intelligence and Neuroscience 2012 (2012)
26. Karhunen, J.: Neural approaches to independent component analysis and source separation. In: Proc 4th European Symposium on Artificial Neural Networks, ESANN 1996 (1996)
27. Data set III, II BCI Competition, motor imaginary, `http://bbci.de/competition/ii/index.html`
28. Raudys, S.J., Jain, A.K.: Small sample size effects in statistical pattern recognition: Recommendations for practitioners. IEEE Transactions on Pattern Analysis and Machine Intelligence 13(3), 252–264 (1991)

An Adequate Representation of Medical Data Based on Partial Set Approximation

Zoltán Ernő Csajbók[1], Tamás Mihálydeák[2], and József Ködmön[1]

[1] Department of Health Informatics, Faculty of Health, University of Debrecen,
Sóstói út 2-4, H-4400 Nyíregyháza, Hungary
{csajbok.zoltan,kodmon.jozsef}@foh.unideb.hu

[2] Department of Computer Science, Faculty of Informatics, University of Debrecen
Kassai út 26, H-4028 Debrecen, Hungary
mihalydeak.tamas@inf.unideb.hu

Abstract. Computer aided medical diagnosis and treatment require an adequate representation of uncertain or imperfect medical data. There are many approaches dealing with such type of data. Pawlak proposed a new method called rough set theory. In this paper, beyond classical and recent methods, the authors propose a basically new approach. It relies on a generalization of rough set theory, namely, the partial covering of the universe of objects. It adequately reflects the partial nature of real–life problems. This new approach called the partial approximation of sets is presented as well as its medical informatics application is demonstrated.

Keywords: Rough set theory, partial approximation of sets, tool–based approximation framework, thyroid disease diagnosis.

1 Introduction

Uncertainty, imprecision or incompleteness are important aspects of patients' medical data. Therefore, it is an inevitable challenge of medical informatics how imperfect medical data can be represented, moreover, how appropriate and powerful approximative inference methods can be provided. There are two classical approaches, Bayesian statistics [1, 25] and Dempster-Shafer theory of evidence [2, 28]. In the middle of 1960s, Zadeh proposed fuzzy set theory [22, 23, 30]. The medical expert system MYCIN uses the so-called certainty factor (CF) model [4]. Recently knowledge-based systems and other artificial intelligence methods have also been used in the medical diagnosis and treatment [17, 18].

In the early 1980s, Pawlak proposed a new method called *rough set theory* [19, 20]. It can be viewed as a new mathematical approach to manage imperfect or vague knowledge [15, 12]. It has extensive applications, among others, in artificial intelligence, cognitive sciences and medicine [11, 26]. Its starting point is a nonempty finite set U of objects and an equivalence relation ε on U [20]. The equivalence classes are called ε-elementary sets. They can be viewed as sets of indiscernible objects characterized by the same available information about them [21, 24]. Definable sets are any unions of ε-elementary sets.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 120–128, 2013.

Then, any set $S \subseteq U$ can be naturally approximated by the so–called lower and upper ε-approximations of S. The former is the union of all ε-elementary sets which are the subsets of S, whereas the latter is the union of all ε-elementary sets which have a nonempty intersection with S.

In rough set theory, the equivalence classes are pairwise disjoint and cover the universe. Giving up the disjoint property but retaining the covering, a natural generalization of rough set theory is obtained. It is called covering–based rough set theory [3, 29, 31–33]. The partial nature of real–life problems, however, requires working out partial approximation schemes. Thus, not only the pairwise disjoint property but also the covering of the universe are given up. This basically new approach is referred to as *partial approximation of sets* [5, 6, 10].

Furthermore, based on partial approximation of sets, a general tool-based approximation framework can be built up [9]. It is a general model which has applications in computer security [7, 8]. In this paper, we apply it to representing imperfect medical data.

The paper is organized as follows. Having reviewed the fundamental notions of partial approximation of sets in Section 2, Section 3 presents the general tool–based approximation framework. Section 4 shows a simplified thyroid disease diagnosis example in order to demonstrate how our approach can be applied. Finally, in Section 5, we conclude the paper.

2 Fundamental Notions of Partial Approximation of Sets

A general set–theoretic approximation framework has four components:

- the *domain* of approximations whose sets are approximated;
- some distinguished sets of the domain as the *basis* of approximations;
- *definable sets* deriving from base sets in some way or other as possible approximations of sets of the domain;
- an *approximation pair* determining the lower and upper approximations of sets using definable sets.

Definable sets represent our available knowledge about the domain. They can be thought of as tools of the approximation process and simply called *tools*. In particular, base sets can be viewed as *primary tools*, definable sets as *derived tools*. The way of getting derived tools from primary tools shows how primary tools are used. An approximation pair prescribes the *utilization* of primary and derived tools in a whole approximation process.

Let U be a *finite* nonempty set.

Let $\mathfrak{B} \subseteq 2^U$ be a nonempty family of nonempty subsets of U.[1] $\mathfrak{B}$ is called the *base system*, its members are the *base sets*. Let $\mathfrak{D}_{\mathfrak{B}} \subseteq 2^U$ denote an extension of $\mathfrak{B}$ in such a way that 1) $\mathfrak{B} \subseteq \mathfrak{D}_{\mathfrak{B}}$, and 2) $\emptyset \in \mathfrak{D}_{\mathfrak{B}}$. The members of $\mathfrak{D}_{\mathfrak{B}}$ are called *definable sets*. It is not necessary that $\bigcup \mathfrak{D}_{\mathfrak{B}} = U$. An approximation framework is *total* if $\bigcup \mathfrak{D}_{\mathfrak{B}} = U$, and it is *partial* otherwise.

[1] 2^U denotes the power set of U.

It is a natural assumption that $\mathfrak{D}_{\mathfrak{B}}$ is obtained (derived) form $\mathfrak{B}$ by some sort of set type transformations (for the most important cases, see [5]). In order to build a generalized Pawlakian partial approximation framework, we define $\mathfrak{D}_{\mathfrak{B}}$ with the following definition:

1. $\emptyset \in \mathfrak{D}_{\mathfrak{B}}$;
2. $\mathfrak{B} \subseteq \mathfrak{D}_{\mathfrak{B}}$;
3. if $\mathfrak{B}' \subseteq \mathfrak{B}$, $\bigcup \mathfrak{B}', \bigcap \mathfrak{B}' \in \mathfrak{D}_{\mathfrak{B}}$.

Next, let $\langle \mathsf{l}, \mathsf{u} \rangle$ be an ordered pair of maps $\mathsf{l}, \mathsf{u} : 2^U \to 2^U$ on $(2^U, \subseteq)$. Of course, their intended meaning is to express the lower and upper approximations of any sets with the help of the beforehand given definable sets as tools. Hence, $\langle \mathsf{l}, \mathsf{u} \rangle$ is called an *approximation pair*.

Here, for any $S \subseteq U$, the lower and upper approximations are a possible generalization of Pawlakian lower and upper ε-approximations:

- lower approximation $\mathsf{l}(S)$ is the union of all definable sets which are the subsets of S:
$$\mathsf{l}(S) = \bigcup \{D \in \mathfrak{D}_{\mathfrak{B}} \mid D \subseteq S\} \in \mathfrak{D}_{\mathfrak{B}};$$
- upper approximation $\mathsf{u}(S)$ is the union of all definable sets which have a nonempty intersection with S:
$$\mathsf{u}(S) = \bigcup \{D \in \mathfrak{D}_{\mathfrak{B}} \mid D \cap S \neq \emptyset\} \in \mathfrak{D}_{\mathfrak{B}}.$$

The universe U, the base system $\mathfrak{B}$, the set $\mathfrak{D}_{\mathfrak{B}}$ of definable sets, and the approximation pair $\langle \mathsf{l}, \mathsf{u} \rangle$ together are called the *partial approximation space* and denoted by the ordered 5-tuple $\langle 2^U, \mathfrak{B}, \mathfrak{D}_{\mathfrak{B}}, \mathsf{l}_{\mathfrak{B}}, \mathsf{u}_{\mathfrak{B}} \rangle$.

Possible interpretations of lower and upper approximations of a set $S \subseteq U$ are the following [20]:

1. $\mathsf{l}(S)$ is the set of all members of U which can *certainly* be classified as belonging to S with respect to $\mathfrak{B}$ (*positive region*);
2. $\mathsf{u}(S)$ is the set of all members of U which can *possibly* be classified as belonging to S with respect to $\mathfrak{B}$;
3. $U \setminus \mathsf{u}(S)$ is the set of all members of U which can *certainly* be classified as not belonging to S with respect to $\mathfrak{B}$ (*negative region*).

3 Elements of Tool-Based Approximation Framework

Tool-based approximation framework is defined relying on partial approximation spaces. For more details, see [9].

Let $A^+, A^- \subseteq U$ be two nonempty subsets of U in such a way that $A^+ \cap A^- = \emptyset$. A^+ and A^- are called the *positive* and *negative reference set*, respectively. The adjectives "positive" and "negative" claim nothing else but that the sets A^+ and A^- are *well separated.*

Furthermore, let $\mathfrak{T}^+, \mathfrak{T}^- \subseteq 2^U$ be two nonempty finite families of nonempty subsets of U, where $\mathfrak{T}^+$ and $\mathfrak{T}^-$ are not necessarily disjoint. The members of $\mathfrak{T}^+$ are called *positive* or $\mathfrak{T}^+$-tools, whereas the members of $\mathfrak{T}^-$ are called *negative* or $\mathfrak{T}^-$-tools.

Requirements for positive and negative tools are the following:

- sets in $\mathfrak{T}^+$ ($\mathfrak{T}^-$) are not necessarily pairwise disjoint;
- neither $\bigcup \mathfrak{T}^+$ nor $\bigcup \mathfrak{T}^-$ covers U necessarily;
- for each subset $T^+ \in \mathfrak{T}^+$ ($T^- \in \mathfrak{T}^-$), it is *easy* to decide whether a member of U belongs to T^+ (T^-) or does not.

Mutual relationship between positive/negative tools and positive/negative reference sets may provide useful information. They can be obtained with the help of partial approximation of positive/negative reference sets within the approximation spaces $\langle 2^U, \mathfrak{T}^+, \mathfrak{D}_{\mathfrak{T}^+}, \mathsf{l}_{\mathfrak{T}^+}, \mathsf{u}_{\mathfrak{T}^+} \rangle$, $\langle 2^U, \mathfrak{T}^-, \mathfrak{D}_{\mathfrak{T}^-}, \mathsf{l}_{\mathfrak{T}^-}, \mathsf{u}_{\mathfrak{T}^-} \rangle$, respectively.

4 Thyroid Disease Diagnosis Support

Let us suppose that we study the symptoms of the thyroid dysfunction [14]. Clinical symptoms which manifest thyroid dysfunction progress slowly they are often nonspecific and could represent other not thyroid disorders. Hence, thyroid function diagnosis via clinical symptoms is an important but inexact classification problem [27, 16].

For the sake of simplicity, we deal with only hypothyroidism and hyperthyroidism thyroid disorders [13]. The thyroid gland produces thyroid hormones, thyroxine (T4) and triiodothyronine (T3). Hyperthyroidism occurs when the thyroid is "overactive", i.e., releases too much hormones, whereas hypothyroidism takes place when the thyroid is "underactive", i.e., does not produce enough hormones.

We have at our disposal an information table (Table 1) containing clinical symptoms which may indicate whether hypothyroidism or hyperthyroidism, or else neither of the two. There are, of course, more symptoms of hypothyroidism and hyperthyroidism, but we have simplified the example here for illustrative purposes.

Table 1. Clinical symptoms of thyroid dysfunction and diagnosis based on test results

No.	Weight change	Edema	Tachycardia	Increased sweating	Affection	Hypothyroidism	Hyperthyroidism
P_1	not change	no	no	no	normal	**no**	**no**
P_2	gain	no	no	no	normal	**yes**	**no**
P_3	gain	no	yes	no	normal	**yes**	**no**
P_4	loss	no	yes	yes	normal	**no**	**yes**
P_5	not change	yes	no	yes	nervousness	**no**	**yes**

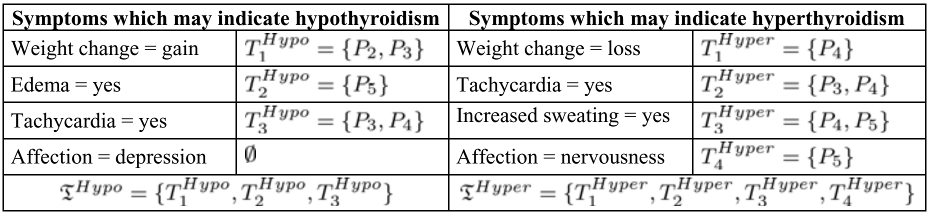
The universe of objects is the set of patients, i.e., $U = \{P_1, P_2, P_3, P_4, P_5\}$.

Hypothyroidism and hyperthyroidism can be accurately diagnosed with laboratory tests. Last two columns in Table 1 are based on these results. Reference sets are formed on them: the "positive" reference set is: $A^{Hypo} = \{P_2, P_3\}$, and the "negative" reference set is: $A^{Hyper} = \{P_4, P_5\}$.

Note that $A^{Hypo} \cap A^{Hyper} = \emptyset$.

Table 2 contains clinical symptoms which may indicate hypothyroidism ("positive tools") and hyperthyroidism ("negative tools"). Notice that $\mathfrak{T}^{Hypo}$ and $\mathfrak{T}^{Hyper}$ do not necessarily cover the universe ($\bigcup \mathfrak{T}^{Hypo}, \bigcup \mathfrak{T}^{Hyper} \subseteq U$) and their members are not pairwise disjoint, respectively.

Table 2. Symptoms which may indicate hypothyroidism/hyperthyroidism ("positive"/ "negative" tools)



Symptoms which may indicate hypothyroidism		Symptoms which may indicate hyperthyroidism	
Weight change = gain	$T_1^{Hypo} = \{P_2, P_3\}$	Weight change = loss	$T_1^{Hyper} = \{P_4\}$
Edema = yes	$T_2^{Hypo} = \{P_5\}$	Tachycardia = yes	$T_2^{Hyper} = \{P_3, P_4\}$
Tachycardia = yes	$T_3^{Hypo} = \{P_3, P_4\}$	Increased sweating = yes	$T_3^{Hyper} = \{P_4, P_5\}$
Affection = depression	$\emptyset$	Affection = nervousness	$T_4^{Hyper} = \{P_5\}$
$\mathfrak{T}^{Hypo} = \{T_1^{Hypo}, T_2^{Hypo}, T_3^{Hypo}\}$		$\mathfrak{T}^{Hyper} = \{T_1^{Hyper}, T_2^{Hyper}, T_3^{Hyper}, T_4^{Hyper}\}$	

Primary tools are (see Table 2):

- $\mathfrak{T}^{Hypo} = \{T_1^{Hypo}, T_2^{Hypo}, T_3^{Hypo}\} = \{\{P_2, P_3\}, \{P_5\}, \{P_3, P_4\}\}$;
- $\mathfrak{T}^{Hyper} = \{T_1^{Hyper}, T_2^{Hyper}, T_3^{Hyper}, T_4^{Hyper}\} = \{\{P_4\}, \{P_3, P_4\}, \{P_4, P_5\}, \{P_5\}\}$.

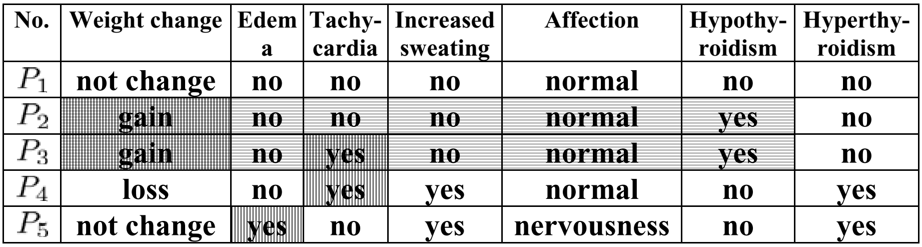
Derived tools $\mathfrak{D}_{\mathfrak{T}^{Hypo}}$ and $\mathfrak{D}_{\mathfrak{T}^{Hyper}}$ are formed by unions and intersections of any subsets of $\mathfrak{T}^{Hypo}$ and $\mathfrak{T}^{Hyper}$, respectively:

$$\begin{aligned}
\mathfrak{D}_{\mathfrak{T}^{Hypo}} = \{&\emptyset, T_1^{Hypo}, T_2^{Hypo}, T_3^{Hypo},\\
&\{P_2, P_3, P_5\}, \{P_2, P_3, P_4\}, \{P_3, P_4, P_5\}, \{P_2, P_3, P_4, P_5\}, \{P_3\}\};\\
\mathfrak{D}_{\mathfrak{T}^{Hyper}} = \{&\emptyset, T_1^{Hyper}, T_2^{Hyper}, T_3^{Hyper}, T_4^{Hyper},\\
&\{P_3, P_4, P_5\}, \{P_4\}, \{P_5\}\}.
\end{aligned}$$

Sample evaluations can be carried out as follows.

Let us take, e.g., the group of patients having hypothyroidism diagnosis: $A^{Hypo} = \{P_2, P_3\} \subseteq U$.

Table 3. Approximating hypothyroidism with clinical symptoms which may indicate hypothyroidism itself



No.	Weight change	Edema	Tachycardia	Increased sweating	Affection	Hypothyroidism	Hyperthyroidism
P_1	not change	no	no	no	normal	no	no
P_2	gain	no	no	no	normal	yes	no
P_3	gain	no	yes	no	normal	yes	no
P_4	loss	no	yes	yes	normal	no	yes
P_5	not change	yes	no	yes	nervousness	no	yes

The evaluation of patients having hypothyroidism with respect to the clinical symptoms which may indicate hypothyroidism, i.e., in the language of partial approximation framework, forming the lower and upper approximations of A^{Hypo} in the partial approximation space $\langle 2^U, \mathfrak{T}^{Hypo}, \mathfrak{D}_{\mathfrak{T}^{Hypo}}, \mathsf{l}_{\mathfrak{T}^{Hypo}}, \mathsf{u}_{\mathfrak{T}^{Hypo}} \rangle$ (see Table 3):

- $\mathsf{l}_{\mathfrak{T}^{Hypo}}(A^{Hypo}) = \mathsf{l}_{\mathfrak{T}^{Hypo}}(\{P_2, P_3\}) = T_1^{Hypo} \cup (T_1^{Hypo} \cap T_3^{Hypo}) = \{P_2, P_3\}$.
  Informally: patients having “weight gain” or “weight gain and tachycardia” can *certainly* be classified as suffering from hypothyroidism ***with respect to the clinical symptoms which may indicate hypothyroidism***.
- $\mathsf{u}_{\mathfrak{T}^{Hypo}}(A^{Hypo}) = \mathsf{u}_{\mathfrak{T}^{Hypo}}(\{P_2, P_3\}) = T_1^{Hypo} \cup T_3^{Hypo} \cup (T_1^{Hypo} \cap T_3^{Hypo}) = \{P_2, P_3, P_4\}$.
  Informally: patients having “weight gain” or “tachycardia” or “weight gain and tachycardia” can *possibly* be classified as suffering from hypothyroidism ***with respect to the clinical symptoms which may indicate hypothyroidism***.

Table 4. Approximating hypothyroidism with clinical symptoms which may indicate *hyperthyroidism*

No.	Weight change	Edema	Tachycardia	Increased sweating	Affection	Hypothyroidism	Hyperthyroidism
P_1	not change	no	no	no	normal	no	no
P_2	gain	no	no	no	normal	yes	no
P_3	gain	no	yes	no	normal	yes	no
P_4	loss	no	yes	yes	normal	no	yes
P_5	not change	yes	no	yes	nervousness	no	yes

The evaluation of patients having hypothyroidism with respect to the clinical symptoms which may indicate *hyperthyroidism*, i.e., in the language of partial approximation framework, forming the lower and upper approximations of A^{Hypo} in the partial approximation space $\langle 2^U, \mathfrak{T}^{Hyper}, \mathfrak{D}_{\mathfrak{T}^{Hyper}}, \mathsf{l}_{\mathfrak{T}^{Hyper}}, \mathsf{u}_{\mathfrak{T}^{Hyper}} \rangle$ (see Table 4) (it can be viewed as a “cross–evaluation”):

- $\mathsf{l}_{\mathfrak{T}^{Hyper}}(A^{Hypo}) = \mathsf{l}_{\mathfrak{T}^{Hyper}}(\{P_2, P_3\}) = \emptyset$.
  Informally: in accordance with the expectations, there are no symptoms under which patients can *certainly* be classified as suffering from hypothyroidism ***with respect to the clinical symptoms which may indicate hyperthyroidism***.
- $\mathsf{u}_{\mathfrak{T}^{Hyper}}(A^{Hypo}) = \mathsf{u}_{\mathfrak{T}^{Hyper}}(\{P_2, P_3\}) = T_2^{Hyper} = \{P_3, P_4\}$.
  Informally: in accordance with the nonspecific nature of thyroid dysfunction symptoms, there are symptoms under which patients can *possibly* be classified as suffering from hypothyroidism ***with respect to the clinical symptoms which may indicate hyperthyroidism***.

5 Conclusion and Future Work

In this paper, beyond classical and recent methods, the authors have proposed a basically new approach for handling imperfect medical data. Its novelty is inherent in the partiality of the applied approximation space. Partial approximation space is a generalization of Pawlakian rough set theory which adequately reflects partial nature of real–life problems. The proposed methods may serve as a basis of computer assistance systems in medicine.

The evaluation of evidence described above is of qualitative nature. Quantifying uncertainty is an important challenge in rough set theory and its generalizations. We will plan to make the model more useful in real applications, first of all, by means of quantifying uncertainty in partial approximation spaces.

Acknowledgements. The publication was supported by the TÁMOP–4.2.2.C–11/1/KONV–2012–0001 project. The project has been supported by the European Union, co–financed by the European Social Fund.

References

1. Ashby, D.: Bayesian statistics in medicine: a 25 year review. Statistics in Medicine 25(21), 3589–3631 (2006)
2. Beynon, M., Curry, B., Morgan, P.: The Dempster-Shafer theory of evidence: an alternative approach to multicriteria decision modelling. Omega 28(1), 37–50 (2000)
3. Bonikowski, Z., Bryniarski, E., Wybraniec-Skardowska, U.: Extensions and intensions in the rough set theory. Information Sciences 107(1-4), 149–167 (1998)
4. Buchanan, B.G., Shortliffe, E.H.: Rule Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project. The Addison-Wesley Series in Artificial Intelligence. Addison-Wesley Longman Publishing Co., Inc., Boston (1984)
5. Csajbók, Z., Mihálydeák, T.: A general set theoretic approximation framework. In: Greco, S., Bouchon-Meunier, B., Coletti, G., Fedrizzi, M., Matarazzo, B., Yager, R.R. (eds.) IPMU 2012, Part I. CCIS, vol. 297, pp. 604–612. Springer, Heidelberg (2012)
6. Csajbók, Z.E.: Approximation of sets based on partial covering. In: Peters, J.F., Skowron, A., Ramanna, S., Suraj, Z., Wang, X. (eds.) Transactions on Rough Sets XVI. LNCS, vol. 7736, pp. 144–220. Springer, Heidelberg (2013)
7. Csajbók, Z.: A security model for personal information security management based on partial approximative set theory. In: Ganzha, M., Paprzycki, M. (eds.) Proceedings of the International Multiconference on Computer Science and Information Technology (IMCSIT 2010), Wisła, Poland, October 18-20, vol. 5, pp. 839–845. Polskie Towarzystwo Informatyczne – IEEE Computer Society Press, Katowice (2010)
8. Csajbók, Z.: Simultaneous anomaly and misuse intrusion detections based on partial approximative set theory. In: Cotronis, Y., Danelutto, M., Papadopoulos, G.A. (eds.) Proceedings of PDP 2011, Ayia Napa, Cyprus, February 9-11, pp. 651–655. IEEE Computer Society Press, Los Alamitos (2011)

9. Csajbók, Z., Mihálydeák, T.: General tool-based approximation framework based on partial approximation of sets. In: Kuznetsov, S.O., Ślęzak, D., Hepting, D.H., Mirkin, B.G. (eds.) RSFDGrC 2011. LNCS (LNAI), vol. 6743, pp. 52–59. Springer, Heidelberg (2011)
10. Csajbók, Z., Mihálydeák, T.: Partial approximative set theory: A generalization of the rough set theory. International Journal of Computer Information Systems and Industrial Management Applications 4, 437–444 (2012)
11. Hassanien, A.E., Abraham, A., Peters, J.F., Schaefer, G.: Rough sets in medical informatics applications. In: Mehnen, J., Saad, M.K.A., Tiwari, A. (eds.) Applications of Soft Computing. AISC, vol. 58, pp. 23–30. Springer, Heidelberg (2009)
12. Keefe, R., Smith, P.: Introduction: Theories of vagueness. In: Keefe, R., Smith, P. (eds.) Vagueness: A Reader, pp. 1–57. MIT Press, Cambridge (1996)
13. Ladenson, P., Kim, M.: Thyroid. In: Goldman, L., Schafer, A.I. (eds.) Goldman's Cecil Medicine, ch. 233. Saunders Elsevier, Philadelphia (2011)
14. Ladenson, P.W., Singer, P.A., Ain, K.B., Bagchi, N., Bigos, S.T., Levy, E.G., Smith, S.A., Daniels, G.H., Cohen, H.D.: American thyroid association guidelines for detection of thyroid dysfunction. Arch. Intern. Med. 160(11), 1573–1575 (2000)
15. Orlowska, E.S.: Logic of vague concepts. Bulletin of the Section of Logic 11(3-4), 115–126 (1982)
16. Ozyılmaz, L., Yıldırım, T.: Diagnosis of thyroid disease using artificial neural network methods. In: Proceedings of ICONIP 2002, Singapore, pp. 2033–2036. Orchid Country Club (2002)
17. Pandey, B., Mishra, R.B.: Knowledge and intelligent computing system in medicine. Comp. in Bio. and Med. 39(3), 215–230 (2009)
18. Patel, V.L., Shortliffe, E.H., Stefanelli, M., Szolovits, P., Berthold, M.R., Bellazzi, R., Abu-Hanna, A.: The coming of age of artificial intelligence in medicine. Artificial Intelligence in Medicine 46(1), 5–17 (2009)
19. Pawlak, Z.: Rough sets. International Journal of Computer and Information Sciences 11(5), 341–356 (1982)
20. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. Kluwer Academic Publishers, Dordrecht (1991)
21. Pawlak, Z., Skowron, A.: Rough sets: Some extensions. Information Sciences 177, 28–40 (2007)
22. Seising, R.: Fuzzy sets in medicine - historical remarks. In: Valafar, F., Valafar, H. (eds.) METMBS, pp. 31–37. CSREA Press (2004)
23. Seising, R.: From vagueness in medical thought to the foundations of fuzzy reasoning in medical diagnosis. Artificial Intelligence in Medicine 38(3), 237–256 (2006)
24. Skowron, A.: Vague concepts: A rough-set approach. In: De Baets, B., De Caluwe, R., De Tré, G., Fodor, J., Kacprzyk, J., Zadrożny, S. (eds.) Proceedings of EUROFUSE 2004, pp. 480–493. Akademicka Oficyna Wydawnicza EXIT, Warszawa (2004)
25. Spiegelhalter, D.J., Abrams, K.R., Myles, J.P.: Bayesian Approaches to Clinical Trials and Health-Care Evaluation (Statistics in Practice). John Wiley & Sons (January 2004)
26. Straszecka, E.: Combining uncertainty and imprecision in models of medical diagnosis. Inf. Sci. 176(20), 3026–3059 (2006)
27. Temurtas, F.: A comparative study on thyroid disease diagnosis using neural networks. Expert Syst. Appl. 36(1), 944–949 (2009)
28. Yager, R.R., Kacprzyk, J., Fedrizzi, M. (eds.): Advances in the Dempster-Shafer theory of evidence. John Wiley & Sons, Inc., New York (1994)

29. Yao, Y.Y.: On generalizing rough set theory. In: Wang, G., Liu, Q., Yao, Y., Skowron, A. (eds.) RSFDGrC 2003. LNCS (LNAI), vol. 2639, pp. 44–51. Springer, Heidelberg (2003)
30. Zadeh, L.A.: Fuzzy sets. Information and Control 8(3), 338–353 (1965)
31. Zakowski, W.: Approximations in the space (U, Π). Demonstratio Mathematica 16(3), 761–769 (1983)
32. Zhu, W.: Topological approaches to covering rough sets. Information Sciences 177(6), 1499–1508 (2007)
33. Zhu, W.: Relationship between generalized rough sets based on binary relation and covering. Information Sciences 179(3), 210–225 (2009)

Bengali Printed Character Recognition – A New Approach

Soharab Hossain Shaikh[1], Marek Tabedzki[2], Nabendu Chaki[3], and Khalid Saeed[4]

[1] A.K.Choudhury School of Information Technology, University of Calcutta, India
soharab.hossain@gmail.com
[2] Faculty of Computer Science, Bialystok University of Technology, Poland
m.tabedzki@pb.edu.pl
[3] Department of Computer Science & Engineering, University of Calcutta, India
nabendu@ieee.org
[4] Faculty of Physics and Applied Computer Science,
AGH University of Science and Technology, Cracow, Poland
saeed@agh.edu.pl

Abstract. This paper presents a new method for Bengali character recognition based on view-based approach. Both the top-bottom and the lateral view-based approaches have been considered. A layer-based methodology in modification of the basic view-based processing has been proposed. This facilitates handling of unequal logical partitions. The document image is acquired and segmented to extract out the text lines, words, and letters. The whole image of the individual characters is taken as the input to the system. The character image is put into a bounding box and resized whenever necessary. The view-based approach is applied on the resultant image and the characteristic points are extracted from the views after some preprocessing. These points are then used to form a feature vector that represents the given character as a descriptor. The feature vectors have been classified with the aid of k-NN classifier using Dynamic Time Warping (DTW) as a distance measure. A small dataset of some of the compound characters has also been considered for recognition. The promising results obtained so far encourage the authors for further work on handwritten Bengali scripts.

Keywords: Bengali character, view-based algorithm, layer-based method, bounding box, unequal partition.

1 Introduction

Character recognition has been a popular field of research for past few decades. Research, in this arena, has been done not only on Bengali but also on some other languages [3], [4]. In [3], a method for handwriting recognition is proposed for Polish alphabet. It is based on Toeplitz matrix minimal Eigen values approach. In [4] a Template Matching based signature recognition algorithm is presented. In [11] a successful trial was made to recognize both typewritten and handwritten English and Arabic texts without thinning on the basis of region growing segmentation. In this

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 129–140, 2013.

work, however, and following the view-based approach of [5], [6], the Bengali language is studied for automatic recognition. Recognition of Bengali script has a lot of importance. Bengali is one of the most popular languages in India. All over the world more than 200 million people speak in Bengali and this is the second most popular script next to Devanagari in India. It also suggests the scripts of two other languages, Assamese and Manipuri. Bengali is the official language of Bangladesh, a neighbour of India.

Recognition of Bengali printed as well as handwritten characters has been a popular area if research in the arena of OCR for past few years as found in the literature [1, 2, 6, 9, 13, 14, 15]. Research is being done on the recognition of both the basic [10] and compound [9] Bengali characters. Attempts have also been made in the recognition of Bengali numerals [13], [18]. The modern Bengali alphabet set consists of 11 vowels and 39 consonants. These characters are called *basic characters*. Bengali text is written from left to right. The concept of upper/lower case is missing in Bengali. Most of the Bengali characters have a running horizontal line on the upper part of the characters; this line is known as *Matra*.

Characters in Bengali are not alphabetical as in English (or Roman) where the characters largely have one-sound one-symbol characteristics. It is a mixture of syllabic and alphabetic characters [9]. The use of modified and compound characters is also very common in Bengali. This paper presents methods for recognizing Bengali printed characters based on view-based approach. Both the top-bottom and left-right view-based approaches have been considered. This work is an extension of [6]. In this paper we have considered unequal partitions of the character images. Also a set of compound characters have been considered for view-based analysis.

The rest of the paper is organized as follows: section 2 is a short review of the existing literature. Section 3 describes the major functional steps involved in the recognition process and feature extraction methods. In section 4 the concept of unequal partitioning is presented followed by the considerations for compound characters. Classification and experimental results are given in section 5.

2 Previous Work

Different techniques have been found in the literature for optical character recognition. The curvelet transform has been heavily utilized in various areas of image processing. In [10] a novel feature extraction scheme is proposed on the basis of the digital curvelet transform. The curvelet coefficients of an original image as well as its morphologically altered versions are used to train separate k–nearest neighbour classifiers. Output values of these classifiers are fused using a simple majority voting scheme to arrive at a final decision. In [22] a method has been suggested based on curvature-based feature extraction strategy for both printed and handwritten Bengali characters. BAM (Bidirectional Associative Memories) neural network has been used in [19] for Bengali character recognition. The conventional methods are used for text scanning to segmentation of a text line to a single character. An efficient procedure is proposed for boundary extraction, scaling of a character and the BAM neural network which increases the performance of character recognition are used. In [20] a modified learning approach, using neural network learning for recognizing Bengali characters, has been presented. Research has been done on the recognition of handwritten Bengali characters [14]. Multi-Layer Perceptron (MLP) trained by back-propagation (BP) algorithm have been used as classifier.

In [18] an automatic recognition scheme for handwritten Bengali numerals using neural network models has been presented. A Topology Adaptive Self Organizing Neural Network is first used to extract from a numeral pattern a skeletal shape that is represented as a graph. Certain features like loops, junctions etc. present in the graph are considered to classify a numeral into a smaller group. If the group is a singleton, the recognition is done. Otherwise, multilayer perceptron networks are used to classify different numerals uniquely. Hidden Markov Models (HMMs) are used for both online and offline character recognition systems for different scripts around the world. A OCR program that uses HMM, for recognition process, has been made for Bengali documents in [12]. For using HMM it is required to have a sequence of objects to traverse through the state sequence of HMM. So the features are shaped into a sequence of objects. For each character component, a tree of features is made and finally the prefix notation of the tree is applied to the HMM. In the tree, the number of child of a node is not fixed, so, the child-sibling approach is applied to make the tree. Hence the prefix notation of the tree will contain nodes in the order: root, prefix notation of the tree rooted at its child, prefix notation of the trees rooted at the child's siblings from left to right order. After that HMM is used for the recognition purpose. Attempts have also been made on methods of segmentation and recognition of unconstrained offline Bengali handwritten numerals [13]. A projection profile based heuristic technique is used to segment handwritten numerals. A neural network based classifier is used for classification purpose. Paper [23] addresses various aspects of the problems associated with processing and recognition of printed and handwritten Bengali numerals. A scheme is proposed in this work for recognizing handwritten as well as printed numerals with different fonts and writing styles including noisy and occluded numerals. Polygon approximation is used to represent the contours of the letters. After that Fourier descriptors are used as shape features. The standard Multi-Layer Perceptron (MLP) augmented with MAXNET was used as a classifier. In [21] a method has been presented based on primitive analysis with template matching to detect compound Bengali characters. Most of the works on Bengali character are recognition of isolated characters. Very few papers deal with a complete OCR for printed document in Bengali. In [17] a chain code method of image representation is used. Thinning of the character image is needless when chain code representation is used. The main difficulties in printed Bengali text recognition are the separation of lines, words and individual characters. In [16] a new approach has been proposed to segment and recognize printed Bengali text using characteristic functions and Hamming network. A new algorithm has been proposed to detect and separate text lines, words and characters from printed Bengali text. The algorithm uses a set of characteristic functions for segmenting upper portion of some characters and characters that come under the Base line. It also uses a combination of Flood-fill and Boundary-fill algorithm for segmenting some characters that cannot be segmented using traditional approach. Hamming network is used for recognition scheme.

Recognition is done for both isolated and continuous size independent printed characters. In [15] a study has been made on handwritten Bengali numerals.

3 Major Functional Steps

Figure 1 shows the flowchart of major functional steps which have been outlined as follows:

i) Binarization: Printed documents written in Bengali have been scanned using a flat-bed scanner. Samples have also been collected using software supporting different Bengali fonts. These samples are converted to images and all the samples have been binarized.

ii) Segmentation: The documents contain Bengali text. Individual character has to be extracted from the text before applying view-based approach. Histogram of individual pixel row and columns of the text is computed. The individual lines containing many words have been segmented out from the text image using a horizontal histogram. The individual letters have been segmented out from the images of lines of text using vertical histogram.

iii) Matra Removal: The *Matra* is removed from top of the character. Standard image-editing software is used for doing the same. After removing the *Matra*, the characters without Matra is stored. View based approach is performed on these images. The importance of this phase is detailed out in section 3.1.1.

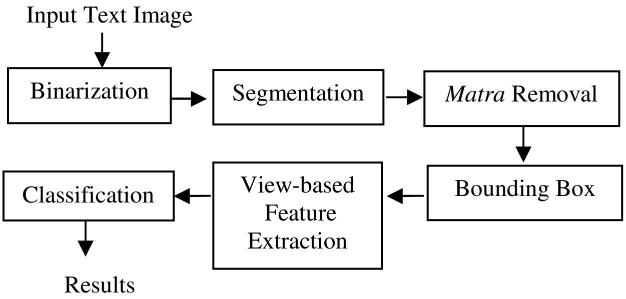


Fig. 1. Flow-chart of Major Functional Steps

iv) Applying Bounding Box: The character is put into a bounding box (rectangle that most tightly contains the character) before applying the view-based approach. The bounding box may be used as an indicator of the relative positions of features in a character.

v) View-based Feature Extraction: The features are extracted from four views of each individual letter. Additionally, the number of changes of the pixel values from white-to-black and vice versa have been calculated for each row and column. In inner-views approach the views of partitioned image are used to extract the features. These values form the feature vector representing the particular letter. This is detailed out in section 3.1. This vector is finally fed to the classifier for classification and recognition.

vi) Classification: The feature vectors are classified with the help of Dynamic Time Warping distance, k-NN and Ensemble of Classifiers in the present experiment. This has been explained in section 5.

3.1 Methods for Feature Extraction

In this section the view-based approach is detailed out. Also the layer based approach and the concept of bounding box is described.

3.1.1 View-Based Approach

This idea was presented in [5], [6] and [8]. The method is based on the fact that for correct character recognition one usually needs only partial information about its shape – or contour.

The “views” of given object are examined to extract from them a characteristic vector, which describes its shape. The view is representation of the coordinates of the points lying on one of four boundaries of the object (top, bottom, left or right) – it consists of pixels belonging to the contour of a word and having extreme values of *x* (for left and right view) or *y* coordinate (for top and bottom). The conventional preprocessing stage of thinning is not essential here because in this approach only the shape of the character is analyzed. However, and in some cases, as will be seen later, the thinning process helps improve the results. The binarization is required to convert the scanned image into a black-and-white one. The main feature extraction step is to take out all the characteristic points from the views.

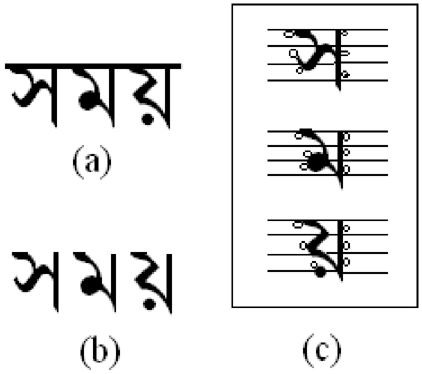


Fig. 2. (a) The Bengali word *Somoy* with *Matra*. (b) *Somoy* after removing *Matra*, (c) The letters of the word segmented and 3 horizontal partition is done. Also characteristic points have been marked on the left and right contour in each segment for Lateral view based consideration.

Thus, characteristic vector for top view consists of maximal values from each pixel-column on an image. To reduce dimensionality of obtained vectors one can introduce some form of down-sampling. In our case it is equivalent to searching for characteristic points in each view that describes the shape. In [7] several methods of selecting these points have been explored. In the method used in this paper, characteristic values are found by dividing original vectors into *n* identical segments, and calculating mean value for each one. In this way four characteristic vectors describing the given letter is found. Next, these four vectors are transformed into a one vector, which describes the given letter.

The top-bottom view based approach cannot be directly applied to Bengali characters. This is because most of the Bengali characters have a line known as the "*Matra*" on top of them. So, the top view for almost every character (except a few characters that do not have a *Matra* on top of them) is same. So the *Matra* should be removed first; after that the character without *Matra* can be processed for feature extraction using top-bottom view-based approach. *Matra* removal phase is not necessary if only lateral-view based approach is applied for feature extraction. But here both the lateral as well as top-bottom view based approaches have been applied, so removing Matra is necessary.

Figure 2 presents this process for a word "Somoy" meaning time in Bengali.

3.1.2 Layer-Based Approach

In layer based approach the number of changes of pixel values from black to white (or transitions from foreground to background) is considered. It tells whether the pixel-column (or row) contains one or several strokes, as is presented in Figure 3. This way some information about the *inner shape* of the letters can be obtained. For example two Bengali characters *Ja* and *Sha* have been considered here. Both the characters look exactly the same as far as the contour is concerned. The only difference is in the internal shape of the characters. Layer based approach can be applied to distinguish these letters. The same can be done for pixel rows. With the use of layer-counting (layer-view), improvements in results have been achieved for word recognition as in [15].

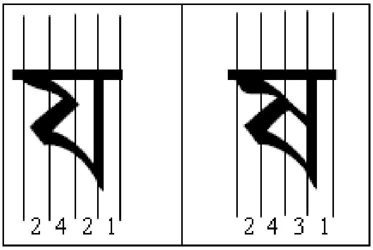


Fig. 3. Number of transitions for the letters *Ja* and *Sh*

Two variations of this approach have been considered. In each pixel column the number of changes of colour from white to black and black to white is counted. The same approach is performed for the pixel rows.

When considered the logical partitioning of the image (3 and 6 partitions) the average number of colour changes in each partition is calculated and used to form a feature vector. However, that lead to very poor recognition accuracy. So, later the whole image without any partition is considered and the layer-based approach over all the pixel rows and columns of the binary image has been performed.

3.1.3 Inner-View Approach

In order to improve the effectiveness of detection another idea was proposed. It implies the inclusion of additional information in the feature vector about the internal structure of the characters. It took the form of internal views. This is an innovative proposal used for the first time.

As with the classical view-based approach, there are also four possible views – top, bottom, left and right. This time, however, they do not form the outer shape of the letter, but the internal one. For this purpose, the character image is divided into two parts. For example, to obtain the top and bottom views, the image is divided horizontally into lower and upper parts. Then a bottom view of the upper half and a top view of the lower one can be obtained. Similarly, for the right and left view - the internal view of the left and right halves is created. The method of characteristic vector extraction for each view does not differ from the one presented in section 3.1.1.

4 Handling Compound Characters

A compound character is one that is formed by clubbing more than one basic character. Examples of some compound characters in Bengali are given in Figure 4. The basic characters forming those compound characters are also shown. There are a few problems in recognizing the compound characters.

Firstly, the problem is that there is no standard segmentation method that suits for all the compound characters properly. Component basic characters may retain their shapes in the compound character. Figure 4 shows two groups of compound characters. In the first group, the shapes of the component basic characters are retained in the compound one whilst in the second group, the basic shapes of the component characters are not retained.

So if the compound character is segmented properly then the basic component characters can be extracted out of it and on the basis of information of the component characters, the compound character can be identified. But as shown in Figure 4, first group, that no specific segmentation method applies well for all the compound characters. For the characters at line 1 and 2 a horizontal segmentation works well. For the characters at line 3 and 4, a diagonal segmentation is required, whereas for the character at line 4, a horizontal segmentation will be suitable.

Secondly, as shown in the second group of Figure 4, the shape of the compound characters may change significantly when two basic characters are clubbed to form it. This compound character contains no clue of the basic components. So no segmentation can be done for recognizing such characters.

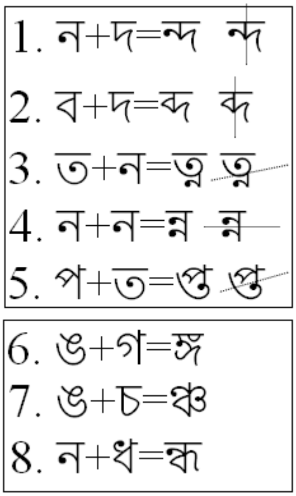


Fig. 4. Compound characters and segmentation for extracting the individual basic component characters

Thus, the methods based on the segmentation of compound characters do not give satisfactory results for recognizing compound characters. In this paper view-based approach is applied for performing this recognition task. The idea is that the contour of each compound character is unique. So, applying view-based approach a feature vector can be formed for each of the compound characters. This feature vectors should be stored in a vector codebook database. Whenever a character has to be identified, the view-based approach is applied to it to form a feature vector. This vector is then compared with those at the codebook to find a similar match. This way a compound character can be identified.

5 Classification and Experimental Results

For the purpose of classification the method based on k-Nearest Neighbor (k-NN) algorithm has been used. It is a type of instance-based learning, where classification is based on closest training examples in the feature space. The training set is composed of labeled examples (feature vectors of correctly classified Bengali characters). In the classification phase, an unlabeled vector is classified by assigning the label which is most frequent among the *k* training samples nearest to that vector. In the present experiments distance is calculated using DTW approach [4], [24].

In all experiments *K*-fold cross-validation method is used, so original database was randomly partitioned into *K* (where *K* = 10) equal size subsets. Classification process was then repeated *k* times, with each of the *k* subsets used exactly once as a test set, where the remaining *K* -1 subsets are used as training data. The results from *K* folds were then combined to produce recognition result estimation.

Tests were performed on the collected database of 1000 characters. Images were binarized, cleaned of noise and prepared to recognize.

Table 1. Results of classification

Test	Views	Layers	Inner Views	KMM	Features	Recognition rate
1	✓				32	43.7%
2	✓			✓	32	57.2%
3	✓			✓	40	56.5%
4	✓			✓	24	55.9%
5		✓		✓	16	46.2%
6			✓	✓	32	62.7%
7	✓	✓	✓	✓	80	**76.8%**

Authors performed a series of experiments, designed to allow the determination of the optimal strategy and parameters. The following table shows some sample results.

In the first approach only views were used (four for each character), wherein from each view 8 points were collected giving a 32-element long feature vector. This has resulted in a rather low efficiency of about 44% characters correctly classified.

The next experiment revealed that prior character skeletonization allows for a significant improvement in efficiency. For this purpose authors used KMM algorithm [25]. In this approach the character image was thinned to a skeleton form before applying view-based method to extract features. Thanks to this the percentage increased to 57%.

Approach 3 and 4 show an attempt to determine the optimal number of points extracted from each view. As can be seen an attempt to reduce or increase this value does not bring improvement indicating that for the studied characters, the optimal number proved to be eight points per view.

The fifth study tested the effectiveness of the layer-based approach. For this purpose the algorithm of section 3.1.2 was used, designating the number of transitions for the vertical and horizontal directions. For each of the two directions only 8 values were considered – using the very same method of downsampling, as in the view-based approach described in section 3.1.1. The percentage is lower than that of the view-based approach, but the attention should be paid to the smaller size of the vector as well as that both of these approaches will be used jointly.

Experiment 6 shows a novel approach used for the first time as presented in this work - it is based on the use of internal views as described in section 3.1.3. Also four views were used and 8 characteristic points were collected from each view. As can be seen, this allows for a larger percentage than the classical approach of over 60%.

The last row of the table shows the results obtained by a combination of all these approaches. In this manner, the effectiveness of recognition has reached 76.8%. The cost is obtaining a large feature vector due to the fact that all of the features of the previous studies: views, layers and inner views were used. The solution to this issue would be to carry out a prior feature selection, it would help to determine which of features are relevant to the process of classification and which may be removed.

Table 2 collects the results of the last survey (approach no 7) breaking them into some selected characters. This allows to observe the individual results of each of them. Columns of the table presents the selected letter and the results of recognition. As can be seen, some characters were classified with efficiency exceeding 90%, while some decreases the performance.

Table 2. Results of classification for individual letters

Letter	Recognition rate
ঔ	95%
ঢ়	90%
ড়	85%
ড	70%
গ	70%
ন	60%
ম	50%
ঘ	50%
প	40%
থ	35%
স	25%
Overall	**76.8%**

In order to determine the causes of low rate obtained for some of the characters, the results of recognition were formed into matrix of confusion, to expose most common mistakes. This revealed that some of the characters were particularly often confused with each other. Some of these pairs are shown in Table 3.

Table 3. Most common mistakes

Pairs of confused letters
ম – স
গ – প
ষ – য
থ – থ

Each of these pairs was confused with each other in at least 20% of cases. As can be seen they are actually very similar to each other and of very similar shape. Hence, the algorithm could not distinguish between them.. Further work will be aimed at more closely look at the similar pairs of characters to find specifications that allow them to differentiate and thus improve the efficiency of recognition.

6 Conclusions and Future Work

In this paper, the authors have proposed a new method for Bengali printed character recognition using View-based and Layer-based methods. The concept of unequal partitioning has been applied in this study. A classifier based on DTW and k-NN has been used for the experimentation purposes. Recognition accuracy of 76.8% has been achieved which is considered to be very high as the Bengali alphabets resemble handwriting rather than type writing. It contains alphabets that are cursive in nature. The proposed method therefore, will be the basis of handwritten character recognition.

The view-based approach is being applied for Bengali character recognition for the first time. The experiments show gradual improvements. The future plan is to modify the recognition engine and use a different pattern classifier like a neural network to study the performance of the proposed method.

Acknowledgement. This work was supported by Bialystok University of Technology grant S/WI/2/2013.

References

[1] Bag, S., Bhowmick, P., Harit, G.: Recognition of Bengali Handwritten Characters Using Skeletal Convexity and Dynamic Programming. In: 2nd International Conference on Emerging Applications of Information Technology (EAIT), Kolkata, pp. 265–268 (2011)

[2] Naser, M.A., Hossain, M.M., Tito, S.R., Hoque, M.A.: Recognition of Bangla Characters using Regional Features and Principal Component Analysis. In: International Conference on Electrical and Computer Engineering (ICECE), Dhaka, pp. 506–509 (2010)

[3] Tabędzki, M.: Handwriting Recognition System on the basis of Toeplitz Matrices. PhD Thesis, Bialystok University of Technology, Bialystok, Poland (2010)

[4] Adamski, M.: Signature Recognition by the use of Template Matching Techniques. PhD Thesis, Bilystok University of Technology, Bialystok, Poland (2010) (in Polish)

[5] Tabedzki, M., Rybnik, M., Saeed, K.: New Results for View-Based Feature Extraction Method for Handwritten Words Recognition without Segmentation. In: 1st International Conference on Image Processing & Communications, Poland (2009)

[6] Shaikh, S.H., Chaki, N., Tabedzki, M., Saeed, K.: A View-based Approach for Recognition of Bengali Printed Characters. In: 8th International Conference on Computer Information Systems and Industrial Management (CISIM), Coimbatore, India (December 2009)

[7] Tabedzki, M., Rybnik, M., Saeed, K.: A Method for Handwritten Word Recognition without Segmentation. Polish Journal of Environmental Studies 17(4C), 47–52 (2008)

[8] Saeed, K., Tabedzki, M.: New Experiments on Word Recognition Without Segmentation. In: Advances in Information Processing and Protection, Part-III, pp. 323–332. Springer, US (2008)

[9] Pal, U., Wakabayashi, T., Kimura, F.: Handwritten Bangla Compound Character Recognition using Gradient Feature. In: 10th International Conference on Information Technology (ICIT), pp. 208–213 (December 2007)

[10] Majumdar, A.: Bengali Basic Character recognition using Digital Curvelet Transform. Journal of Pattern Recognition Research, 17–26 (2007)

[11] Saeed, K., AlBakoor, M.: Region Growing Based Segmentation Algorithm for Typewritten and Handwritten Text Recognition. Applied Soft Computing 9(2), 608–617 (2009)

[12] Monjel, M.S., Khan, M.: Optical Character Recognition for Bangla Documents Using HMM. Technical Report, Centre for Research on Bangla Language Processing, CRBLP (2007)

[13] Das, D., Yasmin, R.: Segmentation and Recognition of Unconstrained Bangla Handwritten Numeral. Asian Journal of Information Technology 5(2), 155–159 (2006)

[14] Bhattacharya, U., Shridhar, M., Parui, S.K.: On Recognition of Handwritten Bangla Characters. In: Kalra, P.K., Peleg, S. (eds.) ICVGIP 2006. LNCS, vol. 4338, pp. 817–828. Springer, Heidelberg (2006)

[15] Chaudhuri, B.B.: A Complete Handwritten Numeral Database of Bangla – A Major Indic Script. In: 10th International Workshop on Frontiers in Handwriting Recognition (2006)

[16] Hasan, M.A.M., Alim, M.A., Islam, M.W.: A New Approach to Bangla Text Extraction and Recognition from Textual Image. In: 8th International Conference on Computer and Information Technology, ICCIT (2005)

[17] Mahmud, J.U., Raihan, M.F., Rahman, C.M.: A Complete OCR System for continuous Bengali Character. In: TENCON 2003, Conference on Convergent Technologies for Asia-Pacific Region, October 15-17 (2003)

[18] Bhattacharya, U., Das, T.K., Datta, A., Parui, S.K., Chaudhuri, B.B.: Recognition of Handprinted Bangla Numerals Using Neural Network Models. In: Pal, N.R., Sugeno, M. (eds.) AFSS 2002. LNCS (LNAI), vol. 2275, pp. 228–235. Springer, Heidelberg (2002)

[19] Syeed, M.M.M., Siddiqui, F.H., Al-Mamun, A.S.A., Tanbeer, S.K., Mottalib, M.A.: Bengali Character Recognition using Bidirectional Associative Memories (BAM) Neural Network. In: 5th International Conference on Computer and Information Technology (ICCIT), pp. 247–251 (2002)

[20] Karim, M.E., Hossain, M., Mottaliband, M.A., Zaman, T.: A Modified Neural Network Learning Approach and its Application to Bengali Character Recognition. Malaysian Journal of Computer Science 11(2), 68–73 (1998)

[21] Chaudhuri, B.B., Pal, U.: A Complete Printed Bangla OCR System. Pattern Recognition 31, 531–549 (1997); Graphics and Image Processing, NCCIS

[22] Dutta, A., Chaudhuri, S.: Bengali Alpha-numeric Character Recognition using Curvature Features. Pattern Recognition 26, 1757–1770 (1993)

[23] Datta, S., Chaudhury, S., Parthasarathy, G.: On Recognition of Bengali Numerals with Back Propagation Learning. In: IEEE International Conference on Systems, Man and Cybernetics, vol. 1, pp. 94–99 (October 1992)

[24] Salvador, S., Chan, P.: Toward accurate dynamic time warping in linear time and space. Intell. Data Anal. 11(5), 561–580 (2007)

[25] Saeed, K., Tabędzki, M., Rybnik, M., Adamski, M.: K3M – A Universal Algorithm for Image Skeletonization and a Review of Thinning Techniques. International Journal of Applied Mathematics and Computer Science 20(2), 317–335 (2010)

Eye Location and Eye State Detection in Facial Images Using Circular Hough Transform

Ömer Faruk Söylemez[1] and Burhan Ergen[2]

[1] Dicle University, Diyarbakir, Turkey
osoylemez@dicle.edu.tr
[2] Firat University, Elazig, Turkey
bergen@firat.edu.tr

Abstract. Recently, eye states are used as inputs to various applications such as facial expression recognition systems, human-computer interaction and driver fatigue detection systems. Especially with the prominence of human computer interaction, eye state detection has drawn great attention in the past decade. In this study, an eye state detection system based on Circular Hough Transform (CHT) has been offered. Initially, face and eye images are extracted from given gray-level images. After some preprocessing steps, existence of circular iris structure is searched within the extracted eye image using CHT. Existence of circular iris structure is searched within the eye image with the help of circular Hough transform. Eyes are decided as open if iris could identified as a circle.

Keywords: Eye state detection, Circular Hough transform.

1 Introduction

Detection of eye location and states has received a great deal of attention because eyes are the most important part of human face. Today, many applications are using eye states as inputs. Facial expression recognition, human-computer interaction and driver fatigue detection systems could be given as examples to such applications. In facial expression recognition systems, subjects mood could be estimated by utilizing eye openness along with other facial features [1]. In driver fatigue detection systems, eye blinking count, time taken by a blink and frequency of eye blinks are combined together to form an alert and warn the driver in case of a driver fatigue level increase [2]. In human computer interaction, by tracing eye movement and focus, computers could acknowledge and process user requests [3]. By employing these kinds of applications in academical and commercial fields, eye state detection has drawn great attention in the past decade. In the field of eye state recognition, studies are employing parameters such as iris, eye perimeter, eye edge and eye lids. Wang and Yang [4] used chain code tracing for extracting edges of iris and then used Circular Hough Transform (CHT) in order to detect iris circle. Tian, Kanade and Cohn [5] took advantage of Gabor wavelets and neural networks in order to detect eye states.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 141–147, 2013.

Seneratne et al. [6] compared eye blocks intensity values using support vector machines (SVM) and Naive Bayes (NB) classification methods to overcome eye state detection problem. Xu, Zeng and Wang [7] tried to detect eye states by using Adaboost based cascade classifier and a histogram property which is named as Local Binary Pattern (LBP) . In this study, we carry out a practice to detect eye states in facial images. Initially, faces and eyes are detected and extracted from intensity images. After some preprocessing steps, edges of the iris are used to detect a circular structure referring to state of the eyes. The detection of a circular structure is achieved by means of CHT.

2 Proposed Method

Our proposed method for eye state detection is composed of three steps. Viola-Jones [9] cascade object detector which gives higher recognition rates on real time object detection has been used for face and eye detection states. Eye regions are estimated sequel to extraction of faces and eyes from given images. CHT is applied to the images consisting of eye region at the end, in order to detect eye states. BioID face database [8] which consists of 1521 gray scale images that are taken from 23 different subjects under varying circumstances is used in our study.

2.1 Face and Eye Detection

Cascade classifier model which is proposed by Ojala T. [10] was executed on 1010 images that provide adequate lighting levels. Face locations were correctly identified on 916 images. Eyes are searched within those locations by the help of Viola-Jones object detection framework just like as face detection. Cascade classifier model which proposed by Marco [11] is used for detecting eye pairs. To detect each eye separately; cascade classifier model which proposed by Shiqi [12] is used. Since our method is based on the intensity of dark region formed by iris and pupil, employing an eye image with the absence of eyebrows became more suitable for detection purposes. Therefore, we employ eye pair images for eye detection instead of separately detected eye images for their minimal eyebrow containment. Cascade classifier was performed on total of 916 images. Eye pairs were detected successfully in 899 images.

2.2 Eye State Detection

In order for an eye to carry out seeing task, a black point which is called pupil has to receive light. Lights received from pupil are focused on to rearmost layer of the eye which is called retina layer. Image that is formed on the retina layer is transmitted to visualization center of the brain via optical nerves and thus visual process completes. Iris and pupil are two circles that share the same center. In accordance with this relation half opening of iris means pupil is also half open and so seeing is probable. We have used this information in our study to assume an eye as open. Our system starts with equalizing contrast of the acquired eye image with the histogram equalization. As a result of this process, iris and pupil, which have a lower intensity value compared to sclera (white of the eye) are going to be easily separated from the sclera and the rest of the eye. Afterwards, gray-scale thresholding is applied to the resulting eye image. With the aid of thresholding, iris and pupil had become greatly separated from the rest of the eye. Even so, some unwanted features such as shades, eyelashes and stains could may have reside on the image. Therefore we removed those little residues after obtaining thresholded image. Yet another great factor that hampers our method is the lights which are reflected from the pupil. These lights are forming a high intensity area (brights) inside the low intensity area which is formed by pupil and iris. As this white area has the ability to misdirect our edge detection output, it has been dealt with morphological techniques. After all of these processes, iris and pupil are completely extracted from the rest of the eye and are ready to be served into next step. Edges of the concerned image are detected with use of canny edge detector. Afterwards, presence of circular shapes are searched with the aid of circular Hough Transform. 492 eye images were used for eye state detection. Those images are selected within 899 eye pairs that are obtained from the previous step and they meet the minimum resolution case for applicability of CHT. In 473 of the images eyes were open and in the other 19 of the images eyes were closed. Open and closed eyes are classified with %94.5 and %63.2 accuracy respectively.

3 Experimental Results

In this section, we will analyze our proposed method by means of error matrix, correct classification rate and kappa statistics. Afterwards, we will present some test results that are obtained with our proposed method.

3.1 Confusion Matrix

Confusion matrix is a tool to measure the performance of a classification system. Each row of the matrix shows number of occurrences of an estimated class, while each column of the matrix shows the occurrence of a real class. A confusion matrix is shown in Table 1, which is used to classify a two class system. As stated above; TP (resp. TN) represents the number of instances of class a (resp. class b) well classified by the system. P=TP+FP (resp. N=FN+TN) represents the total of estimated occurrences of class a (resp. class b). p=TP+FN (resp. n=FP+TN) represents the total of real occurrences of class a (resp. class b). T is the sum of occurrences of both classes.

3.2 Correct Classification Rate

Correct classification rate (CCR) which is also known as success rate, is obtained by dividing the number of correctly classified samples to the number of total samples as shown in Equation 1.

Table 1. Confusion matrix of a classification system

Real Class/Estimated Class	A	B	Total
a	True Positive (TP)	False Negative (FN)	p
b	False Positive (FP)	True Negative (TN)	n
Total	Positive (P)	Negative (N)	Total (t)

$$CCR = \frac{TP + TN}{T} \tag{1}$$

3.3 Cohen's Kappa Coefficient

Cohen's Kappa coefficient (κ) [13] measures the agreement between two raters who each classify N items into C mutually exclusive categories. The equation for Cohens Kappa coefficient is shown in Equation 2.

$$\kappa = \frac{P_0 - P_e}{1 - P_e} \tag{2}$$

Where P_0 is observed agreement proportion that refers to CCR and P_e represents random agreement proportion given by Equation 3.

$$P_e = \frac{1}{T^2}[(P * p) + (N * n)] \tag{3}$$

κ coefficient varies between -1 and 1. When it is equal to 1, it is considered that agreement between the observers on the studied system is perfect. Inversely when it is equal to -1, it is considered that there is a total disagreement on the studied system among the observers. Table 2 shows kappa statistics values along with their interpretations.

Table 2. Kappa statistics values along with their interpretations

Kappa Statistics	Interpretation
$\kappa > 0.8$	Almost Perfect Agreement
$0.8 \geq \kappa > 0.6$	Substantial Agreement
$0.6 \geq \kappa > 0.4$	Moderate Agreement
$0.4 \geq \kappa > 0.2$	Fair Agreement
$0.2 \geq \kappa > 0.0$	Slight Agreement
$0 \geq \kappa$	No Agreement

Table 3. Statistical values of the system

TP	TN	FP	FN	T	CCR	Kappa
447	26	12	7	492	0.93	0.40

Table 4. Eye state detection success rates

Real/Detected	Open	Closed	Total
Open	447(%94,5)	26(%5,5)	473(%96,1)
Closed	7(%36,9)	12(%63,1)	19(%3,9)
Total	454(%92,2)	38(%7,8)	492(%100)

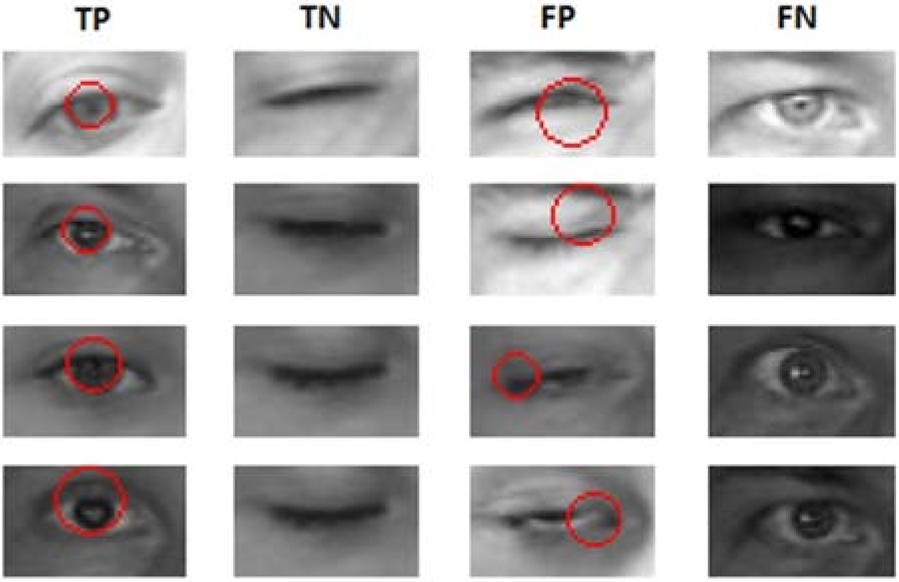


Fig. 1. Examples to detection classes

3.4 Experimental Results

All applications are executed on Matlab software environment by using the system having the following specification: Intel Core2duo T5600 Dual Core Processor clocked at 1.83 GHz, 533 MHz 4 GB RAM. Our software took 0.14 seconds to localize face, 0.06 seconds to localize eye pair and cropping eyes, 0.014 seconds for preprocessing steps and another 0.014 seconds for circular Hough transform. Our software took total of 230 ms in order to process a frame which means 4 fps.

Table 3 shows statistical values of the proposed system. Kappa statistic is found as 0.40 which shows that our system works on a fair agreement level. Eye state detection success rates are given in Table 4. %94,5 of the open eyes and %63,1 of the closed eyes are classified correctly. CCR, which is obtained by dividing correctly classified samples to all samples is calculated as %93.

Samples to the eye detection classes which are TP, TN, FP and FN are shown at Figure 1. Red circles are indicating a valid retina and pupil are found and marks where they reside.

4 Conclusion

With the development of computer vision techniques, face detection and eye state detection has become important for many applications. In this study, an eye state detection system has been proposed. In order to detect faces from given images, Viola-Jones face detector has been used. Faces had been detected successfully in 916 of 1010 images. Sequel to face detection, eyes pairs are detected by utilizing Viola-Jones eye pair detector. 899 eye pairs were detected in 916 face images. Right and left eyes are acquired by cropping eye pairs. For eye state detection, contrast of the eye image -which was obtained in the previous stage- is increased. Then gray level thresholding is applied to this image. Later, the residual areas which are not used in the detection process are eliminated. Holes inside this new image are filled. After this step, edges are extracted via canny edge detection algorithm. Finally, circular shapes are searched within images by using circular Hough transform. Latest image set was composed of 492 images which contains images that provide proper resolution and illumination for eye state detection process to function. % 94,5 percent of the open eye images and % 63,1 of closed eye images are identified correctly. Gray level images are used in this study. Studies that employ colored images could produce better results. Eye state detection could also be possible in low light environment by using capture devices that utilize infrared rays. Eye state detection problem could be solved more efficiently by applying these types of approaches. Eye state information, is used as input to many applications nowadays. With the development of faster and more accurate eye state detection methods there is no doubt that eye state information could utilize itself in broader utilization.

References

1. Sandbach, G., Zafeiriou, S., Pantic, M., Yin, L.: Static and dynamic 3D facial expression recognition: A comprehensive survey. Image and Vision Computing (2012)
2. Wang, Q., Yang, J., Ren, M., Zheng, Y.: Driver fatigue detection: a survey. In: The Sixth World Congress on Intelligent Control and Automation, WCICA 2006, vol. 2, pp. 8587–8591. IEEE (2006)
3. Jacob, R.J., Karn, K.S.: Eye tracking in human-computer interaction and usability research: Ready to deliver the promises. Mind 2(3), 4 (2003)

4. Wang, Q., Yang, J.Y.: Eye location and eye state detection in facial images with unconstrained background. Journal of Information and Computing Science 1(5), 284–289 (2006)
5. Tian, Y.-l., Kanade, T., Cohn, J.F.: Eye-state action unit detection by gabor wavelets. In: Tan, T., Shi, Y., Gao, W. (eds.) ICMI 2000. LNCS, vol. 1948, pp. 143–150. Springer, Heidelberg (2000)
6. Senaratne, R., Hardy, D., Vanderaa, B., Halgamuge, S.: Driver Fatigue Detection by Fusing Multiple Cues. In: Liu, D., Fei, S., Hou, Z., Zhang, H., Sun, C. (eds.) ISNN 2007, Part II. LNCS, vol. 4492, pp. 801–809. Springer, Heidelberg (2007)
7. Xu, C., Zheng, Y., Wang, Z.F.: Efficient eye states detection in realtime for drowsy driving monitoring system. In: IEEE Conf. on ICIA, vol. 1-4, pp. 170–174 (2008)
8. http://www.bioid.com/downloads/software/bioid-face-database.html (February 27, 2012)
9. Viola, P., Michael, J.J.: Rapid Object Detection using a Boosted Cascade of Simple Features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 511–518 (2001)
10. Timo, O., Matti, P., Topi, M.: Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(7), 971–987 (2002)
11. Marco, C., Oscar, D., Cayetano, G., Mario, H.: ENCARA2: Real-time detection of multiple faces at different resolutions in video streams. Journal of Visual Communication and Image Representation 18(2), 130–140 (2007)
12. http://yushiqi.cn/research/eyedetection (February 27, 2012)
13. Fleissa, J.L., Cohen, J., Everitt, B.S.: Large sample standard errors of kappa and weighted kappa. Psychological Bulletin 72(5), 323–327 (1969)

Recognition of Occluded Faces Based on Multi-subspace Classification

Paweł Forczmański and Piotr Łabędź

West Pomeranian University of Technology, Szczecin,
Faculty of Computer Science and Information Systems,
ul. Zolnierska 52, 71-210 Szczecin, Poland
{pforczmanski,plabedz}@wi.zut.edu.pl
http://pforczmanski.zut.edu.pl

Abstract. In the paper we investigate a problem of face recognition in uncontrolled environment – distorted by occlusion, shadows and other local modifications. Such problems are very common for real-world conditions, thus the presented algorithm allows to eliminate them. It is based on dimensionality reduction approach (two-dimensional Karhunen-Loéve Transform) and distance-based classification. We use simple transformations involving face normalization and individual facial regions extraction as a pre-processing. Then, we perform independent recognition of extracted facial regions and combine the results in order to make a final classification. The results of experiments conducted on images taken from 9 publicly available datasets show that a quite simple algorithm is capable of successful recognition without high computing power requirements, as opposite to more sophisticated methods presented in the literature. As it was proved, the presented approach gives significantly better efficiency than a whole image-based recognition.

Keywords: face recognition, subspace method, dimensionality reduction, PCA.

1 Introduction

The problem of automatic face recognition continuously attracts scientists interest. The review of literature and closer look at practical implementations show that although it might has been successfully solved, due to its broad characteristic, it is still interesting. This constant interest is motivated by the observation that most of the proposed solutions operate mainly in the very narrow real-world conditions (i.e. in the controlled conditions of imaging, with strictly defined illumination, orientation, pose, expression etc.).

The continuous research in this area is influenced also by the fact that facial recognition algorithms are to be implemented in the devices having low processing power and very limited resources like smartphones and tablets. Hence, robust solutions as simple as possible, yet without suffering from significant performance degradation are still needed.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 148–157, 2013.

One of the most important problems in face recognition is a low quality of input data processed by a typical Face Recognition System (FaReS). It should be noted, that the quality is often degraded by two main types of distortions observed in the face area: variable lighting and occlusions/incompleteness of face portrait.

Several examples of face occlusions or similar local distortions that can be found in practice are presented in Fig.1. As it can be seen, they include full occlusion of different face areas, partial occlusion by an independent object, local intensity changes or total information loss in a limited region of face. Those problems highly degrade the performance of FaReS, thus this is the main reason of lasting interest of researchers [1].

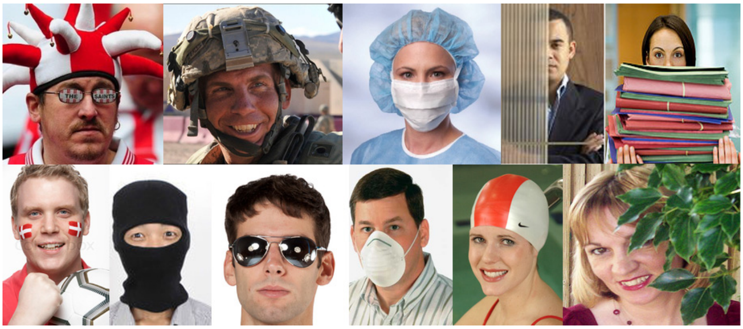


Fig. 1. Real distortion examples in the face area

It should be also emphasized, that in practice we have an access to a limited training database, thus we have to solve the facial portrait recognition without extending FaReS database with images having all possible variants of global and/or local distortions (in contrast to setups defined in [1]). The literature review shows also, that recognizing images under occlusion requires a careful selection of invariant features. Apart from many complex geometry-based approaches, there are several simple appearance-based methods that combine a few elementary features to describe an image, like Gabor Features[2] or LBP Features [3]. However they lead to the very complex multi-tier algorithms, which is unusable in low-performance devices.

One of the most popular approaches to face recognition employs Principal Component Analysis (PCA) also known as Karhunen-Loeve Transform (KLT) [1, 4, 6–9]. There has been many scientific works published in the recent years related to the data dimensionality reduction by means of PCA/KLT, as well as their application to the face recognition problem taking into account the two-dimensional characteristics of images. Such algorithms have got different names i.e. 2DPCA [10, 11] or MatPCA [12]. However, most of them are associated with the processing of two-dimensional data by means of data reduction algorithms performed along one dimension only or by dividing an image into smaller parts and reducing them with help of classical PCA. Moreover, those methods operate on fully visible facial portraits not considering a problem of occlusion and local distortions.

The literature review shows that there is a few attempts to deal with above problems, however they are still not fully solved. An interesting approach was presented by the authors of [13]. They address a problem of reconstructing images using so called Fast-Robust PCA. The experimental part proves that it is possible to rebuild missing pixels in images of known class, however it is computationally expensive. The improved method presented in [14] proved that it is possible to reconstruct distorted pixels in facial portraits using a modified (faster) PCA approach. That method makes it possible to reconstruct occluded regions of the face using Fast Weighted Principal Component Analysis (FW-PCA). The occluded regions are detected and recursively updated. Unfortunately, the authors do not address the problem of recognition accuracy. The extension of classical PCA, called Lophoscopic PCA, aimed at recognition of images under occlusion was presented in [15]. The authors covered only small parts of faces (one part a time) which do not represent real-life conditions. Moreover, the main drawback of above method is a complex and time-consuming calculations (six times slower than a classical PCA).

Another interesting algorithm was presented in [16]. The authors propose an approach which consists of first detecting the presence of scarf/sunglasses and then processing the non-occluded facial regions only. The occlusion detection problem is solved using Gabor wavelets, PCA and Support Vector Machines (SVM), while the recognition of the non-occluded facial part is performed using block-based Local Binary Patterns (LBP). The main drawback in this case is the assumption about the presence of sunglasses or the scarf, not taking into account other distortions. Moreover, LBP are not the most optimal face recognition approach. On more interesting approach was presented in [17], however that solution is devoted to a removal of one specific object, namely glasses.

The general disadvantage of above methods is the computational complexity coming from the iterative nature of calculating a reconstructed image or the need of detecting occluded regions. Thus, it is a major drawback when it comes to robust and fast face recognition.

In this paper we focus on a method involving dimensionality reduction approach operating in two-dimensional domain of KLT, as it proved to be very efficient, yet not very computationally expensive, especially in case of currently available computing power. We present an algorithm of solving occlusion problem in the aspect of facial portraits recognition, which is much simpler in comparison to the above presented approaches, yet its efficiency is very similar. It employs a template database that does not include images with local distortions. At the stage of preprocessing it uses face normalization and facial regions extraction. At the stage of feature extraction it uses 2DPCA/2DKLT as the only instruments of transformation of original data into the low dimensional space [18]. Finally, obtained feature vectors are used at classification step (using simple distance metrics).

2 Processing Algorithm

The main idea behind the recognition algorithm is the observation, that in most cases faces are occluded only to some extent. If a human being is able to recognize such an occluded portrait, a computer also can do it. So we assume the following. If we can successfully recognize *most* of facial parts, then we can decide about the recognition of the whole face.

Facial portraits have different geometrical characteristics, hence it is important to normalize them. Here we assume to have *en face* portraits. The preprocessing of such images includes orientation normalization (based on eyes positions), face cropping and scale standardization. For the purpose of experiments presented in this paper, the eyes positions were marked manually, hoverer it seems that using some sort of automatic approach (i.e. Viola-Jones detector) may give similar results. The intermediate results of pre-processing are presented in Fig.2.

Then, the distances between salient points are used to crop individual face parts: forehead, eyes, nose and mouth/chin. Such distances used at this stage are derived from the distance between eyes (a in Fig.2): $b = 1.5 * a$, $c = 1.8 * a$, and $d = 2 * a$.

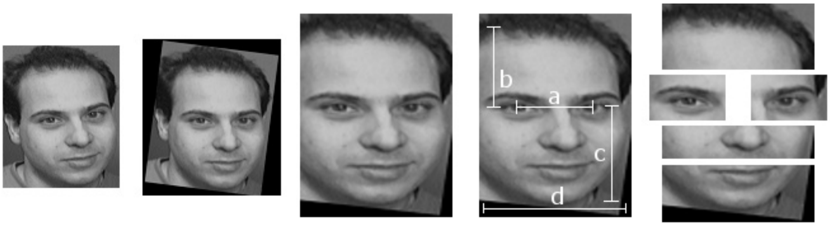


Fig. 2. Processing stages: original image, orientation normalization, scale standardization, salient point detection, facial areas extraction

The recognition algorithm consists of several stages: image pre-processing, facial regions extraction, dimensionality reduction, and recognition. The last stage includes parallel classification in multiple sub-spaces and final voting. In our proposition we divide a face into five, non-overlapping parts and perform an independent recognition on these parts. Such decomposition is motivated by a natural structure of face with five clearly distinguishable areas. As a result, we get several recognition results related to forehead, eyes, nose and mouth/chin areas. Each part belongs to a certain individual enrolled in the database. When we perform a voting on those results we get the final recognition result based on majority of votes.

The recognition scheme is presented in Fig.3. The *Facial Region Extraction* is a block responsible for face region extraction, *2DKLT* is a dimensionality reduction stage, *C* is a comparator (distance metrics) and *Result* blocks represent individual recognition results.

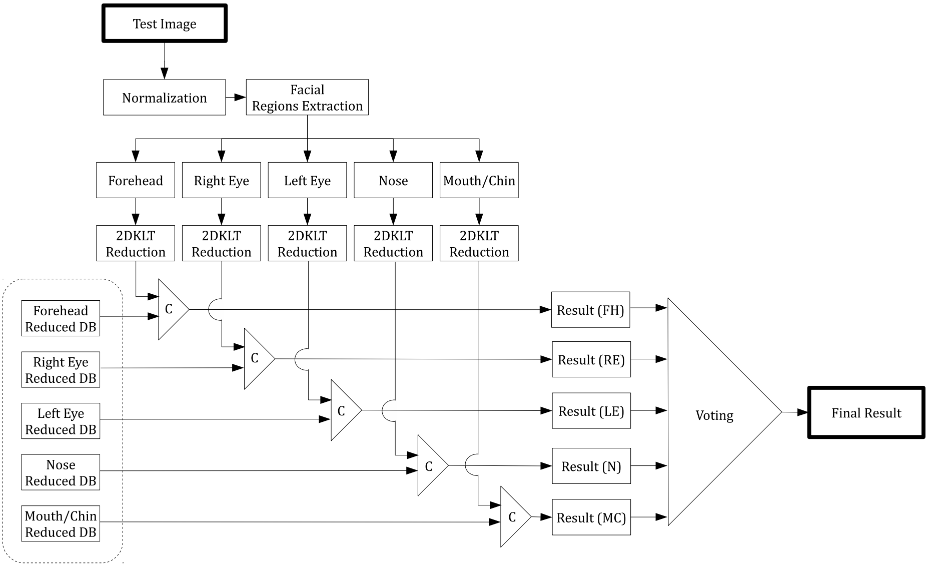


Fig. 3. Recognition scheme

Recognition of images is performed in subspaces related to each individual facial region. While the database images were registered in a precise manner (eyes positions were marked manually), test images are cropped in an approximate manner (since there is no guarantee, the eyes are visible). If the eyes are covered or impossible to localize, some different salient point detector may be used in order to calculate appropriate proportions (i.e. [2]).

Each region forms a database with reduced features and an information about the individual (*owner* of this particular part). We employ two-dimensional variant of well-known PCA/KLT approach as it makes it possible to obtain a compact representation of features [18]. As a distance metrics D_k for individual regions ($k = 1, 2, \dots 5$) we use the Chebyshev distance defined on a vector space where the distance between two feature vectors P and Q is the greatest of their differences along any (i-th) coordinate dimension:

$$D_k(P, Q) = \max_i(|p_i - q_i|). \tag{1}$$

Then we use a 1-Nearest Neighbor approach to find a class center (average vector for all elements in each class), which is the closest one. Finally we perform a voting on the individual results $D_1, D_2, \dots D_5$. We use simply the mode average of all votes, hence the majority indicates the class of the input image. Such an approach results in a fast and reliable recognition without high computational overhead.

Another interesting and practically important feature of presented approach is the possibility to compute a level of confidence. It is evaluated as a number of unanimous votes to the total number of votes. For example, if the voting results in 3 votes indicating the same class (the rest 2 votes indicate different classes), then the level of confidence is equal to 3/5. The minimal acceptable confidence level is equal to 2/5 (40%). Hence we can evaluate the quality of recognition and give a hint to the human operator about the system's performance.

In the voting scheme, there can occur a special case, when there is an equality situation among the votes, i.e. two of the regions say one identity, and two other regions say another identity. It should be noted, that we do not weight the individual results (we do not consider one part of a face be more important than another). Hence, the system results in a identity that is associated with face parts placed in the upper part of the head. In the future, some more sophisticated solutions could be employed, such as presented in [19].

3 Experiments

First, it should be noted, that there is no publicly available large database aimed at testing face recognition under occlusion. One possibility is to use AR database [20] which features frontal view faces with different facial expressions, illumination conditions, and occlusions (sun glasses and scarf). However it does not incorporate images with occlusions of different type, i.e. by independent objects, strong local shadows etc. Thus, the experiments were performed on a self-created database containing images taken from 9 officially available sets: Olivetti Research Lab (AT&T) [21]: 400 images (40 individuals, 10 image per class); Bio ID [22]: 1520 images (23 individuals); Nottingham DB [23]: 465 images (71 individuals); JAFFE DB [24]: 213 images (10 individuals); Georgia Tech Face Database [25]: 750 images (50 individuals); Aberdeen DB [26]: 690 images (90 individuals); Caltech Faces 1999 [27]: 450 images (27 individuals); Pain Expression [28]: 599 images (23 individuals); FEI Face Database [29]: 2800 images (200 individuals);

The final database consist of *en face* portraits only, thus many original images were excluded. All images were converted to grayscale and re-scaled to 160×120 pixels. The dimensions of each facial region extracted from facial portraits are presented in Tab.1. The total number of images is equal to 3445 divided into 365 classes. Each class contains $4 - 24$ images (depending on the source database). The database was split into learning and testing datasest in a random way, so the learning set contains 3106 and the testing set 1695 images (5 distortion variants for each of 339 base images, respectively).

The testing images were altered (using simulated occlusion) and divided into 5 categories, related to the distortion type. Each category is characterized by

Table 1. Dimensions of images representing face regions and mean images

Face part	width	height	% of area
Forehead	120	53	33
Left eye	60	36	11
Right eye	60	36	11
Nose	120	26	17
Mouth/Chin	120	45	28

the area of distortion, the form of distortion and the mean information loss (in terms of the number of pixels changed in comparison to the original image). The information loss/distortion of each part of a face is presented in Tab. 2. The following categories of distortions are included in the database (the examples of simulated images representing above distortions were presented in Fig. 4):

(A) Reference images (with no distortions);
(B) Eyes covered; mean information loss: 16%;
(C) Face area covered with randomly placed object; mean information loss: 13%;
(D) Face area with lower intensity; mean information loss: 28 %;
(E) Randomly covered face area; mean information loss: 36 %.

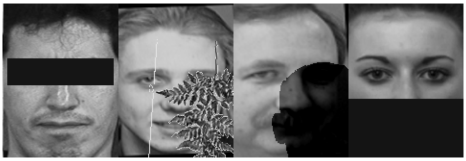


Fig. 4. Distortion examples used in the experiments (B, C, D, and E, respectively)

Table 2. Mean information loss in terms of percentage of changed (lost) pixels in each part in comparison to the each part area and the whole image, respectively

Face part	Distortion category			
	B	C	D	E
Forehead	0 (0)	1.7 (0.6)	9.2 (3.1)	39 (12.9)
Left eye	72.3 (8.1)	4.1 (0.5)	13.9 (1.6)	22 (2.5)
Right eye	71 (8)	22.7 (2.6)	48.7 (5.5)	20 (2.3)
Nose	0 (0)	22.6 (3.7)	44.3 (7.2)	33.8 (5.5)
Mouth/Chin	0 (0)	20.5 (5.8)	40.2 (11.3)	44.6 (12.5)

We performed recognition employing 16, 100 and 400 principal components per image region (80, 500 and 2000 PCs per face image, respectively). In order to evaluate the performance of our method, we additionally performed a recognition of whole images employing 100 and 400 PCs. The results of recognition of sample images are presented in Fig.5, where *query* means testing image, *reconstruction* represents image composed of individually recognized regions, *result(1)* represents our algorithms while *result(2)* shows recognition by whole image approach. The group of pictures on the left hand side show the results where the accuracies of region-based and whole image-based recognition are equal. On the right hand side of the figure, the results of region-based approach are superior to whole image-based approach.

The comparative results are presented in Fig.6. As it can be seen, the recognition of undistorted faces involving whole images - case (A) - gives the highest accuracy (97.6% vs. 95%). In case of portraits with eyes covered with black rectangle - case (B) - presented method of recognition provides lower rate (83.8%

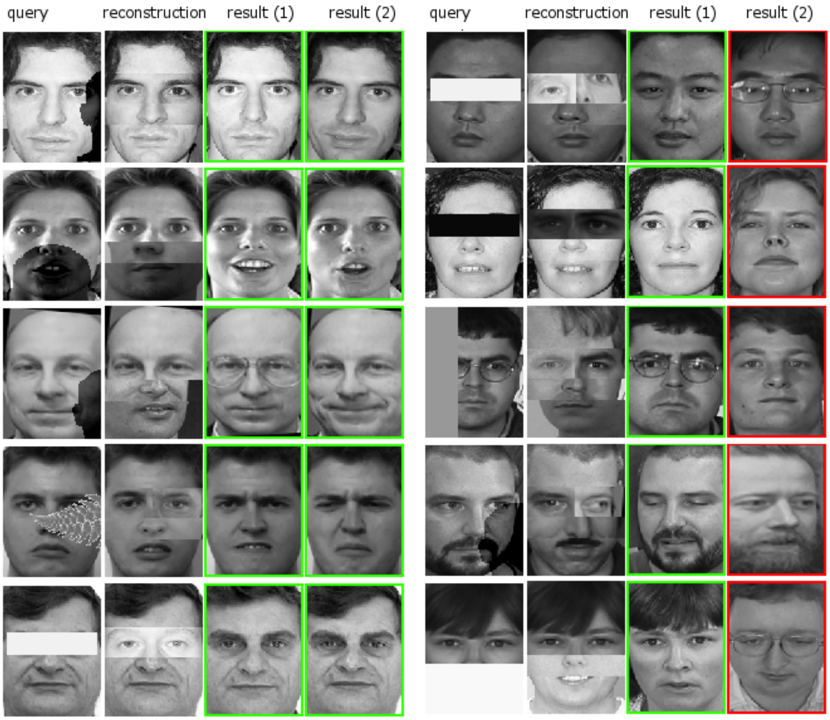


Fig. 5. Sample recognition results

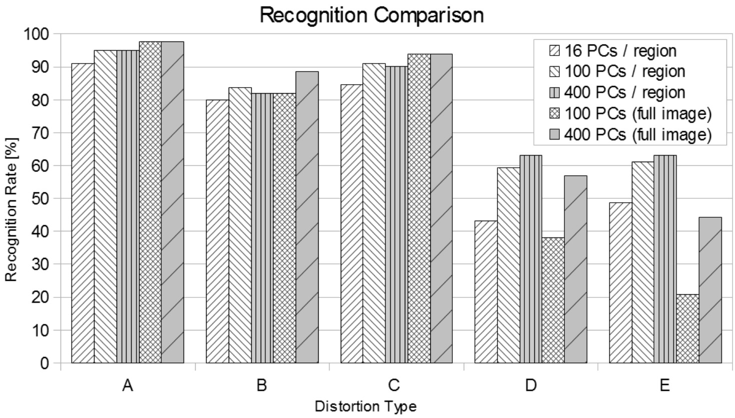


Fig. 6. Comparison of recognition performance for different variants of methods

vs. 88.5%). Images covered with independent object - case (C) - give the highest recognition rate for whole-image approach (93.8% vs. 90.9%). The rest two cases (D) and (E), where the information loss is the highest show the true potential of presented method. The recognition rates for these distortion types and region-based processing are equal to 63.1% vs. 56.9% and 44.2%, respectively.

4 Summary

We presented a novel algorithm of recognizing facial portraits under local distortions (occlusion, shadows, intensity change). It incorporates face normalization, extraction of individual face regions, dimensionality reduction and classification in multiple sub-spaces. The final result is obtained by a voting with a majority principle. The results of experiments performed on large database show the superiority of developed method over traditional approach, mainly in case of large information loss. In case of images not influenced by above distortions, the recognition rate is similar. Although, we did not investigated the influence of low light intensity or degraded image quality (in case of low resolution and low sensitivity cameras), we think the presented method based on PCA is able to cope with such problems (as it has been proved in the literature many times before). In future, another extension of 2DPCA may be employed, namely the processing of colour images [30], in order to make the whole approach more robust and capable of distinguishing between facial and non-facial features.

References

1. Tan, X., Triggs, B.: Preprocessing and Feature Sets for Robust Face Recognition. In: IEEE Conf. on Computer Vision and Pattern Recognition - CVPR 2007, pp. 1–8 (2007)
2. Choraś, M., Andrysiak, T.: Symmetry-Based Salient Points Detection in Face Images. In: Rutkowski, L., Tadeusiewicz, R., Zadeh, L.A., Żurada, J.M. (eds.) ICAISC 2006. LNCS (LNAI), vol. 4029, pp. 758–767. Springer, Heidelberg (2006)
3. Zhang, T., Fang, B., Tang, Y.Y., Shang, Z., Li, D., Lang, F.: Multiscale facial structure representation for face recognition under varying illumination. Pattern Recognition 42(2), 251–258 (2009)
4. Chen, W., Er, M.J., Wu, S.: PCA and LDA in DCT domain. Pattern Recognition Letters 26, 2474–2482 (2005)
5. Choi, S., Choi, C.-H., Kwak, N.: Face recognition based on 2D images under illumination and pose variations. Pattern Recognition Letters 32, 561–571 (2011)
6. Sirovich, I., Kirby, M.: Low-dimensional procedure for the characterization of human faces. Journal of Opt. Soc. Am. 4, 519–524 (1987)
7. Turk, M., Pentland, A.: Eigenfaces for Recognition. Journal of Cognitive Neurosicence 3(1), 71–86 (1991)
8. Swets, D.L., Weng, J.: Using Discriminant Eigenfeatures for Image Retrieval. IEEE Trans. Pattern Analysis and Machine Intelligence 18, 831–836 (1996)
9. Tsapatsoulis, N., Alexopoulos, V., Kollias, S.: A Vector Based Approximation of KLT and its Application to Face Recognition. In: Proceedings of the IX European Signal Processing Conference, EUSIPCO 1998, Island of Rhodes, Greece, pp. 1581–1584 (1998)

10. Yang, J., Zhang, D., Frangi, A.F., Yang, J.-Y.: Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition. IEEE Trans. Pattern Anal. Mach. Intell. 26(1), 131–137 (2004)
11. Nagabhushan, P., Guru, D.S., Shekar, B.H.: Visual learning and recognition of 3D objects using two-dimensional principal component analysis: A robust and an efficient approach. Pattern Recognition 39(4), 721–725
12. Chen, S., Zhu, Y., Zhang, D., Yang, J.-Y.: Feature extraction approaches based on matrix pattern: MatPCA and MatFLDA. PRL 26, 1157–1167 (2005)
13. Storer, M., Roth, P.M., Urschler, M., Bischof, H.: Fast-Robust PCA. In: Salberg, A.-B., Hardeberg, J.Y., Jenssen, R. (eds.) SCIA 2009. LNCS, vol. 5575, pp. 430–439. Springer, Heidelberg (2009)
14. Hosoi, T., Nagashima, S., Ito, K., Aoki, T.: Reconstructing occluded regions using fast weighted PCA. In: 19th IEEE International Conference on Image Processing (ICIP), pp. 1729–1732 (2012)
15. Tarrés, F., Rama, A., Torres, L.: A Novel Method for Face Recognition under Partial Occlusion or Facial Expression Variations. In: 47th International Symposium ELMAR 2005. Multimedia Systems and Applications, Zadar, pp. 163–166 (2005)
16. Min, R., Hadid, A., Dugelay, J.: Improving the recognition of faces occluded by facial accessories. In: IEEE International Conference on Automatic Face Gesture Recognition and Workshops (FG 2011), pp. 442–447 (2011)
17. Park, J.-S., Oh, Y.H., Ahn, S.C., Lee, S.-W.: Glasses removal from facial image using recursive error compensation. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(5), 805–811 (2005)
18. Kukharev, G., Forczmański, P.: Data Dimensionality Reduction for Face Recognition. Machine Graphics and Vision 13(1/2), 99–122 (2004)
19. Gokberk, B., Dutagaci, H., Ulas, A., Akarun, L., Sankur, B.: Representation plurality and fusion for 3-D face recognition. Trans. on Systems Man and Cybernetics, Part B 38(1), 155–173 (2008)
20. Martinez, A.M., Benavente, R.: The AR Face Database. CVC Technical Report #24 (June 1998)
21. AT&T Laboratories Cambridge. Database of Faces, `http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html`
22. BioID-Technology Research. The BioID Face Database, `http://www.bioid.com/downloads/facedb/index.php`
23. Nottingham DB, `http://pics.psych.stir.ac.uk/zips/nottingham_originals.zip`
24. Lyons, M.J., Akamatsu, S., Kamachi, M., Gyoba, J.: Coding Facial Expressions with Gabor Wavelets. In: Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition, Nara Japan, April 14-16, pp. 200–205. IEEE Computer Society (1998)
25. Georgia Tech face database, `http://www.anefian.com/research/face_reco.htm`
26. Abeerden, `http://pics.psych.stir.ac.uk/zips/Aberdeen.zip`
27. Caltech Faces 1999 (1999), `http://www.vision.caltech.edu/Image_Datasets/faces/faces.tar`
28. Pain expression, `http://pics.psych.stir.ac.uk/zips/pain.zip`
29. The FEI face database, `http://fei.edu.br/~cet/facedatabase.html`
30. Forczmański, P.: Comparison of Tensor Unfolding Variants for 2DPCA-Based Color Facial Portraits Recognition. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) ICCVG 2012. LNCS, vol. 7594, pp. 345–353. Springer, Heidelberg (2012)

Mahalanobis Distance-Based Algorithm for Ellipse Growing in Iris Preprocessing

Krzysztof Misztal[1] and Jacek Tabor[2]

[1] AGH University of Science and Technology
Faculty of Physics and Applied Computer Science
al. A. Mickiewicza 30, 30-059 Kraków, Poland
Krzysztof.Misztal@fis.agh.edu.pl
[2] Jagiellonian University
Faculty of Mathematics and Computer Science
Łojasiewicza 6, 30-348 Kraków, Poland
tabor@ii.uj.edu.pl

Abstract. We introduce a new algorithm for ellipse recognition. The approach uses Mahalanobis distance and statistical and analytical properties of circular and elliptical objects. At first stage of the algorithm the starting configuration of initial ellipse is defined. Next we apply a condition which describes how much the shape is ellipse-like on the boundary points.

The algorithm can be easily applied to detection of elliptical objects also on grayscale images. Moreover, we discuss the improvement in iris image preprocessing.

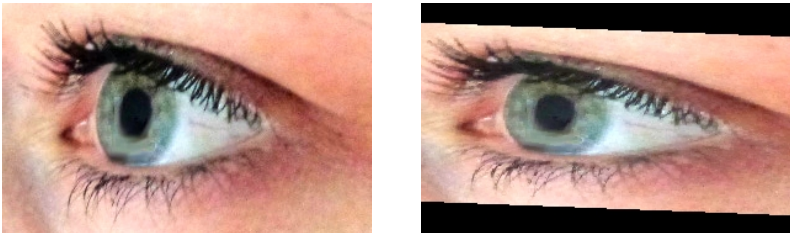
Keywords: Mahalanobis distance, ellipse growing, ellipse detection, pattern recognition, feature extraction.

1 Introduction

Efficient ellipse and circle detection is one of the key tasks in image processing which is widely applied in various computer vision problems, in particular in the computation of the position of 3-D objects in robotic applications [1, 2], in the eye tracking in human-computer interfaces [3], in face detection in biometric identification [4], in character recognition [5]. Consequently the extraction of elliptic shapes form images has captured the interest of researchers for a long time [6]. The methods used for this task can be categorized into several groups. The most widely used is based on Hough transform [7] which takes edge map of image as an input. Other group is based on the least square methods which mostly cast the ellipse fitting problem into a constrained matrix equation problem [8, 9]. Next category uses neural networks to find fast and approximately sufficient solutions [10]. The last group is focused on hybrid approach which makes it more flexible and efficient in many cases [11–15].

In this paper we introduce a new algorithm based on ellipse-growing idea which may help in iris preprocessing, and consequently in iris pattern recognition. A basic limitation of the current iris recognition methods [16] is that they

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 158–167, 2013.



(a) (b)

Fig. 1. Iris image reconstruction. Fig. 1a – original eye image with extracted iris and pupil. After the reconstruction (Fig. 1b) the shape of the iris was changed and it looks like a circle.

require an "on-axis" image of the eye. Clearly in most "real-life" pictures we have only a side-view of the eye, see Fig. 1a, which consequently yields that the eye resembles an ellipse instead of a circle. There are same approaches to deal with this problems which are based on the change in the representation, see [17–19].

We use slightly different approach which allows to use original Daugman's recognition process – namely we modify the picture by respective affine transformation so that the iris becomes circle-shaped. To do so we fit an optimal ellipse to the pupil of an eye, and apply to the picture the affine operation which transformes this ellipse into a circle, see Fig. 1b, which consequently transforms the iris almost into a circle.

To apply the above mentioned procedure we construct a new ellipse extraction method which is based on the Mahalanobis distance and uses the statistical and analytical properties of circular and elliptical objects. In the first step we use thresholding for finding the starting configuration of cluster - initial ellipse shape[1]. Next we use the specified condition to decide whether to add or not the points from the cluster boundary. Since during the calculation we examine just the boundary points of our cluster the calculation proceeds relatively fast. This condition verifies whether the current boundary element fits (with possible error δ) into the optimal ellipse fitted to the data with the use of Mahalanobis distance. More precisely, we add the boundary point x to the data-cluster C if

$$\|x-\mu_C\|_{\Sigma_C} = \sqrt{(x-\mu_C)^T \Sigma_C^{-1}(x-\mu_C)} \le 2+\delta,$$

where μ_C and Σ_C denote the mean and covariance of C. The procedure is repeated until no boundary point belonging to the set satisfies the condition. The main advantage of the proposed method is that no complicated mathematical computation is involved in the implementation. Moreover, it is suitable for ellipse growing and ellipse extraction even on grayscale digital images and can be used for higher dimensional data.

[1] This step can be omitted or replaced by other procedure which ends with proper initial configuration of the cluster.

The remainder of the paper is organized as follows: in the next section we present basics of the Mahalanobis distance. Moreover we provide a natural and intrinsic characterization of elliptical shapes, which is proper even in higher dimensions. In the third section we look more closely at the outcome information from the growing cluster. We use its characterization to correct the image. In section 4 we provide the results of experiments to illustrate the performance of our method. Finally, the last section contains some concluding remarks and possible directions of future investigations.

2 Theoretical Background

In this section for the convenience of the reader we present a brief exposition of the Mahalanobis distance and indicate how these information can be used to construct ellipse growing algorithm.

It is well-known that the Euclidean distance between two points x and y in real space is given by

$$\|x-y\| = \sqrt{(x-y)^T \mathbb{I}(x-y)}, \text{ for } x, y \in \mathbb{R}^N.$$

where T denotes the transpose operation and $\mathbb{I}$ is an identity matrix.

It follows immediately that all points with the same distance from the origin $\|x-0\| = c$ satisfy $x_1^2 + \ldots + x_n^2 = c^2$, which means that all components of an observation x contribute equally to the Euclidean distance of x from the center. But this situation in same cases is not optimal as we often prefer a distance such that components with high variability should have different weight than components with low variability. This can be obtained by using the Mahalanobis distance [20] defined for a positive definite matrix Σ by

$$\|x-y\|_{\Sigma} = \sqrt{(x-y)^T \Sigma^{-1}(x-y)}. \tag{1}$$

In this case the set of all points with the same distance c from a given point x_0

$$\{x \in \mathbb{R}^N | (x-x_0)^T \Sigma (x-x_0) = c^2\}, \tag{2}$$

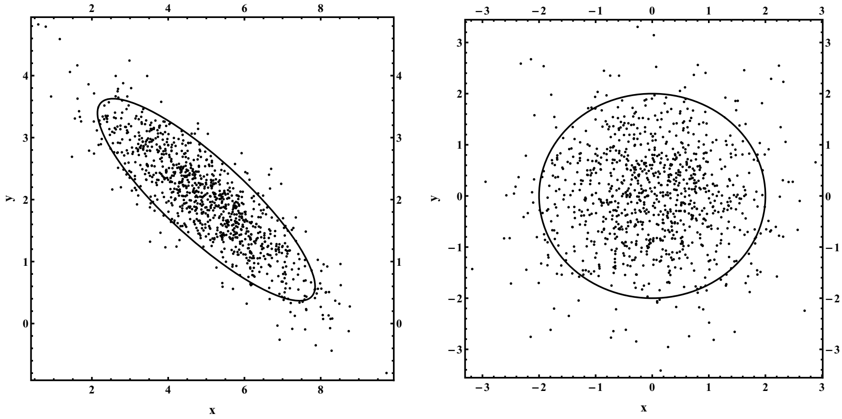
describes and ellipsoid with center at x_0. If $x_0 = 0$ then (2) is the general equation of ellipsoid centered at the origin.

One can easily see that instead of calculating the Mahalanobis distance $\|x-y\|_{\Sigma}$ we can equivalently transform the points by the matrix $\Sigma^{-1/2}$ and compute the Euclidean distance of transformed points:

$$\|x-y\|_{\Sigma} = \|\Sigma^{-1/2}x - \Sigma^{-1/2}y\|,$$

see Fig. 2.

Let C denote the given subset of $\mathbb{R}^N$. By μ_C we denote the mean value of C, and by Σ_C we mean covariance matrix of C. If we want to fit the Mahalanobis distance to our data set C, as Σ we take the covariance matrix Σ_C of C.



(a)

(b)

Fig. 2. Mahalanobis distance vs. Euclidean distance. Fig. (a) – Mahalanobis distance in the original space on the data C, Fig. (b) – Euclidean distance in space transformed by the operation $x \to \Sigma_C^{-1/2}(x - \mu_C)$.

Remark 1. It is worth noting that Mahalanobis distance can be treated as a Euclidean distance in a transformed space, where the mean of the data C is transformed into zero:

$$C \ni x \to \Sigma_C^{-1/2}(x - \mu_C) \in \mathbb{R}^N.$$

Fig. 2 presents the modification of origin given by this operation.

We are going to present some important observations crucial in the construction of our algorithm with the use of Mahalanobis distance. Consider the data uniformly distributed over the circle [2] $\mathbb{B}(0, R) := \{x \in \mathbb{R}^2 \colon \|x\| \leq R\}$ of radius $R \geq 0$.

Remark 2. Consider the circle $\mathbb{B}(x_0, R)$ with radius R on the plane. Then the covariance matrix Σ of the uniform probability distribution on $\mathbb{B}(x_0, R)$ is given by

$$\Sigma = \frac{R^2}{4}\mathbb{I}. \tag{3}$$

Proof of this remark is quite simple and based on the change into polar coordinates (the N-dimensional version can be obtained by spherical representation of N-ball [21, Chap. 8, Thm. 4]).

Next theorem is essential for algorithm construction.

Theorem 3. *Consider the uniform probability density on the ellipse $E \subset \mathbb{R}^2$ with covariance Σ_E. Then*

$$E = \mathbb{B}_{\Sigma_E}(\mu_E, 2). \tag{4}$$

[2] We consider Euclidean norm in this paper.

Proof. By applying the transformation described in Remark 1 we can reduce our reasoning to the case when $E = \mathbb{B}(0, R)$. Then by (3)

$$\begin{aligned}\mathbb{B}_{\Sigma_E}(0,2) &= \{x : \|x\|_{\Sigma_E} \leq 2\} \\ &= \{x : \|x\|^2_{\Sigma_E} \leq 4\} = \{x : x^T(\tfrac{R^2}{4}\mathbb{I})^{-1}x \leq 4\} \\ &= \{x : x^T x \leq R^2\} = \mathbb{B}(0,R) = E.\end{aligned}$$

Fig. 3 presents the the given data-sets C (or more precisely the uniform density on the data) with fitted ellipses given by formula (4) in the presented objects.

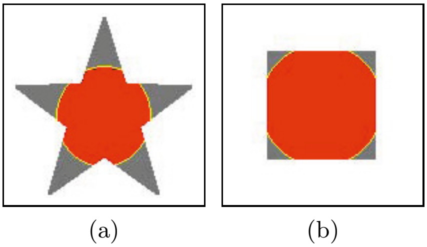

(a) (b)

Fig. 3. Different data-sets with ellipse constructed by equation (4)

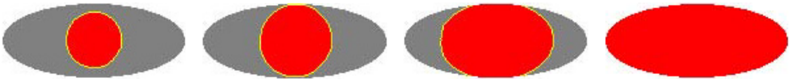
Observe that on the true ellipse E, by Theorem 3 the set E will coincide with $\mathbb{B}_{\Sigma_E}(\mu_E, 2)$. Thus two allow the increase, or more precisely, the grow, of the ellipse we allow error level $\delta > 0$. In other words we allow to add a point to our data-cluster C if it satisfies the condition

$$x \in \mathbb{B}_{\Sigma_C}(\mu_C, 2 + \delta).$$

With different values of δ we will control how close to an ellipse we want to remain. Consequently, the complete version of the authors' algorithm can be described as follows:

```
initial conditions
  choose $\delta > 0$
  choose initial configuration of initial ellipse $C$
  compute mean value $\mu_C$ and covariance matrix $\Sigma_C$ of $C$
repeat
  added ← False
  for each $x$ in $\partial C$ do
    if $x \in \mathbb{B}_{\Sigma_C}(\mu_C, 2 + \delta)$ then
      $C := C \cup \{x\}$
      compute mean value $\mu_C$ and covariance matrix $\Sigma_C$ of $C$
      added ← True
    end if
  end for
until added
```

The algorithm finds the maximal "almost" ellipse (with given error δ) fitting in the input image by growing set C (Fig. 4).



(a) initial configuration (b) 9th iteration (c) 25th iteration (d) 54th iteration

Fig. 4. Fitting ellipse – authors' algorithm iterations

As we know the role of δ is how close to an ellipse we want to stay. However, to start the algorithm we need an explicit construction of the initial configuration of C – we usually select sufficiently large ellipse which fits into our data. To do so we apply the thresholding and some morphological operations (see Fig. 8).

Another important thing is the meaning of the boundary of the cluster ∂C – we understand by this the nearest point which are not the members of the cluster C. In case of digital image we have natural discretization of our space, so we can easily determine the boundary using von Neumann neighborhoods (see Fig. 5).

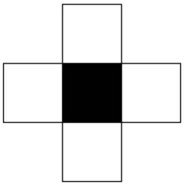


Fig. 5. The boundary points of the black pixel used in our algorithm

Finally, we can optimize the calculation of the mean value and covariance matrix of cluster C. Following well-known remark shows the formula for on-line calculation of the mean and covariance of the modified data.

Remark 4. Let U, V be subsets of $\mathbb{R}^2$, $U \cap V = \emptyset$. We put

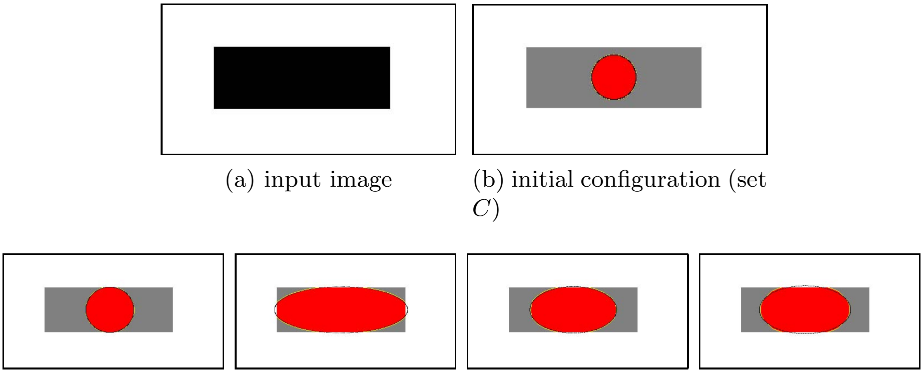
$$w_{U\cup V} = w_U + w_V, \qquad p_U = \frac{w_U}{w_{U\cup V}}, \qquad p_V = \frac{w_V}{w_{U\cup V}}.$$

where w_U and w_V denote the weights (cardinalities) of the sets U and V respectively. Then mean value and covariance matrix of set $U \cup V$ are given by

$$\mu_{U\cup V} = p_U \mu_U + p_V \mu_V,$$

$$\Sigma_{U\cup V} = p_U \Sigma_U + p_V \Sigma_V + p_U p_V (\mu_U - \mu_V)(\mu_U - \mu_V)^T.$$

The algorithm presented above has the ability to fit the maximal elliptical object in the image. It uses calculated value of Mahalanobis distance with the given error-level δ. Clearly, the error level can change the performance of the algorithm. Fig. 6 presents the output of the algorithm for different values of δ.



(a) input image

(b) initial configuration (set C)

(c) $\delta = 0.01$

(d) $\delta = 0.02$

(e) $\delta = 0.03$

(f) $\delta = 0.1$

Fig. 6. Different level of δ for the same image may give different results. Fig. 6a – input image. Fig. 6b initial configuration. Points added to cluster at the earlier stage of the algorithm change the result.

Let now look at the starting configuration of the cluster. Fig. 7 presents the output of the algorithm for different initial configuration of cluster (the initial choice of set C). In most cases we obtain almost the same results. However there is a possibility to damage the algorithm effect as is presented at Fig. 7d and 7h, where we start with initial set C given by an interval. Then the algorithm changes just the horizontal size of the cluster while the vertical stay unchanged.

3 Experimental Results

In this section we present the two connected examples of the algorithm outcome for binary and grayscale iris image formats.

Binary Image. In this case we apply our algorithm for thresholded image[3] – see first row at Fig. 8. The fitted ellipse contain the foreground of the image (marked on black with weight equals 1), since we use just the information obtained after thresholding. Fig. 8a and Fig. 8b presents the initial and outcome ellipse respectively. Fig. 8 shows the outcome of the algorithm (boundary of the fitted ellipse) at the original image.

[3] We use same morphological operations for remove out-layers and reduce noise.

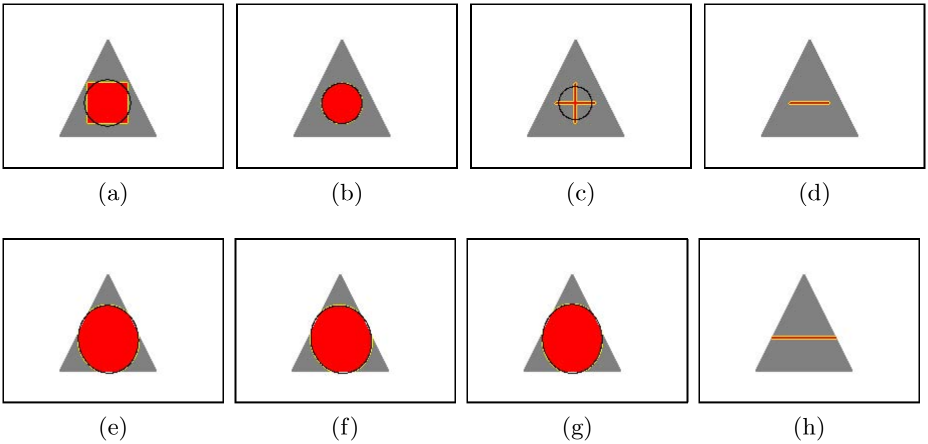


Fig. 7. Different initial configuration of the algorithm may give different outcome. The first row present the initial configuration, while the second row – the algorithm outcome.

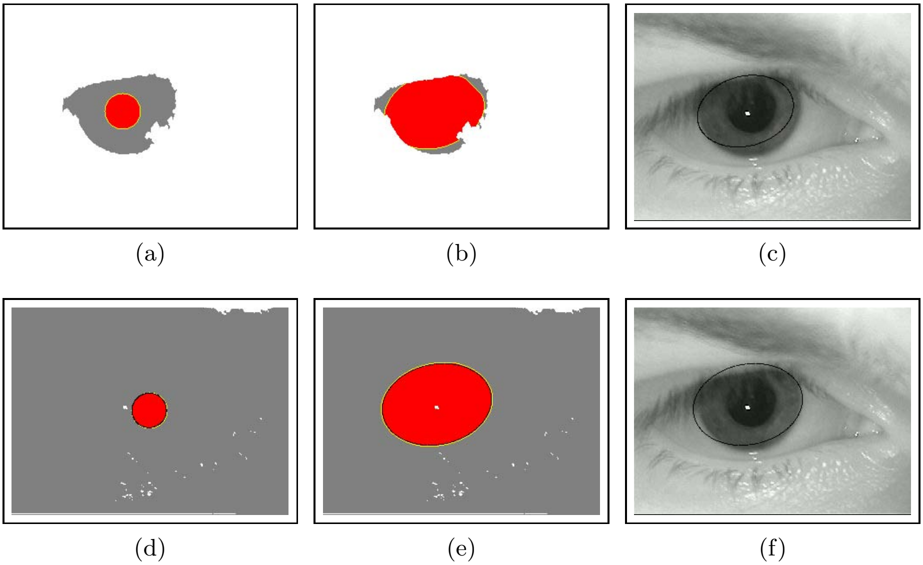


Fig. 8. Application of the authors' algorithm for detection of the pupil and iris at the grayscale images [22] for binary images (first row) and grayscale images: (a), (d) – initial configuration of a cluster, (b), (e) – final size of the cluster, (c), (f) – fitted ellipse at original image

Grayscale Image. In this case we also use the thresholded image for finding the initial ellipse. However in the next steps of the algorithm we use information about the color in each pixel added to the ellipse. This approach is better because we use full information we have and obtained result is more natural.

More advanced example is presented at Fig. 1. As a outcome of the authors' algorithm we obtain the fitted ellipse, which statistical description can be used to define the transformation operation (compare Remark 1). Thus we can transform the iris image to make it more convenient for further processing by iris pattern recognition algorithms.

4 Conclusions

We presented a new method for ellipse growing based on properties of Mahalanobis distance, which can be used in detection of ellipses in: both binary and grayscale images. The numerical experiments have demonstrated that the algorithm works sufficiently well in both cases. We can apply the algorithm for iris preprocessing in real-life pictures where eyes are photographed sideways. Moreover, we can easily adapt the approach to higher dimensional data (for example in 3D medical images).

In the future work we plan to focus on the automation of the algorithm, in particular in the automatic detection of the necessary parameters (e.g. δ level) and image preprocessing methods (e.g. morphological operations). Furthermore we plan a more complete analysis of the behavior of the algorithm performance.

Acknowledgments. This work was supported by the Polish National Centre of Science Grant No. 2012/07/N/ST6/02192.

References

1. Nguyen, T.M., Ahuja, S., Wu, Q.J.: A real-time ellipse detection based on edge grouping. In: IEEE International Conference on Systems, Man and Cybernetics, SMC 2009, pp. 3280–3286. IEEE (2009)
2. Greggio, N., Manfredi, L., Laschi, C., Dario, P., Carrozza, M.C.: Robotcub implementation of real-time least-square fitting of ellipses. In: 8th IEEE-RAS International Conference on Humanoid Robots, Humanoids 2008, pp. 174–181. IEEE (2008)
3. Takegami, T., Gotoh, T., Ohyama, G.: An algorithm for model-based stable pupil detection for eye tracking system. Systems and Computers in Japan 35, 21–31 (2004)
4. Jain, V., Learned-Miller, E.: Fddb: A benchmark for face detection in unconstrained settings. University of Massachusetts, Amherst, Tech. Rep. UM-CS-2010-009 (2010)
5. Jin, Y., Qiu, K., Dai, Y., Xiao, G., Deng, H.: An improved handwritten Chinese character recognition based on localized ellipse model. In: 2010 3rd International Congress on Image and Signal Processing (CISP), vol. 4, pp. 1803–1807. IEEE (2010)

6. Wong, C., Lin, S., Ren, T., Kwok, N.: A survey on ellipse detection methods. In: 2012 IEEE International Symposium on Industrial Electronics (ISIE), pp. 1105–1110. IEEE (2012)
7. Illingworth, J., Kittler, J.: A survey of the hough transform. Computer Vision, Graphics, and Image Processing 44, 87–116 (1988)
8. Fitzgibbon, A., Pilu, M., Fisher, R.B.: Direct least square fitting of ellipses. IEEE Transactions on Pattern Analysis and Machine Intelligence 21, 476–480 (1999)
9. Prasad, D.K., Leung, M.K.: Methods for ellipse detection from edge maps of real images. Machine Vision-Applications and Systems, 135–162 (2012)
10. Chiang, C.C., Ho, M.C., Liao, H.S., Pratama, A., Syu, W.C.: Detecting and recognizing traffic lights by genetic approximate ellipse detection and spatial texture layouts. International Journal of Innovative Computing, Information and Control 7, 6919–6934 (2011)
11. Dufrenois, F.: Ellipse fitting with uncertainty and fuzzy decision stage for detection. Application in videomicroscopy. In: Benferhat, S., Besnard, P. (eds.) ECSQARU 2001. LNCS (LNAI), vol. 2143, pp. 432–443. Springer, Heidelberg (2001)
12. Prasad, D.K., Leung, M.K.: A hybrid approach for ellipse detection in real images. In: Second International Conference on Digital Image Processing, p. 75460I. International Society for Optics and Photonics (2010)
13. Chia, A.S., Rahardja, S., Rajan, D., Leung, M.K.: A split and merge based ellipse detector with self-correcting capability. IEEE Transactions on Image Processing 20, 1991–2006 (2011)
14. Mai, F., Hung, Y., Zhong, H., Sze, W.: A hierarchical approach for fast and robust ellipse extraction. Pattern Recognition 41, 2512–2524 (2008)
15. Tabor, J., Misztal, K.: Detection of elliptical shapes via cross-entropy clustering. In: Sanches, J.M., Micó, L., Cardoso, J.S. (eds.) IbPRIA 2013. LNCS, vol. 7887, pp. 656–663. Springer, Heidelberg (2013)
16. Daugman, J.G.: High confidence visual recognition of persons by a test of statistical independence. IEEE Transactions on Pattern Analysis and Machine Intelligence 15, 1148–1161 (1993)
17. Daugman, J.: New methods in iris recognition. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 37, 1167–1175 (2007)
18. Chun, C.-N., Chung, R.: Iris recognition for iris tilted in depth. In: Petkov, N., Westenberg, M.A. (eds.) CAIP 2003. LNCS, vol. 2756, pp. 530–539. Springer, Heidelberg (2003)
19. Dorairaj, V., Schmid, N.A., Fahmy, G.: Performance evaluation of non-ideal iris based recognition system implementing global ica encoding. In: IEEE International Conference on Image Processing, ICIP 2005, vol. 3, p. III–285. IEEE (2005)
20. Mahalanobis, P.C.: On the generalized distance in statistics. Proceedings of the National Institute of Sciences of India, New Delhi 2, 49–55 (1936)
21. Königsberger, K.: Analysis 2. Springer (2004)
22. Institute of Automation, Chinese Academy of Science: CASIA iris image database (2013), `http://www.cbsr.ia.ac.cn/IrisDatabase.htm`

The Data Exploration System for Image Processing Based on Server-Side Operations

Magdalena Ładniak, Adam Piórkowski, and Mariusz Młynarczuk

Department of Geoinformatics and Applied Computer Science,
AGH University of Science and Technology,
al. Mickiewicza 30, 30-059 Cracow, Poland
mladniak@geol.agh.edu.pl,
{pioro,mlynar}@agh.edu.pl
http://www.geoinf.agh.edu.pl

Abstract. In this paper the possibilities for construction of an ad hoc search system to examine large-sized raster image data sets, e.g. rock images or medical images, for analysis of its characteristic parameters are presented. A new solution for image exploration based on any attributes extracted with computer image analysis by using extensions for server-side operations is proposed.

Keywords: image mining, image processing, databases, user defined functions, image clusterization.

1 Introduction to Image Exploration

Most of scientific disciplines belonging to natural sciences are based on experimental data. Although theories, obviously stimulate the development of new branches, but in doubtful cases, experimental verification is of pivotal importance [1]. In recent years an increasing need for supporting measurements of rock structures with image analysis methods is observed. These methods are being successfully implemented into researches related to geology [2], mining [3] and rock mechanics [4]. Image data exploration is a current issue. This includes a number of approaches to image mining [6, 7]. However, many authors often present no solutions enabling computations and restrict themselves to theoretical deliberations. A classification scheme for medical images has been proposed in [8], but this solution requires image processing beyond a database that may pose a problem when integrating with such data exploration systems as Weka or Statistica.

The aim of this paper is to consider a possibility for construction of a system for mining a large-sized set of rock microsections by using database-side processing. Such solution avoids a complex construction of heterogenic environment. The weak points of the existing solutions are strong dependence on interfacing with various versions of different software that can make uniform application impossible, especially when the software is updated. The use of external programs for image processing might cause difficulties with file arrangement, description and ordering in the case of large amount of photos. That is why the Authors propose

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 168–176, 2013.

a system that provides data-base side raster image storage and analysis, thus allowing data exploration with the data mining software commonly used. Research on preliminary image clasterization by the describing parameters is proposed.

2 An Approach to Image Selection Using Server-Side Operations

The literature, which is the Authors known, on this subject is limited to the description of the image processing only on the local computer and using images in the computer local memory or imported from databases. The Authors think that the future of image analysis involves the calculation of the total transfer of database servers. However, desktop computers (in the form of desktops, laptops, tablets, smartphones etc.) will be used only to manage the calculations and to review the obtained results.

2.1 The Idea of Data Processing and Analysis by Using Database Technologies and Statistical Tools

Database queries can be processed both with a database-side engine and from the statistical tool level. At baseline the database contains only raw images with a basic description like for example their names. An effect of such query is a set of numerical data and its further analysis depends only on analyst needs. Based on observations he/she can request the database to make an additional measurement for a selected feature and to store the result permanently or temporarily in the database. Such solution enables easy and direct extraction of desired information without the use of surplus software. Information flow and schematic context of analysis for the presented example is presented below (Figure 1). The detailed description of the structure and performance issues of the proposed system are available in the literature [9].

The use of the mechanism of User Defined Functions in a database environment gives the user complete freedom in creating his own image processing operations and advanced algorithms. These algorithms were created by the Authors for the purpose of this study. Absolute freedom in designing a database schema should be emphasized here. The stored images can be taken out by DMBS at the user's request, processed and the results can be placed into a result table. Any number of operations can be involved each time in processing. This may be either a single operation that builds a complex algorithm or a set of cyclic image processing operations. Any number of input images can be processed. In addition, it is possible to write images being partial solutions in the database. The mechanisms used allows easy reference also to these images what is of utmost importance to the creation process and verification of proper operation of new image processing algorithms. The application of the proposed solution enables construction of knowledge base of very large number of parameters. This can be used in the next step in statistical data analysis.

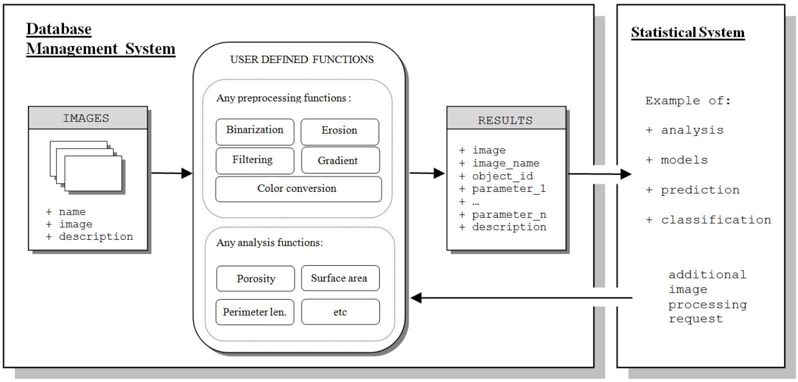


Fig. 1. Information flow and schematic context of analysis

There are many tools for the whole process of converting data into usable knowledge. In study described here Weka (Waikato Environment for Knowledge Analysis) was used. This is an open source software under General Public License. The possibility of direct communication with a database system and graphic presentation of data were utilized in this software. Thanks to provided communication with a database the user's request for image processing is sent directly by doing SQL (Structured Query Language) queries. In practice this indicates a capability to run image analysis routines directly from a statistical tool when the user sees a potential need for an additional analysis. This approach is well known in methods of gaining knowledge from a sets of numerical data, for example in time series analysis [10].

2.2 Combination of Search Criteria

It is often necessary to make a selection based on many criteria concurrently. An example is finding the images for which a group of parameters fulfills the specified standard conditions. The manual solution of this task induces a series of operations to be executed connected with switching between various solutions, e.g. 'get from the database, batch convert, save results to a file, import to statistical tool'. The Authors propose to move such operations onto database platforms, thus allowing uniform combination of search criteria. This consists of constructing appropriate conditions to narrow and group the query result. The solution unifies the data storage method, avoid problems related to entering various access paths to images, systemizes places where addition information on computed parameters are stored and allows return to results at the selected stage of analysis.

2.3 Input Data

The input data in the system under consideration are images of rock specimen microsections made by the Authors. The process of test material preparation consisted of taking a rock specimen, cutting it into thin discs, impregnating with an adhesive with pigment addition, polishing, taking a photo with a digital camera integrated with a microscope at 100x magnification [11]. The impregnation process contributed in gaining clear color marking of pore spaces, its shape and size [12]. This fact was the basis for selection of segmentation algorithm based on color analysis.

2.4 Preprocessing

Image processing is a well known dynamically developing discipline, so its basic issues were omitted here, while referencing to the existing publications [13–18]. In computer image analysis the purpose of transformations is most often to gain a properly segmented binary image. To obtain such image a series of transformations of the input image is carried out. As a result a contrast imaging is obtained where objects (e.g. potential pore cross sections) are clearly distinguishable from the background [14].

Threshold Binarization. To prepare a binary image two segmentation algorithms were tested by using threshold binarization:

- In the first algorithm, the thresholds were chosen empirically based on color analysis and the YIQ color space model was employed. As an input signal represent the chrominance information (I channel) was used.
- In the second algorithm, the thresholds were chosen by maximum entropy method [16].

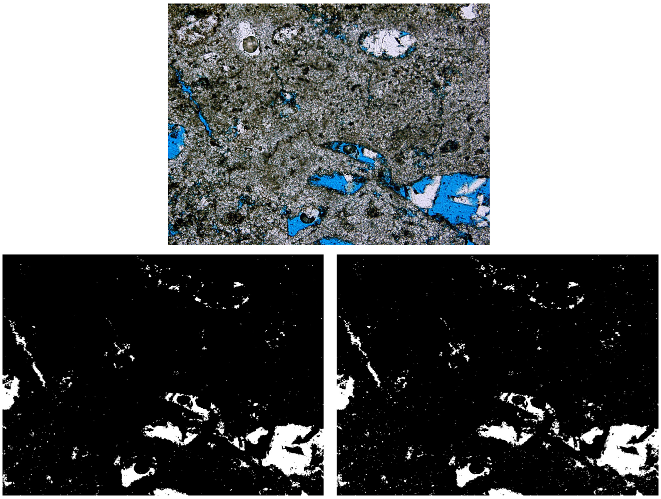


Fig. 2. Sample results of binarization by using methods 1 and 2

For further analyses the results of binarization with manually chosen thresholds were used because of better results of the method.

Noise Reduction. The next stage of preprocessing was to remove the smallest objects that did not indicate any pore voids. The feature of microscopic measurements is that they are made at a specified magnification. On this basis the object elimination criterion was established. The isolated small areas (here of few pixels in size) were removed. The boundaries of the marked areas were smoothened. To do it the erosion and dilatation operations were used. When these operations are used separately they cause different changes in surface area of objects under transformation. Erosion decreases it, while dilatation increases it. To eliminate this effect the transformations being a superposition of the previous ones, namely opening and closing were used. Opening results in removal of small objects without significant object size changes. The aim of closing is to fill in narrow nicks and small holes inside the object, while having no significant effect on its shape [19]. The set of parameters used in analyses is presented below.

2.5 Search Parameters

The starting point for making automatic measurements is proper segmentation. In the case of binary images an object should be understood as a set of pixels of brightness 1 connected to each other. Despite the fact that measuring methods are quite well defined, its limitations should be always taken into account. To convert the values of some parameters into nominative values it is necessary to do image calibration, i.e. entering information about its actual dimensions. For reasons of clarity this process was omitted in this paper [20].

- Surface area - area was calculated by summing all pixels within the object.
- Coefficient of porosity - the value of this coefficient was assumed to be the ratio of pore volume to the total volume of tested specimen expressed in percent [21].
- Number of objects - the counting process for such objects was carried out by using the labeling algorithm. This consists of viewing the binarized image line by line until a point belonging to the object is encountered. Then it is assigned a label and the values of preceding pixels are analyzed. The labeled vertices are based on associativity to neighbor values. Depending on neighborhood type it is possible to analyze both higher or left neighbor (4-connected neighborhood) or three higher and one left neighbor (8-connected neighborhood) [22].
- Average object chord length - it is possible to determine the average chord lengths at vertical and horizontal. To do it the method consisting of computing the average length of the intersection between the segmented image and the regular set of horizontal/vertical lines was used (refinement was defined symmetrically).

- Specific surface area - specific surface area is understood as the surface area of the skeleton of porous medium per the total volume of this medium [23]. The knowledge of object chord lengths allows specific surface area to be computed from the following formula known from the literature [25]:

  $$S = 4 * \frac{n}{N} \tag{1}$$

  where:
  n – number of objects (chords) being superposition of the image and the regular grid of vertical/horizontal lines
  N – length of all lines belonging to the regular grid
- Object perimeter length - to determine approximate length of object perimeters in the photographs the erosion subject image was subtracted from the original one, and then summing of the marked pixels was done.

2.6 Analysis Results

The described measuring methodology was implemented in detection of potential pore cross sections in rock images. There following rocks were selected: dolomite from Buszewo, dolomite from Baszyna, dolomite from Koscian and limestone from Santok. The photographs were taken under optical microscope equipped with a specialized CCD camera at magnification 100x [9]. The set search criterion was strictly defined values of parameters describing the material shown below in photographs. Sample result of clasterization according to criteria: origin place (Buszewo, Baszyna, Koscian, Santok) and specific surface area (Figure 3), origin place (Buszewo, Baszyna, Koscian, Santok) and number of objects Figure 4).

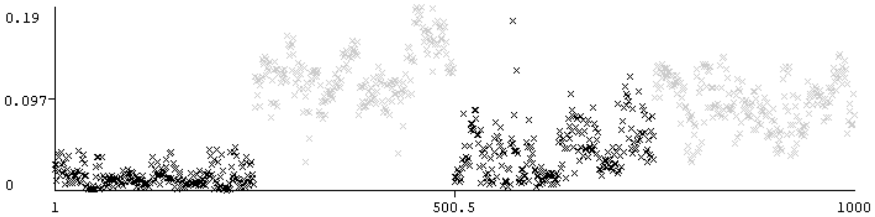


Fig. 3. Sample result of graphical clasterization, *X axis* - origin place (from the left in sequence: Santok (1-250), Koscian (251-500), Baszyna (501-750), Buszewo (751-1000)), *Y axis* - specific surface area

Efficiency. The efficiency tests were carried out on the set of 1000 photographs at the standard size of 1024x768 pixels. The Microsoft SQL Server 2012 engine was used based on the performed preminaly tests [9]. For programming the following technologies were used: Transact-SQL, and SQL CLR (CLR - Common Language Runtime) to create procedures that then to be used by the

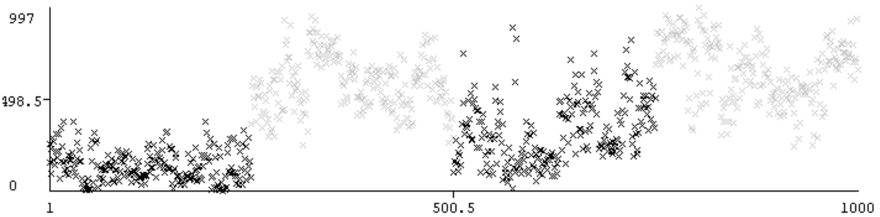


Fig. 4. Sample result of graphical clasterization, *X axis* - origin place (from the left in sequence: Santok (1-250), Koscian (251-500), Baszyna (501-750), Buszewo (751-1000)), *Y axis* - number of objects

Table 1. Function execution time for selected user defined functions

	Operations:	Times [ms]					
		TSQL		CLR			
		1	2	1		2	
		s	s	s	p	s	p
preprocessing	Erosion/Dilatation	33167	25741	156	75	64	33
	Opening/Closing	65967	34331	304	160	125	53
	Logical: AND, OR	7873	2590	12	7	5	2
	Subtracting/Summing	7827	2481	14	8	6	3
	Conversions between color spaces	9437	4321	8	5	11	6
	Threshold binarization	9388	4305	26	17	11	6
	Maximum entropy method	10451	5692	28	18	12	8
analysis	Determination of surface area	1925	975	8	6	4	2
	Determination of coefficient of porosity	1947	1036	8	5	4	2
	Determination of specific surface area	10986	4087	29	21	13	10
	Counting the number of objects	10662	6015	22	-	9	-
	Counting the chord diameter length	18302	9965	31	19	13	8

database engine as an external library [22]. The procedures were stored in The Microsoft.NET 4.5 using C# language. Two computer environments were used:

- 1. PC, HDD Seagate SATA ST3500418AS (500GB), access time 14,9 ms, CPU Intel Celeron E3200 2.40 GHz 2 Cores, 2 threads, RAM 4GB DDR2
- 2. PC, HDD Seagate 7200 RPM ST2000dl003-9VT166 (2000 GB), access time 11,9 ms, CPU Intel Core i7 2600 4 Cores, 8 threads, RAM 4GB DDR3.

The preprocessing performed by using a Transact-SQL language is an extremely time-consuming task. By adding external authorial libraries CLR to the database system and using parallel computing it was possible to speed up the research considerably, thus the system executes the tasks in an adequate time. The Parallel Class (Microsoft.NET 4.5) that provides support for parallel loops and regions was used [24].

The lead times, serial as *s* and parallel as *p*, are listed in the table below.

3 Conclusions

The tests we carried out demonstrate possibilities for construction and practical use of the system (understood as a set of user's database functions) that enables analysis to be carried out based on the results of searching by the specified parameters. The Authors' intention was to present possibilities for construction of a dedicated system to meet the needs of the specified problem related to automatic measurements of rock geological structures where parametrization plays an important role. Execution of complicated conditionally point computations in image analysis is a time-consuming task for the database server, so the Authors undertake research on timing optimization of such solutions. Execution of complicated conditionally point computations in image analysis is a time-consuming task in general [26, 27] so the Authors undertake research on timing optimization of such algorithms implemented on database servers by using more advanced methods of database systems and parallel programming. It was accepted that benefits resulting from independence from various systems, its cooperation (also in the context of versioning) are so such large that further studies are justified and should be continued.

Acknowledgement. This work was financed by the AGH - University of Science and Technology, Faculty of Geology, Geophysics and Environmental Protection as a part of statutory project.

References

1. Tadeusiewicz, R.: Data mining as a chance for relative cheap scientific discoveries obtained by looking up seemingly fully explored empirical data. Statystyka Data Mining w Badaniach Naukowych, StstSoft Polska, Krakow (2006)
2. Mlynarczuk, M.: Some Remarks on the Application of Image Analysis and Image Processing for the Description of the Geometrical Structures of Rock. Physicochemical Problems of Mineral Processing 33, 107–116 (1999)
3. Wierzbicki, M., Mlynarczuk, M.: Structural aspects of gas and dolomite outburst in Rudna copper mine, Poland. International Journal of Rock Mechanics and Mining Sciences 57, 113–118 (2012)
4. Mlynarczuk, M.: Description and classification of rock surfaces by means of laser profilometry and mathematical morphology. International Journal of Rock Mechanics and Mining Sciences 47(1), 138–149 (2010)
5. Lay, B.J.: Image processing software - multiple solutions for a single problem. In: Proceedings of the 9th European Congress on Stereology and Image Analysis, Zakopane, pp. 55–70 (2005)
6. Wang, S., Mingquan, Z., Guohua, G.: Application of fuzzy cluster analysis for medical image data mining. In: 2005 IEEE International Conference on Mechatronics and Automation, vol. 2. IEEE (2005)
7. You, F.C., Yong, B.Z.: Elliptic Object Features Extraction and Measurement in Image Data Mining. In: International Conference on Measuring Technology and Mechatronics Automation, ICMTMA 2009, vol. 1. IEEE (2009)
8. Mazurkiewicz, A., Krawczyk, H.: A parallel environment for image data mining. In: Proceedings of the International Conference on Parallel Computing in Electrical Engineering, PARELEC 2002. IEEE (2002)

9. Ladniak, M., Piorkowski, A., Mlynarczuk, M.: Structure of systems for data exploration for raster images. Studia Informatica 33(2B), 7–20 (2013)
10. Chuchro, M.: The structure of influent time series in wastewater treatment plants. Environmental Engineering III A 3rd Congress of Environmental Engineering, Lublin (2009)
11. Mlynarczuk, M.: Application of image analysis and mathematical morphology to quantitative description of rock fracture surface. Archives of Mining Sciences, Monography 3 (2008)
12. Mlynarczuk, M.: Stereological description of sedimentary rocks with methods of image analysis. Polskie Towarzystwo Mineralogiczne, Prace Specjalne 27 (2005)
13. Matheron, G.: Random Sets and Integral Geometry. Wiley, New York (1975)
14. Serra, J.: Image Analysis and Mathematical Morphology. Academic Press, New York (1982)
15. Serra, J., Mlynarczuk M.: Morphological merging of multidimensional data. In: Proc. STERMATV 2000, Cracow (2000)
16. Tadeusiewicz, R., Korohoda, P.: Komputerowa analiza i przetwarzanie obrazow. Wydawnictwo Fundacji Postepu Telekomunikacji (1997)
17. Korkosz, M., Bielecka, M., Bielecki, A., Skomorowski, M., Wojciechowski, W., Wójtowicz, T.: Improved fuzzy entropy algorithm for X-ray pictures preprocessing. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2012, Part II. LNCS, vol. 7268, pp. 268–275. Springer, Heidelberg (2012)
18. Bernaś, M.: Objects detection and tracking in highly congested traffic using compressed video sequences. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) ICCVG 2012. LNCS, vol. 7594, pp. 296–303. Springer, Heidelberg (2012)
19. Coster, M., Chermant, J.L.: Precis d'Analyse d'Images. Press du CNRS (1989)
20. Mlynarczuk, M.: Potential applications of image analysis and mathematical morphology to stereological analysis of rock structures. Archives of Mining Sciences 49, 117–140 (2004)
21. Jeske, T., Przedecki, T., Rosinski, B.: Mechanika gruntow. Panstwowe Wydawn. Naukowe (1966)
22. Radomski, P., Jarosinski, A.: Determination of specific surface area of the granular materials in aspects of its use in selected technological processes. Technical Transactions, Chemistry (10) (2010)
23. Lay, B.J.: Image processing software - multiple solutions for a single problem. In: Proceedings of the 9th European Congress on Stereology and Image Analysis, Zakopane, pp. 55–70 (2005)
24. Toub, S.: Patterns of parallel programming. Understanding and applying parallel patterns with the .Net Framework 4 and Visual C#. Parallel Computing Platform, Microsoft Corporation. Version (February 16, 2010)
25. Bodziony, J., Konstankiewicz, K., Pukos, A.: Stereology as image analysys method of agricultural materials. Int. Agrophysics 9, 293–309 (1995)
26. Bielecki, A., Buratowski, T., Smigielski, P.: Recognition of two-dimensional representation of urban environment for autonomous flying agents. Expert Systems with Applications 40, 3623–3633 (2013)
27. Bielecka, M., Skomorowski, M., Bielecki, A.: Fuzzy syntactic approach to pattern recognition and scene analysis. In: Proceedings of the 4th International Conference on Informatics in Control, Automatics and Robotics, ICINCO 2007. ICSO Intelligent Control Systems and Optimization, Robotics and Automation, vol. 1, pp. 29–35 (2007)

Image Restoration Using Anisotropic Stochastic Diffusion Collaborated with Non Local Means*

Dariusz Borkowski[1] and Katarzyna Jańczak-Borkowska[2]

[1] Faculty of Mathematics and Computer Science, Nicolaus Copernicus University,
Chopina 12/18, 87-100 Toruń, Poland
dbor@mat.umk.pl

[2] Institute of Mathematics and Physics, University of Technology and Life Sciences,
al. prof. S. Kaliskiego 7, 85-789 Bydgoszcz, Poland
kaja@utp.edu.pl

Abstract. In this paper we explore the problem of the reconstruction of images with additive Gaussian noise. In order to solve this inverse problem we use stochastic differential equations with reflecting boundary and famous non local means algorithm. Expressing anisotropic diffusion in terms of stochastic equations allows us to adapt the concept of similarity patches used in non local means. This novel look on the reconstruction problem is fruitful, gives encouraging results and compares favourably with other image denoising filters.

Keywords: stochastic anisotropic diffusion, non local means.

1 Introduction

Let D be a bounded, convex domain in $\mathbf{R}^2$, $u : \overline{D} \to \mathbf{R}$ be an original image and $u_0 : \overline{D} \to \mathbf{R}$ be the observed image of the form

$$u_0 = u + \eta,$$

where η stands for a white Gaussian noise. We assume that u and u_0 are appropriately regular. We are given u_0, the problem is to reconstruct u. This is a typical example of an inverse problem [2].

Problem of image denoising using fully automatic and reliable methods is one of the most important issues of digital image processing and computer vision. Efficient and effective reconstruction of images is an essential element of most image processing and recognising algorithms. Reconstruction algorithms allow us to make initial treatment of data for further analysis, which is very important, especially in astronomy, biology or medicine.

Various techniques were proposed to tackle this inverse problem. One may quote the linear filtering, DCT [31], wavelets theory [12, 15], variational methods [12, 23] and stochastic modelling which are generally based on the Markov field

* This research was supported by the National Science Centre (Poland) under decision number DEC-2012/07/D/ST6/02534.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 177–189, 2013.

theory and Bayesian approach [12, 16, 22]. The most important methods in the last decade in image processing have been methods driven by nonlinear diffusion equation [11, 21, 29, 30], where some papers [3–7, 17, 26, 27] involve advanced tools of stochastic analysis such as stochastic differential equations. In another class, one could include methods that take advantage of the non-local similarity of patches in the image. Among the most famous, we can name non local means (in short NL-means) [8, 9], BM3D [13, 14, 18], NL-Bayes [19] and K-SVD [1, 20].

In this paper we focus on two methods using to image denoising: anisotropic diffusion and non local means. We propose a method which combines these two approaches. Reconstructed pixel is expressed in terms of stochastic anisotropic diffusion which is itself driven by similarity of patches of the noisy image.

2 Mathematical Preliminaries

Let $D \subset \mathbf{R}^n$ be a domain with closure $\overline{D}$ and boundary ∂D. Let $T > 0$ and by $\mathbf{C}([0,T];\mathbf{R}^n)$ we denote a set of continuous functions $f : [0,T] \to \mathbf{R}^n$.

Definition 1. *Let $y \in \mathbf{C}([0,T];\mathbf{R}^n)$, $y_0 \in \overline{D}$. A pair $(x,k) \in \mathbf{C}([0,T];\mathbf{R}^{2n})$ is said to be a solution to the Skorokhod problem associated with y and D if*

1. $x_t = y_t + k_t, \quad t \in [0,T]$,
2. $x_t \in \overline{D}, \quad t \in [0,T]$,
3. *k is a function with bounded variation $|k|$ on $[0,T]$, $k_0 = 0$ and*

$$k_t = \int_0^t n_s \, d|k|_s, \; |k|_t = \int_0^t 1_{\{x_s \in \partial D\}} \, d|k|_s, \; t \in [0,T],$$

where $n_s = n(x_s)$ is an inward normal unit vector at $x_s \in \partial D$.

It is known that if D is a convex set, then there exists a unique solution to the Skorokhod problem [25].

Definition 2. *Let $(\Omega, \mathcal{F}, \mathcal{P})$ be a probability space.*

1. *An n-dimensional stochastic process $X = \{X_t; t \in [0,T]\}$ is a parametrised collection of random variables defined on a probability space $(\Omega, \mathcal{F}, \mathcal{P})$ with values in $\mathbf{R}^n$.*
   For each fixed $\omega \in \Omega$ the function $X_t(\omega)$, $t \in [0,T]$ is called a trajectory of X and is denoted by $X(\omega)$.
2. *A filtration $(\mathcal{F}_t) = \{\mathcal{F}_t; t \in [0,T]\}$ is a nondecreasing family of sub-σ-fields of $\mathcal{F}$, i.e. $\mathcal{F}_s \subseteq \mathcal{F}_t \subseteq \mathcal{F}$ for $0 \le s < t \le T$.*
   By $(\mathcal{F}_t^X)$ we denote a filtration generated by a process X, i.e.
   $\mathcal{F}_t^X = \sigma(X_s; 0 \le s \le t)$.
3. *A stochastic process X is adapted to the filtration $(\mathcal{F}_t)$ (X is $(\mathcal{F}_t)$ adapted) if for each $t \in [0,T]$, X_t is $\mathcal{F}_t$ - measurable random variable.*

Definition 3. *Let Y be $(\mathcal{F}_t)$ adapted process with continuous trajectories, $Y_0 \in \overline{D}$. We say that a pair (X,K) of $(\mathcal{F}_t)$ adapted processes is a solution to the Skorokhod problem associated with Y and D, if for almost every $\omega \in \Omega$, $(X(\omega), K(\omega))$ is a solution to the Skorokhod problem associated with $Y(\omega)$ and D.*

In what follows, by $W = \{W_t; t \in [0,T]\}$ we shall denote a Wiener process starting from zero. We assume that we are given a point $x_0 \in \overline{D}$ and some function $\sigma : \mathbf{R}^n \to \mathbf{R}^n \times \mathbf{R}^m$.

Definition 4. *Let Y be an $(\mathcal{F}_t)$ adapted process. A pair $(X, K^{\overline{D}})$ of $(\mathcal{F}_t)$ adapted processes is called a solution to reflected stochastic differential equation (in short reflected SDE)*

$$X_t = x_0 + \int_0^t \sigma(X_s)\, dW_s + K_t^{\overline{D}},\ \ t \in [0,T], \tag{1}$$

if $(X, K^{\overline{D}})$ is a solution to the Skorokhod problem associated with

$$Y_t = x_0 + \int_0^t \sigma(X_s)\, dW_s,\ \ t \in [0,T] \ \text{ and } \ D.$$

The process X is called the process with reflection. The proof of existence and uniqueness of the solution to reflected SDEs can be found in [25].

3 Stochastic Anisotropic Diffusion

Following [21, 30] we propose the following stochastic model of an anisotropic diffusion:

$$X_t = x + \int_0^t \begin{bmatrix} -\frac{(G_\gamma * u_0)_{x_2}(X_s)}{|\nabla(G_\gamma * u_0)(X_s)|}, 0 \\ \\ \frac{(G_\gamma * u_0)_{x_1}(X_s)}{|\nabla(G_\gamma * u_0)(X_s)|}, 0 \end{bmatrix} dW_s + K_t^{\overline{D}},$$

where $u_{x_i}(y) = \frac{\partial u}{\partial x_i}(y)$. To avoid false detections due to noise, u_0 is convolved with a Gaussian kernel $G_\gamma(x) = \frac{1}{2\pi\gamma^2} e^{-\frac{|x|^2}{2\gamma^2}}$ (in practice a 3 × 3 Gaussian mask).

The reconstruction pixel is given by

$$u(x) = \mathbf{E}\left[u_0(X_T)\right] \approx \frac{1}{M} \sum_{i=1}^{M} u_0(X_T^m(\omega_i)), \tag{2}$$

where $X^m(\omega_i)$ is the approximation of trajectory of stochastic process X and M is the number of Monte Carlo method iterations.

3.1 Euler's Approximation

Consider the following numerical scheme

$$\begin{aligned} X_0^m &= X_0, \\ X_{t_k}^m &= \Pi_{\overline{D}}[X_{t_{k-1}}^m + \sigma(X_{t_{k-1}}^m)(W_{t_k} - W_{t_{k-1}})], k = 1, ..., m, \end{aligned} \tag{3}$$

where $t_k = kh$, $h = \frac{T}{m}$, $k = 0, 1, ..., m$ and $\Pi_{\overline{D}}(x)$ denotes a projection of x on the set $\overline{D}$. Since D is convex, the projection is unique.

Theorem 1. *Let* $(X, K^{\overline{D}})$ *be the solution to the reflected SDE (1). If there exists* $C > 0$ *such that*

$$\|\sigma(x) - \sigma(y)\|^2 \leq C|x - y|^2,$$

then

$$\lim_{m \to +\infty} |X_T^m - X_T| = 0 \quad \text{almost surely.}$$

The proof of the above theorem can be found in [24].

3.2 Modified Diffusion

The numerical scheme (3) gives good results, but only with a small value of the time-step parameter $h = \frac{T}{m}$ (for example $h = 0.05$). Calculating the mean value using Monte Carlo method for small h is not effective and takes a long time. To omit this problem, we improve the scheme (3) by adding a controlled parameter p [4].

$$\begin{aligned} X_0^m &= X_0, \\ H_{t_k}^m &= \Pi_{\overline{D}}[X_{t_{k-1}}^m + \sigma(X_{t_{k-1}}^m)(W_{t_k} - W_{t_{k-1}})], \\ X_{t_k}^m &= \begin{cases} H_{t_k}^m, & \text{if } \Theta, \\ \\ X_{t_{k-1}}^m, & \text{elsewhere,} \end{cases} \qquad k = 1, 2, ..., m, \end{aligned} \tag{4}$$

where by Θ we mean the condition

$$|(G_\gamma * u_0)(H_{t_k}^m) - (G_\gamma * u_0)(X_{t_{k-1}}^m)| < p.$$

Note that the parameter $p > 0$ guarantees that if the image exhibits a strong gradient then the process X^m diffuses as a process with small value of the parameter h and at locations where variations of the brightness are small, the process X^m can diffuse with a large value of h (for example $h = 4$).

The figure Fig. 1. illustrates a difference between the scheme (3) and the scheme (4). There are shown three examples of trajectories of the process $u_0(X_t^m)$ from the pixel A to the pixel B. Trajectories (I) and (III) were generated using the scheme (3) for large and small value of the parameter h, respectively. Trajectory (II) was generated using the scheme (4) for large h. It is easy to see, that at locations where the image is constant, trajectory (II) diffuses as trajectory (I). At locations where the image has strong gradient, the trajectory (II) is similar to the trajectory (III).

For small h or $p = +\infty$ (in practice $p > 255$) the numerical scheme (4) is equivalent to the scheme (3).

3.3 Modified Diffusion with Random Terminal Time

At locations where gradient is large in all directions it is possible that condition Θ does not hold as many times as we would expect. To avoid this we propose the following modification [6]:

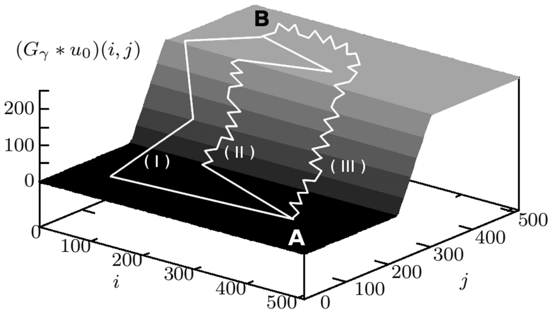


Fig. 1. Example of trajectories of the process $(G_\gamma * u_0)(X_t^m)$ from pixel A to B: (I) – with using the scheme (3) and large h, (II) – with using the scheme (4) and large h, (III) – with using the scheme (3) and small h

$$X_0^m = X_0,$$
$$H_{t_k}^m = \Pi_{\overline{D}}[X_{t_{k-1}}^m + \sigma(X_{t_{k-1}}^m)(W_{t_k} - W_{t_{k-1}})],$$

$$X_{t_k}^m = \begin{cases} H_{t_k}^m, & \text{if } \Theta, \\ \\ X_{t_{k-1}}^m, & \text{elsewhere}, \end{cases} \qquad k = 1, 2, ..., \tau_m,$$

where $\tau_m = \min\{k; k \geq m \,\text{and}\, \Theta \text{ is true } m \text{ times}\}$.

Terminal time τ_m guarantees that the numerical simulation of the diffusion trajectory gives at least m values of $X_{t_k}^m$ which differ from the value in the previous step.

3.4 Implementation

The above schemes are simple to implement. Observe that the schemes works well only if the model of the digital image $G_\gamma * u_0$ is continuous. In practice, we can use a linear interpolation to get the value of the image $G_\gamma * u_0$, for any point $x \in \overline{D}$. We note also that since $W_{t_k} - W_{t_{k-1}} \sim \mathcal{N}(0, t_k - t_{k-1})$, it can be approximated by a random number generator of the normal distribution.

4 Non Local Means Algorithm

In this section we cite results from [8–10].

Let $v = \{v(i) | i \in I\}$ be a discrete noisy image and $\{w(i,j)\}$ be the weights that depend on the similarity between the pixels i and j and satisfy the usual conditions $0 \leq w(i,j) \leq 1$ and $\sum_j w(i,j) = 1$. The reconstructed value $NL(v)(i)$ for a pixel i is defined as a weighted average of all pixels in the image

$$NL(v)(i) = \sum_{j \in I} w(i,j)v(j).$$

The weight $w(i,j)$ depends on the similarity of the intensity gray level vectors of neighbourhoods centred at pixels i and j.

Definition 5. *A neighbourhood system on I is a family $\mathcal{N} = \{\mathcal{N}_i\}_{i \in I}$ of subsets of I such that for all $i \in I$,*

1. *$i \in \mathcal{N}_i$,*
2. *$j \in \mathcal{N}_i \Rightarrow i \in \mathcal{N}_j$.*

The subset $\mathcal{N}_i$ is called the neighbourhood or the similarity window of i.

The weights can be defined by

$$w(i,j) = \frac{1}{Z(i)} e^{-\frac{d(\mathcal{N}_i, \mathcal{N}_j)}{s^2}}$$

where $Z(i)$ is the normalising factor $Z(i) = \sum_j e^{-\frac{d(\mathcal{N}_i, \mathcal{N}_j)}{s^2}}$ and $d(\mathcal{N}_i, \mathcal{N}_j)$ is some measure of distance between intensity gray level vectors of similarity windows. The number s is a parameter that controls the decay of the exponential function.

In original approach [8] the authors propose to use square windows of fixed size as similarity windows (see Fig. 2) and the distance between neighbourhoods was measured as a decreasing function of the weighted Euclidean distance i.e. $d(\mathcal{N}_i, \mathcal{N}_j) = \|v(\mathcal{N}_i) - v(\mathcal{N}_j)\|_{2,a}^2$, where $v(\mathcal{N}_i) = (v(j), j \in \mathcal{N}_i)$ and $a > 0$ is the standard deviation of the Gaussian kernel.

4.1 Patchwise Implementation

By $\|B_{i,r} - B_{j,r}\|_2$ we denote the Euclidean distance between $B_{i,r}$ and $B_{j,r}$, where patch $B_{i,r}$ means a neighbourhood of a size $2r+1 \times 2r+1$ pixels centred at i. Patchwise implementation [10] is based on a simple observation. When computing the Euclidean distance $\|B_{i,r} - B_{j,r}\|_2$, all pixels in the patch $B_{i,r}$ have the same importance, and therefore the weight $F(\|B_{i,r} - B_{j,r}\|_2)$, where F is a decreasing function, can be used to denoise all pixels in the patch $B_{i,r}$ and not only i. For computational purposes the searching of similar windows can be restricted from all pixels in the image to some square window $B_{i,f}$. The denoising of an image v and a certain patch $B_{i,r}$ is equal to

$$\hat{B}_{i,r} = \frac{1}{Z} \sum_{j \in B_{i,f}} v(B_{j,r}) w(B_{i,r}, B_{j,r}),$$

where $Z = \sum_{j \in B_{i,f}} w(B_{i,r}, B_{j,r})$ and the weight function is given by

$$w(B,Q) = e^{-\frac{\max\left(\|B-Q\|_2^2 - 2\rho^2, 0.0\right)}{s^2}}.$$

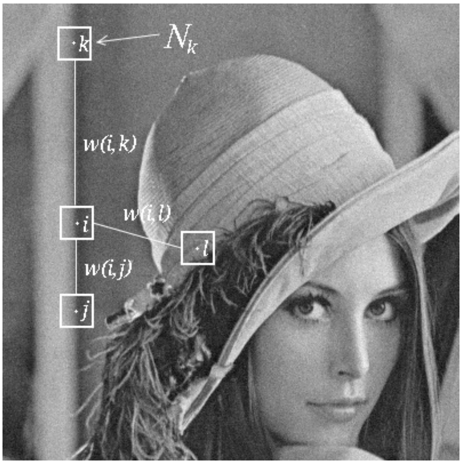


Fig. 2. Idea of NL-means from [8]: Pixels j and k have large weights $w(i,j)$ and $w(i,k)$ because their similarity windows are similar to that of i. The weight $w(i,l)$ is much smaller because the intensity grey values in the similarity windows are very different.

Here by ρ we denoted the standard deviation of the noise and by s the filtering parameter set depending on the value of ρ. The weight function is chosen in order to average similar patches up to noise. That is, patches with square distances smaller than $2\rho^2$ are set to 1, while larger distances decrease rapidly accordingly to the exponential kernel.

By applying the procedure for all patches in the image, we will get $(2r+1)^2$ possible estimates for each pixel. These estimates can be finally averaged at each pixel location in order to build the final denoised image

$$NL(v)(i) = \frac{1}{(2r+1)^2} \sum_{j \in B_{i,r}} \hat{B}_{j,r}(i).$$

5 Anisotropic Stochastic Diffusion Collaborated with Non Local Means

In this section we propose a new method of the image reconstruction based on modified diffusion with random terminal time and patchwise implementation of non local means.

In the case of numerical scheme with random terminal time the reconstructed formula of anisotropic diffusion (2) can be written as

$$u(x) \approx \sum_{i=1}^{M} \frac{1}{M} u_0(X^m_{\tau_m}(\omega_i)),$$

which means that each pixel $u_0(X^m_{\tau_m}(\omega_i))$ is weighted with the value $\frac{1}{M}$. But since pixels have different intensities we may consider them with different weights depending on their neighbourhood. We follow NL-means algorithm and propose a new method of the image restoration based on modified diffusion but such that the weights depend on patches similarity:

$$u(x) = \frac{1}{Z}\sum_{i=1}^{M} u_0(X^m_{\tau_m}(\omega_i))w(B_{x,r}, B_{X^m_{\tau_m}(\omega_i),r}).$$

In the above formula we used the following notations:

$$Z = \sum_{i=1}^{M} w(B_{x,r}, B_{X^m_{\tau_m}(\omega_i),r}),\ w(B,Q) = e^{-\frac{\max\left(\|B-Q\|_2^2-2\rho^2,0.0\right)}{s^2}}$$

$$X^m_0(\omega_i) = x,$$

$$H^m_{t_k}(\omega_i) = \Pi_{\overline{D}}[X^m_{t_{k-1}}(\omega_i) + \sigma(X^m_{t_{k-1}}(\omega_i))(W_{t_k} - W_{t_{k-1}})],$$

$$X^m_{t_k}(\omega_i) = \begin{cases} H^m_{t_k}(\omega_i), & \text{if } \Theta, \\ \\ X^m_{t_{k-1}}(\omega_i), & \text{elsewhere,} \end{cases} \qquad k = 1, 2, ..., \tau_m.$$

The meaning of the parameters in the new method is the same as in original approaches. This method of the image reconstruction we will call the stochastic diffusion with non local means (in short SDNLM).

6 Experimental Results

Some measures of quality for our evaluation experiments regarding new method, non local means algorithm, anisotropic Perona-Malik model [21] and anisotropic stochastic diffusion are presented in Table 1, Table 2, Fig. 3 and Fig. 4. The results refer to greyscale images *pirate* and *cameraman* corrupted with the Gaussian noise with standard deviation ρ. The maximum values of Peak Signal to Noise Ratio (in short PSNR) and Structural Similarity Index (in short SSIM) obtained using tested methods are given in tables. Parameters of SSIM were set to the default values as recommended by [28].

The analysis of the measures of image quality shows that in most cases the new method performs better. Moreover, when comparing the figures one can observe that the image created by the SDNLM is visually more pleasant. The reason for this is that the NL-means approach shows clear evidence of a halo of noise effect around the edges whereas anisotropic diffusions smooth details too much.

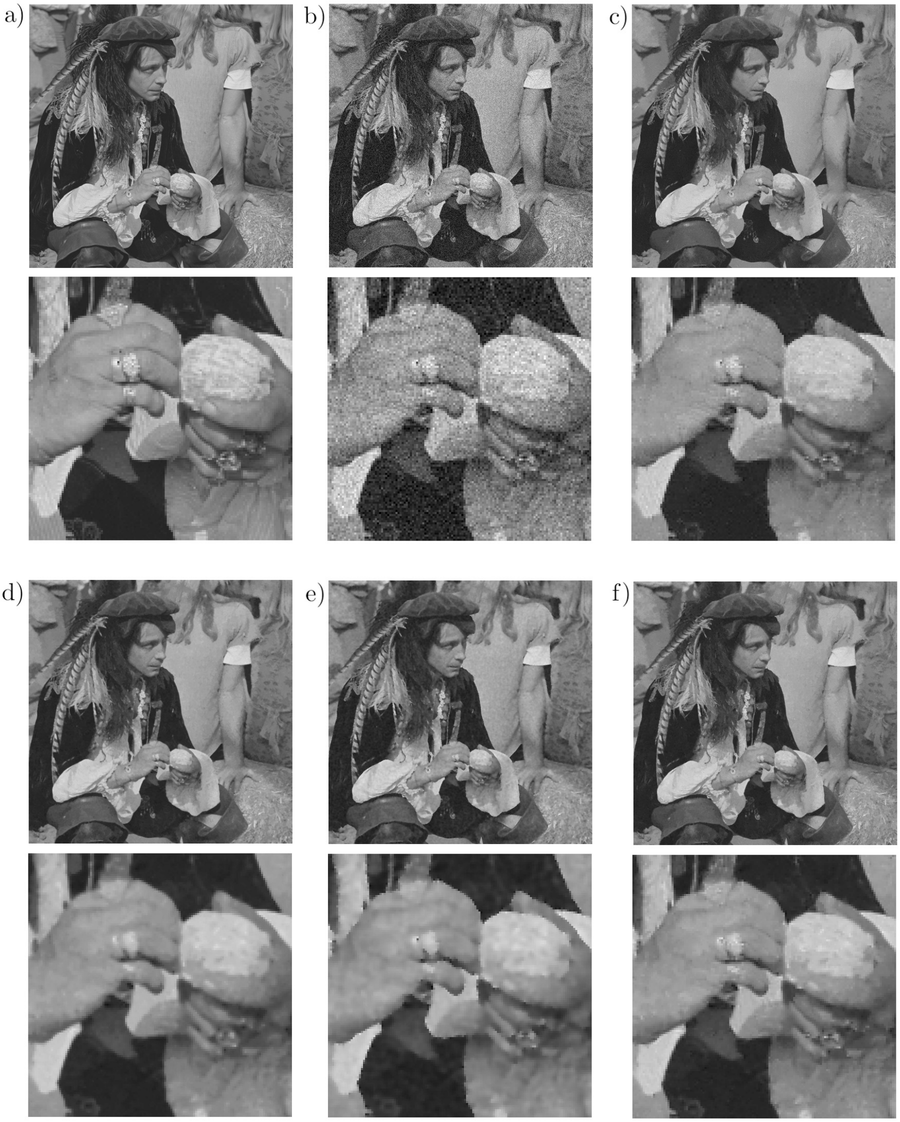


Fig. 3. a) Original image: 512 × 512 (top) and 128 × 128 (bottom) b) Noisy image: $\rho = 15$ c) NL-means: SSIM = 0.9245 d) Stochastic anisotropic diffusion: SSIM = 0.9246 e) Perona-Malik: SSIM = 0.9204 f) New method: SSIM = 0.9257

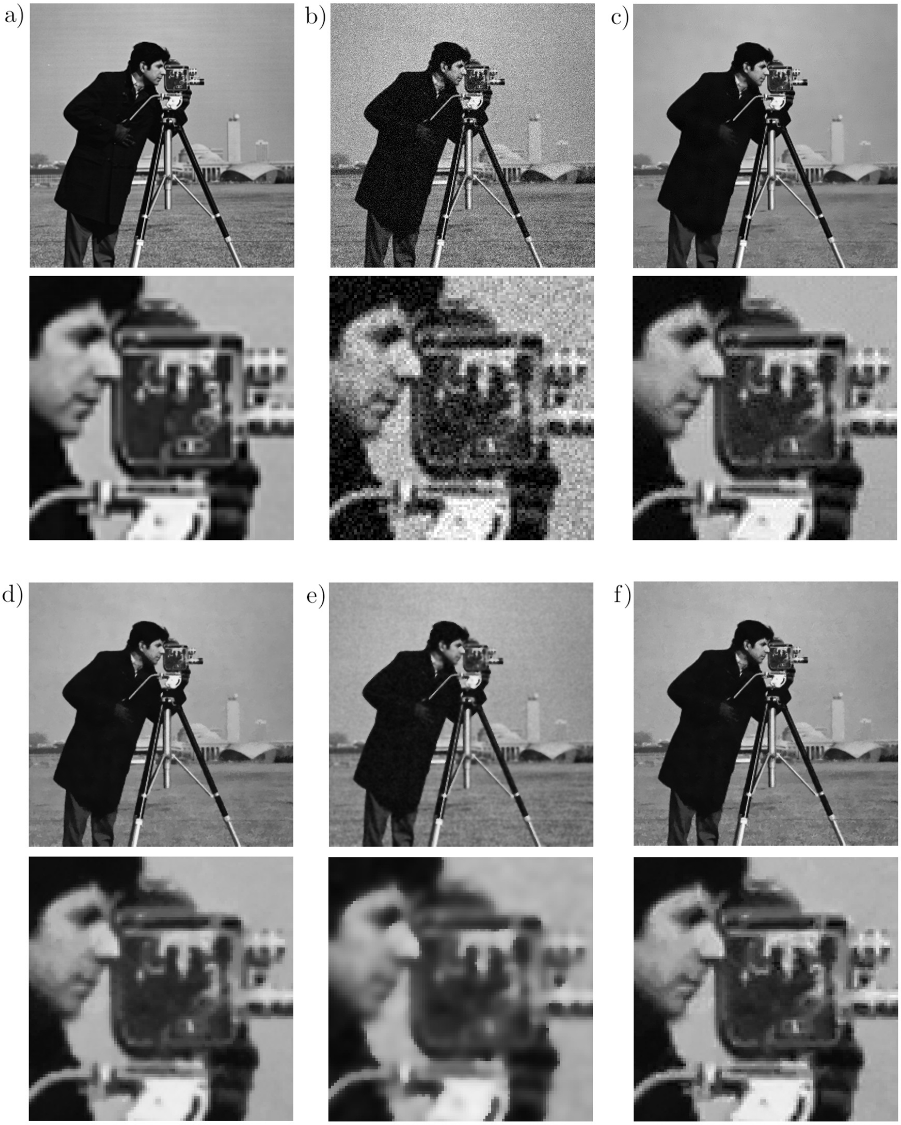


Fig. 4. a) Original image: 512×512 (top) and 96×96 (bottom) b) Noisy image: $\rho = 20$ c) NL-means: SSIM = 0.9214 d) Stochastic anisotropic diffusion: SSIM = 0.9144 e) Perona-Malik: SSIM = 0.8838 f) New method: SSIM = 0.9196

Table 1. Maximum values of PSNR

Image	**Noise** ρ	**NL-means algorithm [10]**	**Stoch. anisotropic diffusion [6]**	**Perona-Malik [21]**	**New method**
Pirate	10	33.0394	32.0450	32.8268	**33.1379**
	15	30.7474	30.5158	30.8167	**30.8944**
	20	29.6731	29.3619	29.5095	**29.6914**
Cameraman	10	35.3814	35.2304	34.9002	**35.8238**
	15	32.9025	33.7815	32.8394	**33.8727**
	20	32.1931	**32.2683**	31.2357	32.1571

Table 2. Maximum values of SSIM

Image	**Noise** ρ	**NL-means algorithm [10]**	**Stoch. anisotropic diffusion [6]**	**Perona-Malik [21]**	**New method**
Pirate	10	**0.9562**	0.9533	0.9519	0.9558
	15	0.9245	0.9246	0.9204	**0.9257**
	20	0.8952	0.8935	0.8913	**0.8974**
Cameraman	10	0.9600	0.9583	0.9436	**0.9606**
	15	0.9334	0.9369	0.9173	**0.9393**
	20	**0.9214**	0.9144	0.8838	0.9196

7 Conclusion

In this paper we proposed a new method of digital image denoising. Expressing anisotropic diffusion in terms of stochastic equations allows us to adapt the idea from non local means approach. The new method takes what is the best both from anisotropic diffusion and non local means method: reconstructed image is smooth and at the same time details are preserved.

As a future work, the algorithm can be extended to vector valued images, in particular, colour images.

References

1. Aharon, M., Elad, M., Bruckstein, A.: K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. IEEE Trans. Image Process. 54(11), 4311–4322 (2006)
2. Aubert, G., Kornprobst, P.: Mathematical problems in image processing. Springer, New York (2002)
3. Borkowski, D.: Chromaticity Denoising using Solution to the Skorokhod Problem. In: Image Processing Based on Partial Differential Equations. Mathematics and Visualization, Part II, pp. 149–161 (2007)
4. Borkowski, D.: Modified diffusion to Image Denoising. Adv. Soft. Comp. 45, 92–99 (2007)
5. Borkowski, D.: Smoothing, Enhancing Filters in Terms of Backward Stochastic Differential Equations. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) ICCVG 2010, Part I. LNCS, vol. 6374, pp. 233–240. Springer, Heidelberg (2010)

6. Borkowski, D.: Euler's Approximations to Image Reconstruction. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) ICCVG 2012. LNCS, vol. 7594, pp. 30–37. Springer, Heidelberg (2012)
7. Borkowski, D., Jańczak-Borkowska, K.: Application of Backward Stochastic Differential Equations to Reconstruction of Vector-Valued Images. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) ICCVG 2012. LNCS, vol. 7594, pp. 38–47. Springer, Heidelberg (2012)
8. Buades, A., Coll, B., Morel, J.M.: A non local algorithm for image denoising. IEEE Computer Vision and Pattern Recognition 2, 60–65 (2005)
9. Buades, A., Coll, B., Morel, J.M.: A review of image denoising algorithms, with a new one. Multiscale Model. Simul. 4(2), 490–530 (2006)
10. Buades, A., Coll, B., Morel, J.M.: Non-Local Means Denoising. Image Processing On Line (2011)
11. Catte, F., Lions, P.L., Morel, J.M., Coll, T.: Image selective smoothing and edge detection by nonlinear diffusion. SIAM J. Numer. Anal. 29(1), 182–193 (1992)
12. Chan, T.F., Shen, J.J.: Image Processing and Analysis – Variational, PDE, wavelet, and stochastic methods. SIAM, Philadelphia (2005)
13. Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising by sparse 3D transform-domain collaborative filtering. IEEE Trans. Image Process. 16(8), 2080–2095 (2007)
14. Danielyan, A., Katkovnik, V., Egiazarian, K.: Bm3d frames and variational image deblurring. IEEE Trans. Image Process. 21(4), 1715–1728 (2012)
15. Donoho, D.L., Johnstone, I.M.: Ideal spatial adaptation via wavelet shrinkage. Biometrika 81(3), 425–455 (1994)
16. Geman, S., Geman, D.: Stochastic relaxation, gibbs distributions and the bayesian restoration of images. IEEE Pat. Anal. Mach. Intell. 6, 721–741 (1984)
17. Juan, O., Keriven, R., Postelnicu, G.: Stochastic Motion and the Level Set Method in Computer Vision: Stochastic Active Contours. Int. J. Comput. Vision 69(1), 7–25 (2006)
18. Katkovnik, V., Danielyan, A., Egiazarian, K.: Decoupled inverse and denoising for image deblurring: variational BM3D-frame technique. In: Proceedings of IEEE International Conference on Image Processing (2011)
19. Lebrun, M., Buades, A., Morel, J.M.: Implementation of the Non-local Bayes image denoising. Image Processing On Line (2011)
20. Mairal, J., Elad, M., Sapiro, G.: Sparse representation for color image restoration. IEEE Trans. Image Process. 17(1), 53–69 (2008)
21. Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE Trans. Pattern Anal. Mach. Intell. 12(7), 629–639 (1990)
22. Richardson, W.H.: Bayesian-based iterative method of image restoration. JOSA 62(1), 55–59 (1972)
23. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Phys. D 60, 259–268 (1992)
24. Słomiński, L.: Euler's approximations of solutions of SDEs with reflecting boundary. Stoch. Proc. Appl. 94, 317–337 (2001)
25. Tanaka, H.: Stochastic differential equations with reflecting boundary condition in convex regions. Hiroshima Math. J. 9(1), 163–177 (1979)
26. Unal, G., Krim, H., Yezzi, A.: Stochastic differential equations and geometric flows. IEEE Trans. Image Process. 11(12), 1405–1416 (2002)
27. Unal, G., Ben-Arous, G., Nain, D., Shimkin, N., Tannenbaum, A., Zeitouni, O.: Algorithms for stochastic approximations of curvature flows. In: Image Processing, Proceedings ICIP 2003, vol. 2-3, pp. 651–654 (2003)

28. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. IEEE Trans. Image Process. 13(4), 600–612 (2004)
29. Weickert, J.: Theoretical Foundations of Anisotropic Diffusion in Image Processing. Computing Suppement 11, 221–236 (1996)
30. Weickert, J.: Coherence-Enhancing Diffusion Filtering. Int. J. Comput. Vision 31(2/3), 111–127 (1999)
31. Yaroslavsky, L.P.: Local adaptive image restoration and enhancement with the use of DFT and DCT in a running window. In: Proceedings of SPIE, vol. 2825, pp. 2–13 (1996)

A Practical Certificate and Identity Based Encryption Scheme and Related Security Architecture

Tomasz Hyla and Jerzy Pejaś

West Pomeranian University of Technology in Szczecin
Faculty of Computer Science and Information Technology, Poland
{thyla,jpejas}@wi.zut.edu.pl

Abstract. Group encryption schemes based on general access structures can be used to build advanced IT systems, which store and manage confidential documents. The paper proposes a reference architecture of public key cryptography infrastructure required to implement CIBE-GAS scheme. The CIBE-GAS scheme is a certificate-based group-oriented encryption scheme with an effective secret sharing scheme based on general access structure and bilinear pairings. The security architecture required to implement the scheme must be compliant with common standards and technical specifications, e.g. X.509 certificate format and XML-encryption standard for messages. In order to encrypt arbitrary-length messages, we also suggest a new CIBE-GAS-H scheme with a key encapsulation mechanism based on the techniques of Bentahar *et al.*, and combined with one-time symmetric-key encryption.

Keywords: group encryption, general access structures, security architecture, pairing based cryptosystem.

1 Introduction

The certificate and ID-based group encryption scheme with bilinear pairings allows to design the cryptographic access control mechanisms for protection of sensitive information. Due to these mechanisms, the information can be stored in the network in an encrypted form and decrypted only by authorised users [1, 2]. However, the development of a system that uses such mechanisms requires consideration of many additional architectural problems. These problems include trust and certification models selection and choice of proper access structures.

Public keys certification models correspond to different methods of public keys' management, including theirs generation, certification, distribution and revocation. The architecture consists of a set of well-defined components, their functions and relations between them, including the trust relationship. The primary purpose of trust models is creation of trust relationship between any entities within the same or between different key management architectures (KMA). The trust models are based on the certification models with distinguished local trust authorities (TA).

An access structure is a rule that defines how to share a secret, or more widely, who has an access to particular assets in IT system. Access structures can be classified

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 190–205, 2013.

as structures with and without threshold. Although threshold access structures are frequently used, the non-threshold structures (called also as general access control) are more versatile.

ID-based cryptosystems (IBC) had received considerable interest to cryptographic researchers since A. Shamir's work [3]. However, the question was how to construct effectively such systems. After slightly more than a decade ago, in 2001, Boneh and Franklin [4] proposed the first practical cryptographic IBE scheme based on bilinear pairings. Since then many extensive researches have been done, but only limited results of them have been implemented into commercial products. This is mainly due to the low commercial maturity of ID-based cryptography schemes measured by the number of available products and standards.

However, the current state of the commercialising IBC schemes and developing standards is slowly changing. We know three practical implementations of identity based cryptographic techniques:

- a commercial product for encrypted e-mail based on Boneh-Franklin IBE scheme (Voltage Security Inc, http://www.voltage.com);
- an application for secure e-mail encryption based on the Sakai-Kasahara ID- based key encapsulation mechanism (Trend-Micro, www.trendmicro.com/us);
- a smart-card implementation of IBE based on the Boneh-Franklin scheme (Gemalto, http://www.gemalto.com/press/gemplus/2004/id_security/02-11-2004-Identity-Based_Encryption.htm).

Among standards, there are:

- a draft standard of IEEE P1363.3/D1 for *Identity-based Public-key Cryptography Using Pairings* [5];
- RFC 6508 *Sakai-Kasahara Key Encryption (SAKKE)* [6];
- RFC 5091 *Identity-Based Cryptography Standard (IBCS) #1: Supersingular Curve Implementations of the BF and BB1 Cryptosystems* [7];
- RFC *5408 Identity-Based Encryption Architecture and Supporting Data Structures* [8].

1.1 Our Contributions

In this paper we introduce a new practical CIBE-GAS-H scheme (the modification of our previous CIBE-GAS scheme [1]). The new CIBE-GAS-H scheme is designed to work with arbitrary length messages while original CIBE-GAS scheme works with limited length messages only. We also propose reference security architecture to implement CIBE-GAS-H scheme, together with analysis of security issues related to implementation of the scheme.

We propose a hybrid system, which merges traditional PKI solutions with our CIBE-GAS-H scheme in order to achieve the good scalability, comparable with traditional X.509 based architectures. Combining these two cryptosystems in a single framework has advantages of the traditional PKI and ID-based public key cryptography and, for example, allows to authenticate both users of PKI domain and ID-based domain. The framework defines data structures that can be used to implement the proposed hybrid trust system. These structures are required to support on-line interactions between CIBE-GAS users and PKI/TA management entities. We define messaging system, describing how the components work together, and data structures that support the system operation (with the extension of the standard X.509 certificate).

1.2 Paper Organisation

The paper is organized as follows. Section 2 contains description of basic bilinear pairings and short description of our CIBE-GAS scheme introduced in [1]. Section 3 introduces CIBE-GAS-H scheme, which is an extension of CIBE-GAS scheme for arbitrary length messages. Section 4 contains description of architecture for CIBE-GAS-H scheme and related security issues. Moreover, the section provides description of current standards that can be used in CIBE-GAS-H system architecture. The paper ends with summary and conclusions.

2 Background

2.1 Bilinear Groups and Security Assumptions

A pairing $\hat{e}$ is defined as a bilinear map between elements of two finite, cyclic and additive groups G_1 and G_2 to a third finite cyclic group G_T defined multiplicatively. Both of G_1 and G_2 are of prime order q, as it is in the case of G_T. In practice, pairing $\hat{e}$ allows to solve certain problem in one group, even if the problem is said to be hard in another group.

Depending upon the structure of the group G_2, a bilinear pairing can be classified as one of the following three types [9, 10]:

- **Type 1:** $G_2 = G_1$.
- **Type 2:** $G_2 \neq G_1$, but there is an efficiently computable isomorphism φ from G_2 to G_1.
- **Type 3:** $G_2 \neq G_1$, and there are no efficiently computable isomorphism φ from G_2 to G_1.

Note that since G1 and G2 are both cyclic groups of the same prime order, they are certainly isomorphic.

Here we simply consider symmetric pairings (i.e. the case of Type 1) in prime-order groups, using notations similar to those presented by Al-Riyami, S., et al. [11].

Definition 1. Let $(G_1, +)$ and $(G_T, \cdot)$ be two cyclic groups of some prime order $q>2^k$ for security parameter $k \in N$. The bilinear pairing is given as $\hat{e}: G_1 \times G_1 \to G_T$ and must satisfy the following three properties:

1. **Bilinearity:** $\hat{e}(aP, bQ) = \hat{e}(abP, Q) = \hat{e}(P, abQ) = \hat{e}(P, Q)^{ab}$ for all P, $Q \in G_1$ and all a, b Î Z_q^* ; this can be restated in the following way: for P, Q, $R \in G_1$, $\hat{e}(P+Q, R) = \hat{e}(P, R)\hat{e}(Q, R)$ and $\hat{e}(P, Q+R) = \hat{e}(P, Q)\hat{e}(P, R)$.
2. **Non-degeneracy:** some P, $Q \in G_1$ exists such that $\hat{e}(P, Q) \neq 1_{G_2}$; in other words, if P and Q are two primitive elements of G_1 , then $\hat{e}(P, Q)$ is a generator of G_2 .
3. **Computability:** given P, Q$\in G_1$, an efficient algorithm computing $\hat{e}(P, Q)$ exists.

Note that a pairing $\hat{e}$ is symmetric, since $\hat{e}(aP, bP) = \hat{e}(P, P)^{ab} = \hat{e}(bP, aP)$. The most commonly used pairings arise from the theory of elliptic curves, where G_1 is subgroup of points on an elliptic curve over a finite field, whereas G_T is a subgroup of the multiplicative group of a finite field.

2.2 One-Time Symmetric-Key Encryption

A one-time symmetric-key encryption (*SKE*) scheme consists of two deterministic polynomial time secret key (*SK*) algorithms, E_{SK} and D_{SK}, where key, message and ciphertext spaces are given by $K_{SK}(\kappa)$, $M_{SK}(\kappa)$, $C_{SK}(\kappa)$ for some security parameter $\kappa \in Z^+$. A deterministic encryption algorithm E_{SK} takes a message $M \in M_{SK}(\kappa) = (0, 1)^*$ and a key $K \in K_{SK}(\kappa)$ as inputs, and outputs a ciphertext $C = E_{SK}(K, M)$. Another deterministic algorithm D_{SK} is a decryption algorithm that takes a ciphertext C and a key K as inputs and outputs a message $M = D_{SK}(K, C)$ or $\perp$, when some error has occurred.

We assume that the scheme is sound, i.e. for all M we have $D_{SK}(K, E_{SK}(K, M)) = M$ and the key length $|M|$ is a polynomial function of the security parameter κ .

We do not define concrete one-time symmetric-key encryption scheme $SKE = (E_{SK}, D_{SK})$, but we assume that this scheme fulfils the security requirements given in [12, 13] and is secure against passive attacks (standard algorithms like AES, Blowfish or chaos-based ciphers, e.g. [14] can be used).

2.3 Full Certificate-Based Encryption Scheme with General Access Structure

In this Section first we review our Certificate-Based Encryption Scheme with General Access Structure (CIBE-GAS) [1]. This group encryption algorithm is intended to encrypt short plaintext messages M, which are a bit strings of length p. In Section 3 extension of CIBE-GAS scheme allowing to encrypt arbitrary length messages is presented.

Definition 2 (CIBE-GAS scheme). Assume that are given: *n*–element set containing all shareholders $U=\{u_1,u_2,...,u_n\}$, *m*–element access structure[1] $\Gamma=\{A_1,A_2,...,A_m\}$, dealer $D\notin U$ and combiner $Com\in U$. Eight probabilistic algorithms specify the original certificate-based encryption scheme with general access structure (CIBE-GAS).

- **Setup** $(1^K)\to(s, params)$
  Trusted Authority (TA) runs this algorithm. The algorithm takes a security parameter 1^K as an input and returns the master private key $s\in_R Z_q^*$, the master public key P_0 and the system parameter *params*:

  $$params=\{G_1,G_2,\hat{e},q,P,P_0,H_1,H_2,H_3,H_4,H_5,H_6\} \quad (1)$$

  where P – the primitive element of G_1, $P_0=sP$ - the public key, $H_1:\{0,1\}^*\times G_1\times G_1\to G_1^*$, $H_2:\{0,1\}^*\times G_1\times G_1\to Z_q^*$, $H_3:G_2\times\{0,1\}^*\to Z_q^*$, $H_4:\{0,1\}^p\times\{0,1\}^p\to Z_q^*$, $H_5:G_2\to\{0,1\}^p$ and $H_6:\{0,1\}^p\to\{0,1\}^p$ are secure hash functions. TA runs the algorithm and, after completion, keeps secret the master private key s, while P_0 and *params* are publicly accessible by all users in the system.
- **SetSecretValue** $(params)\to(s_E, Pk_E)$
  The entity *E*, i.e. any shareholder $u_i\notin U$, dealer $D\notin U$ and combiner $Com\in U$ runs this algorithm. The algorithm takes as an input the *params* and outputs the secret value s_E and the public key Pk_E for *E*.
- **CertGen** $(s, params, ID_E, Pk_E)\to Cert_E$
  This algorithm takes as an input the master private key s, the system parameter *params*, and an entity *E*'s identity ID_E and *E*'s public key Pk_E. It outputs the certificate $Cert_E$. The TA runs this algorithm once for each entity.
- **SetPublicKey** $(params, ID_E, Pk_E, Cert_E)\to\{yes, no\}$
  This algorithm takes as an input a system parameter *params*, an entity *E*'s identity ID_E, *E*'s public key Pk_E and $Cert_E$, and returns the positive result if the certificate is valid or the negative result in opposite case. It is run by the entity, and in positive case the resulting public key Pk_E is widely and freely distributed.

[1] The set $\Gamma=\{A_j \mid j=1,2,\ldots,m\}\subseteq 2^U$ is an *access structure*, if any secret x can be reconstructed by gathering all the shares x_i secretly owned by u_i in A_j. All sets in access structure Γ are called *authorised* or *qualified sets*.

- **ShareDistribution** (*params*, prm_D, prm_U, Γ) → *pubVal*
  This algorithm is run by a dealer *D*. It takes as an input a system parameter *params*, dealer parameters prm_D, shareholders parameters prm_U and current access structure Γ, and returns public values for each authorised subset $A_j \in \Gamma$. We assume that:

  $prm_U = (Cert_U , ID_U) = \{ \left(Cert_{u_i}, ID_{u_i}\right) | \; u_i \in U \}$

  $prm_D = \{ s_d , ID_d , Pk_d , Cert_d \}$

  $pubVal = (\beta, f(1), Y, Y_{-1}, (d_1,\ldots,d_m), (\gamma_1,\ldots,\gamma_m), (k_{1,1},\ldots,k_{1,m}; \ldots, k_{n,1},\ldots,k_{n,m}))$

- **Encryption** $(M, ID_d, Pk_d, F, pubVal, params; \sigma) \rightarrow C$

  On input of an limited length message $M \in \{0, 1\}^p$, the dealer *D*'s identity ID_d, his public key Pk_d, the filter *F*, which superimposed on the access structure Γ allows only privileged groups with indexes from F to decrypt information *M*, publicly known parameters *pubVal*, the system *params* and possibly some randomness $\sigma \in \{0, 1\}^p$, this algorithm outputs a ciphertext $C = (C_1, C_2, C_3, C_4, C_5, C_6)$. This algorithm is run when the dealer creates a new encrypted message. The ciphertext is calculated as follows:

$$r = H_4(\sigma, M), \; C_1 = r(H_2(ID_d, Pk_d)P + X_d) \tag{2}$$

$$C_2 = \sigma \oplus H_5\left(\hat{e}(P,Y)^r\right), \; C_3 = m \oplus H_6(\sigma), \; C_4 = \hat{e}(P, f(1)P)^r \tag{3}$$

$$C_5 = \left\{ v_k = \hat{e}(P, \gamma_k P)^r, \forall k \in F \subseteq 2^m \right\}, \; C_6 = rY_{-1} \tag{4}$$

- **SubDecryption** $(C, ID_{u_{i_j}}, Cert_{ID_{u_{i_j}}}, s_{i_j}, A_j, params, pubVal) \rightarrow \delta_{i_j,j}$

  Every shareholder (with an identity $ID_{u_{i_j}}$ and a certificate $Cert_{ID_{u_{i_j}}}$) from the privileged subset $u_{i_j} \in A_j \in \Gamma$, where $A_j = \{u_{1_j}, u_{2_j}, \ldots\}$, runs this algorithm and partially decrypts ciphertext *C* using his share s_{i_j}. The decrypted value $\delta_{i_j,j}$ is sent to a combiner.

- **Decryption** $(C, ID_d, Pk_d, A_j, (\delta_{1_j,j}, \ldots, \delta_{|Aj|_j,j}), params, pubVal) \rightarrow M'$

  This algorithm is run by a combiner $Com \in A_j$. On the input of a ciphertext *C*, a dealer's (ID_d, Pk_d), partially decrypted shares $\delta_{1_j,j}, \ldots, \delta_{|Aj|_j,j}$, system parameters *params* and public values for authorised subset $A_j \in \Gamma$, the algorithm outputs the corresponding value of the plaintext *M* or the failure symbol ⊥. The decryption algorithm does as follows:

$$\Delta = \Delta_1^{\frac{d_j}{d_{j-1}}} \cdot \Delta_2^{\frac{-1}{d_{j-1}}},\ \Delta_1 = C_4,\ \Delta_2 = v_j \prod_{u_{i_j} \in A_j} \delta_{i_j,j} \tag{5}$$

$$\sigma = C_2 \oplus H_5(\Delta),\ M = C_3 \oplus H_6(\sigma),\ r = H_4(\sigma, M) \tag{6}$$

If $C_1 \neq r(H_2(ID_d, Pk_d)P + X_d)$, then an algorithm raises an error condition and exits with ⊥, otherwise sets the plaintext to *M*.

This completes the high-level description of CIBE-GAS scheme. A workflow of all algorithms is depicted in Fig. 1. Each user (say, *E*) runs **SetSecretValue** algorithm that generates a private/public key pair by taking system parameters as an input. Then the user *E* sends the registration request (1) to the Registration Authority (RA) and asks the latter to issue the certificate (**CertGen** algorithm, see step (3)). RA examines the *E*'s information (ID_E, Pk_E) and initiates some process to verify the identifying information provided by the user. When the registration request has been approved, the RA sends the confirmation (2) to the Trusted Authority (TA). The TA checks confirmation, and if everything is correct it generates the certificate which binds together *E*'s (ID_E, Pk_E) and other information. At the end of this phase the certificate is sent to *E*. Before or after receiving the message (4) from TA (it is omitted here) the user *E* should provide the proof of his knowledge of the relevant private key s_E (so called proof of possession[2]). Next steps of CIBE-GAS scheme presented in Fig.1 are fully compliant with Definition 3 and are omitted here.

3 Extension for Arbitrary Length Messages

Here we extend our CIBE-GAS scheme to deal with arbitrary length messages. A s-imple and an efficient way to build an encryption scheme that has an unrestricted message length is to build a hybrid one. Loosely speaking, such a scheme is based on the well-known KEM-DEM framework [13, 15] using the key encapsulation mechanism (KEM) and data encapsulation mechanism (DEM). The KEM uses a public key encryption technique to derive and encrypt a shared key, while DEM uses the shared key in a symmetric key algorithm to encrypt the arbitrarily long message.

[2] There are numerous methods that can be used to show proof of possession. In the simplest one, the certificate issued by TA can be encrypted using *E*'s public key. Only the holder of the private key is able to decrypt the certificate to use it.

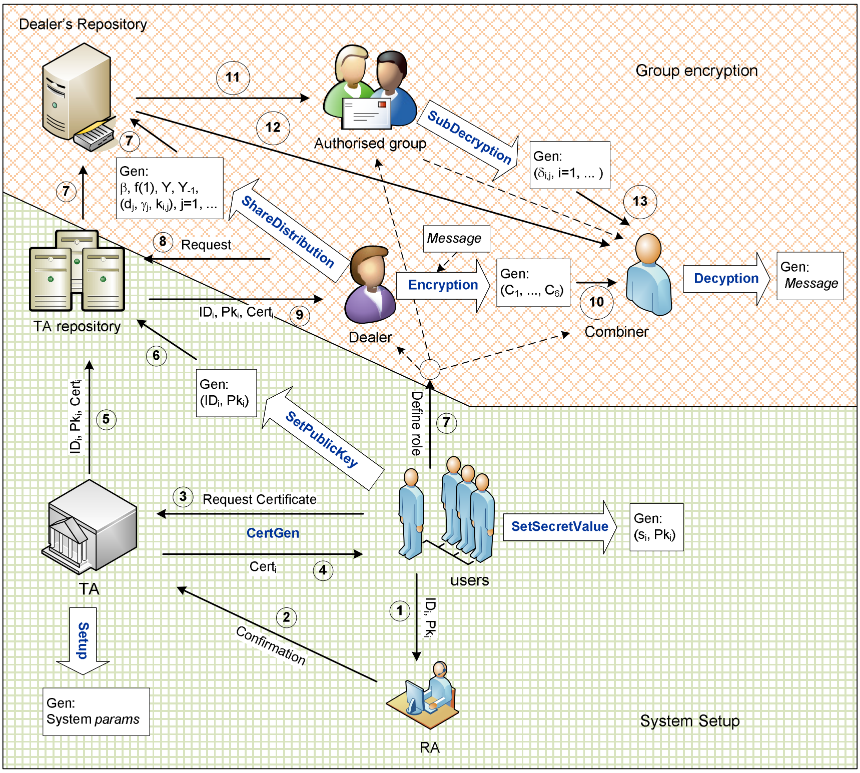


Fig. 1. Certificate and ID based group encryption scheme CIBE-GAS

Our extended encryption scheme is based on KEM-DEM framework given by L. Chen, *et al.* [12]. Following L. Chen, *et al.* formalisation of hybrid encryption, we assume that a hybrid construction *CIBE-GAS-H* = (*Setup-H*, *SetSecretValue-H*, *CertGen-H*, *SetPublicKey-H*, *ShareDistribution-H*, *Encryption-H*, *SubDecrytpion-H*, *Decryption-H*). For definitions of DEMs and their security definitions we refer to [12, 13, 15].

Definition 3 (*CIBE-GAS-H* scheme). For the same assumption as in Definition 2, the *CIBE-GAS-H* scheme consists of the following algorithms:

- **Setup-H** (1^{κ}) → (s , *params*)

  As in the previous CIBE-GAS scheme (see Definition 2). Additionally, the setup algorithm chooses a one-time symmetric key encryption scheme $SKE = (E_{SK}, D_{SK})$. The public parameters *params* are as follows:

$$params = \{G_1, G_2, \hat{e}, q, P, P_0, H_1, H_2, H_3, H_4, H_5, H_6, H_7, SKE\} \quad (7)$$

  Comparing to CIBE-GAS scheme, this construction uses the new cryptographic hash function $H_7 : \{0,1\}^p \rightarrow \{0,1\}^{\kappa}$ for some $\kappa \in Z$; the κ parameter means the length of the resulting symmetric key $\mathcal{K}$.

- The algorithms **SetSecretValue-H**, **CertGen-H**, **ShareDistribution-H** and **SubDecryption-H** are the same as **SetSecretValue**, **CertGen**, **ShareDistribution** and **SubDecryption** respectively, as in the previous non-Hybrid CIBE-GAS scheme (see Definition 2).
- **Encryption-H** *(M, ID_d , Pk_d , F, pubVal, params)* → (c_K, c_M)
  This operation is performed by the dealer. Given the dealer's identity ID_d , his public key Pk_d , the filter *F*, publicly known parameters *pubVal* and the system *params*, this algorithm outputs a ciphertext $C = (c_K, c_M)$, where c_K encapsulates the key *K* and c_M encapsulates the message *M*. We refer to such a construction as hybrid. The encryption algorithm with the key encapsulation mechanism (KEM) to calculate these values does as follows:
  (a) pick the random values $\sigma, m \in \{0, 1\}^p$;
  (b) calculate encrypted key material c_K = **Encryption** *(m,* ID_d , Pk_d , *F, pubVal, params;* σ);
  (c) calculate session key $K = H_7(m)$;
  (d) calculate $c_M = E_{SK}(K, M)$;
  (e) output (c_K, c_M)*;*
- **Decryption-H** $((c_K,c_M), ID_d , Pk_d , A_j ,(\delta_{1_j,j}, \ldots, \delta_{|A_j|_j,j}$), *params*, *pubVal*)→*K'*
  This algorithm is run by the combiner once per encrypted values (c_K, c_M). Given (c_K, c_M), the dealer's (ID_d, Pk_d), partially decrypted shares $\delta_{1_j,j}, \ldots, \delta_{|A_j|_j,j}$, system parameters *params* and public values for authorised subset $A_j \in \Gamma$, the algorithm outputs the corresponding message *M* or the failure symbol ⊥. The decryption algorithm works as follows:
  (a) calculate *m* = **Decryption** (c_K, ID_d , Pk_d , A_j , ($\delta_{1_j,j}, \ldots, \delta_{|A_j|_j,j}$), *params*, *pubVal*);
  (b) if (*m* == ⊥), then return ⊥; otherwise continue;
  (c) calculate session key $K = H_7(m)$;
  (d) decrypt message $M = D_{SK}(K, c_M)$;
  (e) return *M*.

4 CIBE-GAS-H System Architecture

In the following section we describe CIB-GAS-H architecture required to implement CIBE-GAS-H scheme (including original CIBE-GAS scheme). The component diagram shows basic system components and data flows between them. Next, we discuss security issues and describe messages exchanged between components.

4.1 Components

The architecture of CIBE-GAS system is based on SOA paradigm [16] and consists of five basic components. Fig. 2 presents a simplified component model (internal component structures and connection between interfaces are omitted to improve clarity of the figure). The most important component is CIBE-GAS library. This component consists of implemented algorithms from CIBE-GAS and CIBE-GAS-H schemes (ANSI C functions, based on PBC library [17]). This library component is directly used by other components (i.e. by Trusted Authority (TA), Dealer (DL) and Decrypter (DC) components) to perform cryptographic operations, which require to use CIBE-GAS-H scheme cryptographic operations.

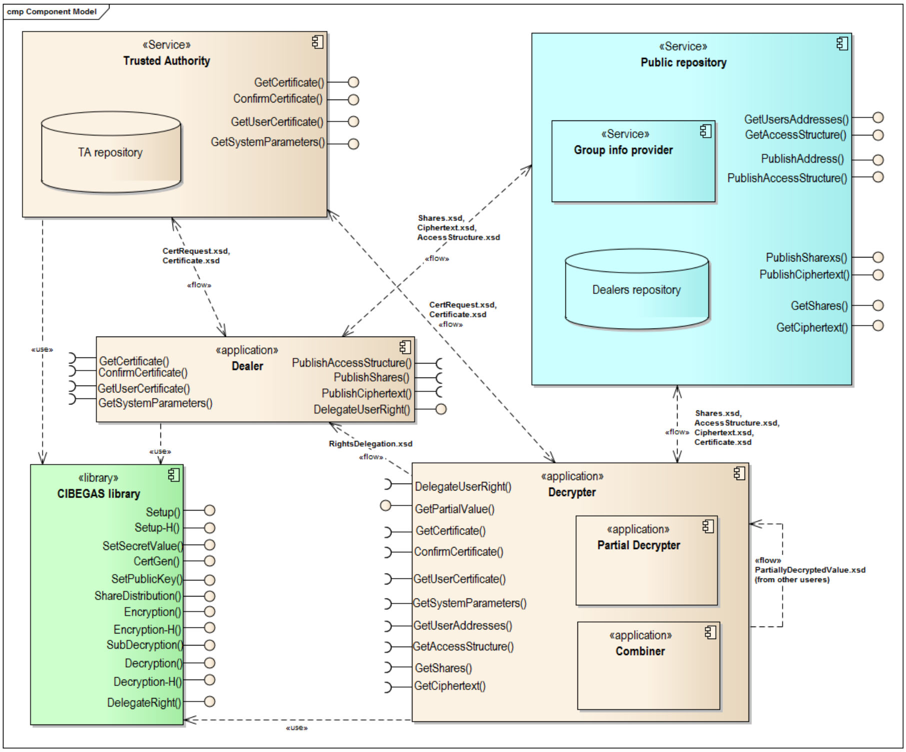


Fig. 2. Component model

The Trusted Authority (TA) component is responsible for management. It issues certificates and stores them in an internal trusted repository. Other TA interfaces provide user certificates and public system parameters. TA is an element of standard public key infrastructure. It means that TA should be considered subordinate to other TA and plays the role of intermediate certificate authority, which has a certificate issued by its predecessor, say CA_{prev}, in the trust model used. TA certificate issued by CA_{prev} contains system parameters (including TA's public key) and authorizes TA to issue certificates to end users.

Public Repository (PR) component stores information that can be kept in untrusted repository, i.e. if adversary gets any information from PR, he will not be able to decrypt any secret information. The PR stores public information required in the scheme (i.e. shares, access structures, ciphertext). Security analysis with description what can be made public is presented in further sections.

A Dealer component contains implementation of dealer specific functions from the scheme. The two most important functions are the shares and ciphertext creation. The dealer component gets all required information (e.g. user certificates and public system parameters) from TA (from its internal repository). The component sends ciphertext and shares to PR through appropriate interfaces. This component also can create and publish access structures. Additionally, it is possible that the dealer in on-line mode sends the shares or ciphertext directly to users (i.e. decrypting components). This variant of the architecture is not presented on the diagram.

A Decrypter is a component that contains all functions necessary to decrypt a document. It contains two major internal components. One is responsible for subdecryption process and the second for combining the partial decrypted values into a clear text. The decrypter component also handles requests from other decrypting components to subdecrypt the document.

4.2 Security Considerations

The approach to security issues presented in RFC 5408 [8] concerning IBE architecture is used to determine CIBE-GAS-H system security. The security of cryptographic algorithm comprising CIBE-GAS scheme against IND-CID-GO-CPA attacks was described in [1]. Due to this fact the following theorem can be formulated:

Theorem 1. The proposed CIBE-GAS-H encryption scheme is secure against adaptive chosen ciphertext attack IND-CID-GO-CCA2, assuming that (1) the hash function H_7 is modelled as random oracle and (2) the underlying CIBE-GAS scheme is an ID-OW-CPA secure encryption scheme.

The proof is similar to the proof of [13] and is omitted here.

We assume that the adversary is not able to access or change the cryptographic material of a TA. To achieve this goal the key material should be protected with FIPS 140-2 [18], Level 3 validated hardware that performs all key management, key storage, and key operations (such as digital signing and decryption) exclusively within hardware.

It is assumed also that authentication is done before communication between any two components (with some exceptions regarding Public Repository, described later). The authentication could be done using standard TLS protocol, for example.

The TA component belongs to some trusted zone of standard PKI. It is protected by physical, organizational and operational measures according to the best practices, e.g. [19]. Three basic reasons for binding TA with standard PKI are as follows (compare also [20]):

(a) resistance against Denial-of-Decryption (DoD) attack;
(b) solution of the public keys distribution problem for encryption schemes;
(c) PKI commercial maturity.

DoD attack is similar to Denial of Service (DoS). In DoD attack the adversary cannot gain any secret information, but any authorised user is also not able to decrypt this information and get the normal service. The adversary can succeed to launch this attack since there is no checking whether the public key is associated with the corresponding person or not.

The problem in distributing public keys for encryption schemes can be derived from DoD attack, and relies on the fact that the encrypter cannot correctly identify which public key to use from a range that are made available to him, knowing that choosing the wrong one will result in his message not getting through.

The certificate and ID-based public key cryptography is a new technology and at present that technology mainly exists in theory and is being tested in practice. This is in contrast to PKI-based cryptography, which has been an established and widespread technology for many years already. Therefore, the proper integration of a new encryption schemes with existing PKI architectures is required and should speed up their entering the market.

Public repository stores and provides encrypted documents and other public information required to decrypt documents. However, in CIBE-GAS-H (CIBE-GAS) scheme it is not possible to decrypt documents using only information from PR. Hence, it is not necessary to protect documents from unauthorized disclosure. The attack, which will result in modification of PR, will cause that the decryption of documents by authorized users might be impossible. Public repository does not require authentication, although authentication prevents unauthorized users (i.e. users which are not members of any group in the general access structure) from learning who is authorized to decrypt specific documents. Hence, further it is assumed that PR also requires authentication and provides information only to authorised users.

Dealer and decrypter components use secret keys internally. These keys must be protected from unauthorized disclosure. The components might use hardware components for key protection. The attacks compromising cryptographic keys or authentication between components will defeat the security of CIBE-GAS system.

During the design of CIBE-GAS system the following types of attacks where considered: passive monitoring of communication channels, masquerade as TA and denial of service. All message exchanges between components are protected by TLS protocol. CIBE-GAS system relies on TLS security mechanism established to prevent masquerade and passive monitoring (like in the architecture proposed in [21]). As the protection against DDoS attacks is generally very difficult, CIBE-GAS system relies on firewall, IDS or other network security mechanisms to protect against these kinds of attacks.

4.3 Messages

The system architecture is service oriented and XML messages are exchanged between components using SOAP protocol. These messages are designed for standalone CIBE-GAS system. If CIBE-GAS system is integrated with other system, then messages can be encapsulated into existing formats, e.g. XML Encryption [22].

Names of messages' XML schemas exchanged between components are presented on Fig.2. There are 8 basic definitions of xml messages (*CertRequest, Certificate, RightsDelegation, Ciphertext, Shares, AccessStructure, Users, PartiallyDecryptedValue*).

Fig.3 contains content model view of XML schemas of *Ciphertext*, *Shares* and *AccessStructure* messages, which are necessary to decrypt a document. The *Ciphertext* message consists of cipher values c_K consisting of C_1, C_2, C_3, C_5, C_6 and C_4 for each privileged set *A,* which contains an encrypted symmetric key material and encrypted message c_M . *Shares* messages contain public share parameters. *AccessStructure* defines *A* sets. The *Ciphertext* message also contains *SharesID* and *AccessStrutureID* attributes which are references to related share parameters and access structure, respectively.

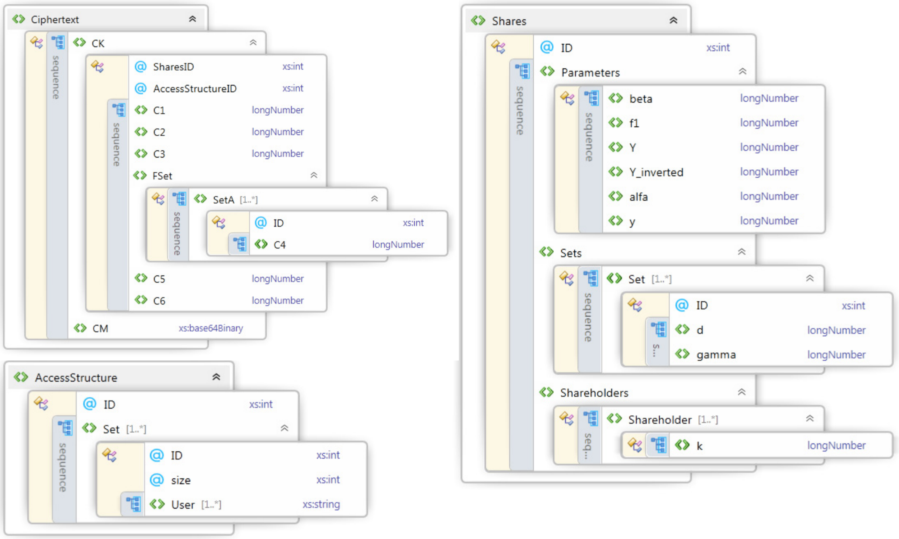


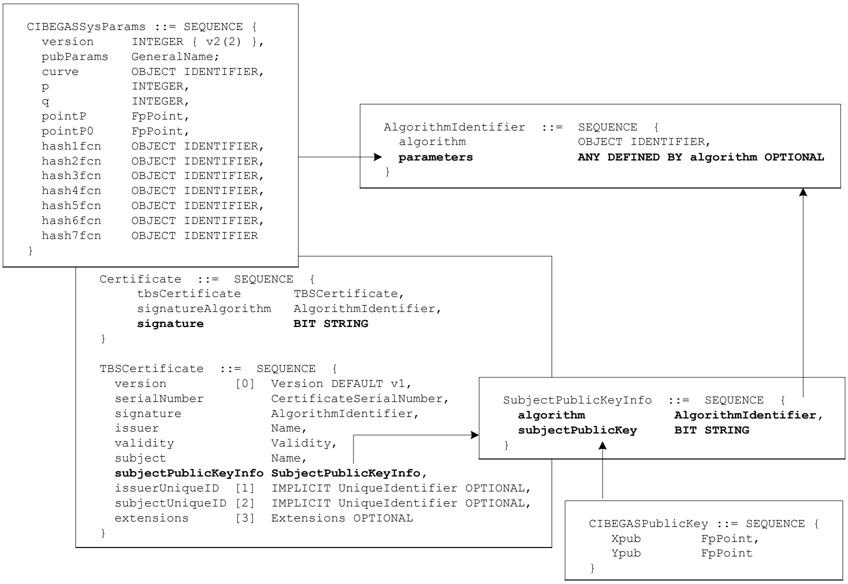
Fig. 3. Content model view of chosen messages

After downloading *Ciphertext* user downloads *Shares* and *AccessStructure*. Next step is determination of *A* set (or sets) to which the user belongs and for which *Ciphertext* contains corresponding C_4 value. When the user decides which *A* set he will use to decrypt document (playing combiner role), he sends to PR request in the message *UserRequest* and receives the message *Users* with IP addresses of other users from the chosen *A* Set. IP addresses are used to get from other users their partially decrypted values in on-line mode. These values are send using *PartiallyDecryptedValue.*

The *CertRequest* message (user name, public key(*X, Y*)) is used by users to request *Certificate* from TA. The *Certificate* message encapsulates user certificate in X.509 format (see section 4.4). Public system parameters (pairing parameters, hash functions, TA public values *P* and P_0) required to perform every operation using CIBE-GAS library component are encapsulated inside user certificate. *RightsDelegation* message is a special message send by a user to a dealer when the user wants to delegate his right to decrypt a document to another user.

4.4 Public Key Certificate in X.509 Format

Here we omit the contents of TA's certificate and show only how to extend the semantics of an end entity's X.509 certificate to apply CIBE-GAS-H scheme in practice. The CIBE-GAS-H scheme is different from the conventional public key algorithm and it means that public key and its algorithm identifier should be explicitly included in the certificate.



```
CIBEGASSysParams ::= SEQUENCE {
  version     INTEGER { v2(2) },
  pubParams   GeneralName;
  curve       OBJECT IDENTIFIER,
  p           INTEGER,
  q           INTEGER,
  pointP      FpPoint,
  pointP0     FpPoint,
  hash1fcn    OBJECT IDENTIFIER,
  hash2fcn    OBJECT IDENTIFIER,
  hash3fcn    OBJECT IDENTIFIER,
  hash4fcn    OBJECT IDENTIFIER,
  hash5fcn    OBJECT IDENTIFIER,
  hash6fcn    OBJECT IDENTIFIER,
  hash7fcn    OBJECT IDENTIFIER
}
```

```
AlgorithmIdentifier  ::=  SEQUENCE  {
  algorithm               OBJECT IDENTIFIER,
  parameters              ANY DEFINED BY algorithm OPTIONAL
}
```

```
Certificate  ::=  SEQUENCE  {
     tbsCertificate       TBSCertificate,
     signatureAlgorithm   AlgorithmIdentifier,
     signature            BIT STRING
}

TBSCertificate  ::=  SEQUENCE  {
  version         [0]  Version DEFAULT v1,
  serialNumber         CertificateSerialNumber,
  signature            AlgorithmIdentifier,
  issuer               Name,
  validity             Validity,
  subject              Name,
  subjectPublicKeyInfo SubjectPublicKeyInfo,
  issuerUniqueID  [1]  IMPLICIT UniqueIdentifier OPTIONAL,
  subjectUniqueID [2]  IMPLICIT UniqueIdentifier OPTIONAL,
  extensions      [3]  Extensions OPTIONAL
}
```

```
SubjectPublicKeyInfo  ::=  SEQUENCE  {
  algorithm            AlgorithmIdentifier,
  subjectPublicKey     BIT STRING
}
```

```
CIBEGASPublicKey ::= SEQUENCE {
   Xpub        FpPoint,
   Ypub        FpPoint
 }
```

Fig. 4. X.509 certificate structure

The Fig.4 presents the syntax of X.509 certificate compliant with ASN.1. The CIB-GAS entity's public key should be included in the *subjectPublicKeyInfo* basic field of the certificate. This field has two subfields: *algorithm* and *subjectPublicKey*. The value for *algorithm* field can be the public key algorithm identifier with its parameters included into *CIBEGASSysParams* structure. The value for *subjectPublicKey* is bit string of DER encoding of public key given in *CIBEGASPublicKey* structure.

The *signatureValue* field contains a digital signature computed upon the ASN.1 DER encoded *tbsCertificate*. For CIBE-GAS scheme, the value of this field is the Boneh *et al.*'s short signature [20, 23] of the certificate information of the *tbsCertificate* field.

5 Conclusions

The proposed CIBE-GAS-H system is designed to work with messages of arbitrary length. These messages can be encrypted using the rules contained in access structures. In practice, it is possible using access structures to describe any access rules (e.g. hierarchical or threshold ones). The proposed architecture visualises the properties of the system. The most important properties are security related. The system is divided into components based on security analysis. Only the successful attack on the security of TA will compromise the complete system. The architecture is also designed to be scalable. Users can be added and removed dynamically and the length of a message is practically not restricted.

In the paper, we show only how to combine the two level CIBE-GAS-H certification domain with traditional PKI. However, hierarchical identity based cryptography HIBC constructions (see [24]) can be used to extend the scheme and propagate down the trust to identities that are not registered at the trusted node (TA level).

Future work will be carried out simultaneously in two directions. In the first the risk analysis method will be applied to CIBE-GAS-H system [25, 26] and in the second one our method for long-term preservation of documents digital signatures [27] will be enhanced to support confidentiality using CIBE-GAS-H technique.

Acknowledgment. This scientific research work is supported by NCBiR of Poland (grant No PBS1/B3/11/2012) in 2012-2015.

References

1. Hyla, T., Pejaś, J.: Certificate-Based Encryption Scheme with General Access Structure. In: Cortesi, A., Chaki, N., Saeed, K., Wierzchoń, S. (eds.) CISIM 2012. LNCS, vol. 7564, pp. 41–55. Springer, Heidelberg (2012)
2. Sang, Y., Zeng, J., Li, Z., You, L.: A Secret Sharing Scheme with General Access Structures and its Applications. International Journal of Advancements in Computing Technology 3(4), 121–128 (2011)
3. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (1985)
4. Boneh, D., Franklin, M.: Identity-based encryption from the Weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001)
5. IEEE-P1363.3. IEEE P1636.3TM/D1 draft standard for identity-based public-key cryptography using pairings (2008)
6. RFC 6508 Sakai-Kasahara Key Encryption, SAKKE (2012)
7. RFC 5091 Identity-Based Cryptography Standard (IBCS) #1: Supersingular Curve Implementations of the BF and BB1 Cryptosystems (2007)
8. RFC 5408 Identity-Based Encryption Architecture and Supporting Data Structures (January 2009)
9. Chatterjee, S., Sarkar, P.: Identity-Based Encryption. Springer, New York (2011)
10. Galbraith, S.D., Paterson, K.G., Smart, N.P.: Pairings for cryptographers. Discrete Applied Mathematics 156(16), 3113–3121 (2008)

11. Al-Riyami, S.S., Paterson, K.G.: Certificateless public key cryptography. In: Laih, C.-S. (ed.) ASIACRYPT 2003. LNCS, vol. 2894, pp. 452–473. Springer, Heidelberg (2003)
12. Chen, L., Cheng, Z., Malone–Lee, J., Smart, N.P.: Efficient ID-KEM based on the Sakai–Kasahara key construction. IEE Proceedings, Information Security 153(1), 19–26 (2006)
13. Bentahar, K., Farshim, P., Malone-Lee, J., Smart, N.P.: Generic constructions of identity-based and certificateless KEMs. Journal of Cryptology 21, 178–199 (2008)
14. Burak, D., Chudzik, M.: Parallelization of the Discrete Chaotic Block Encryption Algorithm. In: Wyrzykowski, R., Dongarra, J., Karczewski, K., Waśniewski, J. (eds.) PPAM 2011, Part II. LNCS, vol. 7204, pp. 323–332. Springer, Heidelberg (2012)
15. Cramer, R., Shoup, V.: Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. SIAM Journal on Computing 33, 167–226 (2004)
16. Bell, M.: Service-Oriented Modeling (SOA): Service Analysis, Design, and Architecture. Wiley & Sons (2008) ISBN 978-0-470-14111-3
17. Lynn, B.: PBC Library Specification, `http://crypto.stanford.edu/pbc/` (retrieved 2013)
18. FIPS PUB 140-2: Security Requirements for Cryptographic Modules, National Institute of Standards and Technology (2001)
19. Souppaya, M., Wack, J., Kent, K.: Security Configuration Checklist Program for IT Products - Guidance for Checklist Users and Developers, NIST Special Publication SP 800-70 (May 2005)
20. Boneh, D., Lynn, B., Shacham, H.: Short signatures from the weil pairing. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 514–532. Springer, Heidelberg (2001)
21. Appenzeller, G., et al.: Identity-Based Encryption Architecture and Supporting Data Structures, RFC5408, IETF (2009)
22. Imamura, T., et al.: XML Encryption Syntax and Processing. W3C Recommendation (2002)
23. Boneh, D., Gentry, C., Lynn, B., Shacham, H.: Aggregate and verifiably encrypted signatures from bilinear maps. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 614–629. Springer, Heidelberg (2003)
24. Gentry, C., Silverberg, A.: Hierarchical ID-based cryptography. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 548–566. Springer, Heidelberg (2002)
25. El Fray, I., Kurkowski, M., Pejaś, J., Maćków, W.: A New Mathematical Model for Analytical Risk Assessment and Prediction in IT Systems. Control and Cybernetics 41(1), 241–268 (2012)
26. El Fray, I.: A Comparative Study of Risk Assessment Methods, MEHARI & CRAMM with a New Formal Model of Risk Assessment (FoMRA) in Information Systems. In: Cortesi, A., Chaki, N., Saeed, K., Wierzchoń, S. (eds.) CISIM 2012. LNCS, vol. 7564, pp. 428–442. Springer, Heidelberg (2012)
27. Hyla, T., El Fray, I., Pejaś, J., Maćków, W.: Long-term Preservation of Digital Signatures for Multiple Groups of Related Documents. IET Information Security 6(3), 219–227 (2012)

Study of Security Issues in Pervasive Environment of Next Generation Internet of Things

Tapalina Bhattasali[1], Rituparna Chaki[2], and Nabendu Chaki[1]

[1] Department of Computer Science & Engineering
University of Calcutta, Kolkata, India
tapolinab@gmail.com, nabendu@ieee.org
[2] A.K. Choudhury School of IT
University of Calcutta, Kolkata, India
rituchaki@gmail.com

Abstract. Internet of Things is a novel concept that semantically implies a world-wide network of uniquely addressable interconnected smart objects. It is aimed at establishing any paradigm in computing. This environment is one where the boundary between virtual and physical world is eliminated. As the network gets loaded with hitherto unknown applications, security threats also become rampant. Current security solutions fail as new threats appear to destruct the reliability of information. The network has to be transformed to IPv6 enabled network to address huge number of smart objects. Thus new addressing schemes come up with new attacks. Real time analysis of information from the heterogeneous smart objects needs use of cloud services. This can fall prey to cloud specific security threats. Therefore need arises for a review of security threats for a new area having huge demand. Here a study of security issues in this domain is briefly presented.

Keywords: Internet of Things, Any Paradigm, Smart Objects, Security Issues, IPv6, Cloud Services.

1 Introduction

Kevin Ashton, cofounder and executive director of MIT Auto-ID Centre, first introduced the term "Internet of Things" in 1999 [1]. It is an emerging concept where smart objects are equipped with sensors or actuators, tiny microprocessor, communication interface, and power resource. As per Libelium report [2], it is estimated that more than 50 billion devices will be accessing the Internet by the year 2020. The tremendous increase in demand makes mapping of resource constrained sensor nodes and Internet a big challenge. The term 'Thing' used in "Internet of Things" can be defined as physical or digital or virtual entity that is capable of being identified [3] to integrate heterogeneous data, semantics, and objects. IoT combines several techniques such as RFID,Zigbee,Wi-Fi,3G/4G,embedded devices, sensing devices.

To ensure sensing data availability at any time, at any place,effective processing of large amounts of collected sensing data is necessary in the application areas such as

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 206–217, 2013.

environmental monitoring, weather forecasting, transportation, business, healthcare, military application etc. Combining wireless sensor networks with cloud makes it easy to share and analyze real time sensor data on-the-fly. The issues of storage capacity may be overcome by low-cost cloud computing technique [4][5]. For security and easy access of data, it is widely used in distributed and mobile environment. The objective of the integrated sensor-cloud framework is to facilitate the shifting of high volume of sensing data from sensor networks to the cloud computing environment; so that scientifically and economically feasible data can be fully utilized to visualize the concept of next generation Internet i.e. Internet of Things. Figure 1 represents "any" paradigm in the context of IoT. In order to provide anytime, anywhere services, a number of smart sensing objects attached to Internet have to communicate with each other through uniquely assigned identity [6]. This is where128 bits IPv6 steps in. So that, it can support 2^{128} addresses, which is approximately 340 undecillion or 3.4×10^{38} addresses.

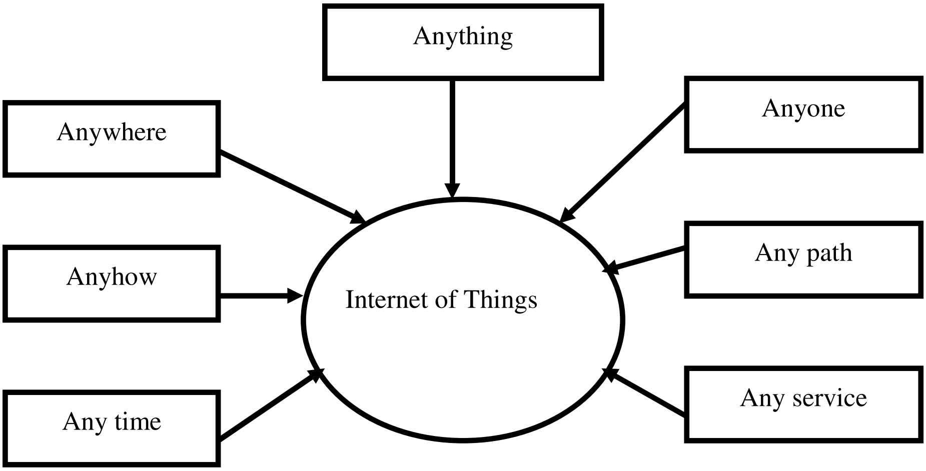


Fig. 1. "Any" Paradigm in IoT

The design of protocol stack for smart objects must be matched with existing Internet hosts in order to create the extended Internet, which is the aggregation of Internet with the IoT. It is very critical to implement IPv6 in low powered sensing objects. As IPv4 address spaces are already used, existing IPv4 (32 bits) enabled systems need to be upgraded to IPv6 compatible devices. The huge number of heterogeneous devices being connected to one another gives rise to newer security threats. The amount of flexibility and availability of information also leads to hitherto unforeseen security breaches. At present, there is a lack of analyzers for the new threats in the network. Securing IP based ubiquitous sensor network (IPv6 or IPv4-Ipv6) in IoT is of great concern for researchers to meet future market demands and satisfaction. In comparison to IPv4, IPv6 provides simplicity, improved routing speed, quality of service and security. IPv6 brings significant assurance of a higher level of security and confidentiality of the transmitted data.

The heterogeneous, resource-constrained and distributed natures of the network make conventional security methodologies inefficient. More research works are required to ensure security, performance and interoperability between next generation IoT and existing Internet.

The remainder of this paper is organized as follows. Section 2 introduces the concept of next generation network i.e. IoT. Section 3 illustrates open issues in IoT environment. In section 4, possible security threats are explained briefly. Section 5 gives basic idea about security solutions in IoT environment. Section 6 discusses about a case study in this environment. It is followed by a conclusion in section 7.

2 Next Generation Internet of Things

IoT is an integrated part of future Internet, which may be termed as next generation network. Ambient intelligence which is hidden in IoT environment supports people in carrying out their everyday activities in an easy and natural way. Radio-frequency identification (RFID) is often seen as essential requirement for serving Internet of Things. If all objects around the environment are equipped with radio tags, they can be easily identified. This technology is very useful in health monitoring applications such as automated monitoring of patient's heart condition. In order to track patients' medical history, RFID chips can be implanted in their body. Internet-connected devices can also directly communicate with the required emergency services when sensed data shows the sign of deterioration in the patient's condition .In this way pervasive environment of smart healthcare technique has the capacity to save lives.

Pervasive computing, which is often similar to ubiquitous computing, is an important field of research leading to the domain of IoT. Tremendous developments in technologies such as wireless communications, mobile computing, wearable computers, sensors, RFID tags have led to the evolution of next generation IoT. The goal of pervasive computing is to create ambient intelligence where devices embedded in the environment always provide services to improve quality of life by hiding underlying technologies. In this pervasive environment, intelligent objects are interconnected to autonomously collect, process and transport data in a cooperative way, in order to adapt IoT concept [7]. It can be said that, IoT is more vulnerable to serious security threats than existing network [1] because it includes various constraint such as low resource objects, heterogeneous nature, open environment deployment, dynamic nature. In the next section, some of the considerable open issues for security implementation in IoT are discussed.

3 Open Issues in IoT Environment

In IoT environment, the system needs a flexible security mechanism to update it easily according to the requirement. Both the user and the system need a secure access control and authorization mechanism within the limitations imposed by IoT environment. Some of the open issues are as follows.

- IoT creates a heterogeneous environment facilitated by mobility of the objects. It can give rise to inconsistent interpretations of data collected from different domains. Due to distributed and ad-hoc nature, it is open to several unique vulnerabilities whose solutions are unknown. There is no central control that can provide required security features. Therefore burden of the security features may be too large for small and limited capacity objects.
- In this surrounding, a secure communication channel is needed along with object authentication to track interacting objects. But request for establishing secure channel is also transmitting through the shared, unreliable wireless medium. Therefore feasible solution for tracking objects through insecure channel is still in the phase of research.
- If data is shared with unknown objects, the probability of data security reduces automatically. The exact criteria for trust establishment between communicating parties need to be determined.
- To work in a smarter way, pervasive environment needs to deal with users' personal data. But sometimes it poses serious threat to the privacy of the user, especially in the situations where people do not want to disclose their detailed personal information.

Therefore an intelligent system should be considered to analyze these issues and to adapt dynamic mechanism. Next section focuses on possible security threats in IoT environment.

4 Possible Security Threats

Various security challenges can arise in heterogeneous features of IoT [8]. There are several known attacks in IoT environment whose solutions are available in the market. But aim of this paper is to consider novel challenges in IoT environment whose feasible solutions are still under research. Most of the security issues such as eavesdropping, false routing, message tampering, unauthorized usage, DoS attack are common with existing internet. But the issues related to specific attacks may be quite different. Some issues like secret extractions, tampering of nodes are more serious in IoT environment. This paper considers that security challenges in IoT may occur from compromised wireless sensor network, usage of novel IPv6 protocol, integrated cloud environment, smart sensing objects. In our previous papers [9][10][11], we have already discussed about security related issues in wireless sensor network. Now the main focus is to identify new threats in IPv6 based integrated sensor-cloud environment, which is based on number of smart sensing objects.

4.1 Security Attacks Related to IP-Enabled Environment

There are several existing threats in IP-enabled environment. Security in IPv6 is almost same as IPv4 security in many ways. IPv6 is normally considered as more secure than IPv4 because version 6 includes the concept of IP Security (IPSec). Beside this, some significant differences exist between IPv4 and IPv6 [12]. In the transition period, coexistence of IPv4 and IPv6 especially creates problem regarding security issues. It is because transition mechanisms provide new, previously unknown possibilities of intrusion. There are several transition mechanisms, such as tunneling, dual-stack configurations. It is very important to understand security implications of the transition mechanisms in order to apply proper security mechanisms. On dual-stack configuration hosts applications can be targeted by both IPv4 and IPv6 attacks. Tunneling mechanisms may also bring misuse possibilities. Tunneling, especially automatic tunneling can facilitate an intruder to avoid ingress filtering checks. Among two major methods of automatic tunneling, "6 to 4" method encapsulates IPv6 packet directly into an IPv4 packet and “Teredo" method encapsulates IPv6 packet into an IPv4 UDP packet. By misusing 6to4 transition mechanism, a DoS attack can be targeted to IPv6 node, IPv4 node or other 6 to 4 node. In ‘Teredo” tunneling, all receiving nodes must allow decapsulation of packets that can be sourced from anywhere. This can be a serious security problem. Addresses within IPv4 and IPv6 headers may be spoofed to be used for Denial of Service (DoS) attacks. In this section, the main focus is on new attacks which may arise in IPv6 or IPv6-IPv4 environment. Some of the specific attacks in IP enabled network are briefly discussed.

- Reconnaissance Attack

IPv4 reconnaissance attack uses ping sweep and port scan techniques. It can be mitigated by filtering certain types of messages used by an intruder. The default subnet size of an IPv6 subnet is 64 bits, or 2^{64}, compared to the subnet size in IPv4 of 8 bits, or 2^{8}. This increases the scan size to check each host on a subnet by 2^{64} - 2^{8} (approximately 18 quintillion). So ping sweep and port scan are much more difficult to complete in an IPv6 network. New multicast addresses in IPv6 enable intruder to identify key resources such as routers more easily. Additionally, IPv6 networks are even more dependent on ICMPv6 to function properly. Aggressive filtering of ICMPv6 can have negative effects on network functions. As public services on the network need to be reachable with DNS, adversary can attack at least a small number of critical hosts within the victim network. DNS Server can be easily compromised because of using dynamic DNS mechanisms for large nature of IPv6 addresses and the lack of a strict requirement for Network Address Translation (NAT).

- Fragmentation Attack

Minimum recommended MTU size for IPv6 is 1280 octets. IPv6 protocol specification does not allow packet fragmentation by intermediary devices. It is recommended to drop all fragments less than 1280 octets unless the packet is the last in the flow. However an intruder may achieve port numbers by using fragmentation and can bypass security monitoring devices. An attacker can also cause overload of reconstruction buffers on the target system by sending a large number of small fragments, which forces a system to crash.

- ICMPv6 Misuse Attack

Some important mechanisms in IPv6 networks, such as neighbor discovery and path MTU discovery, are dependent on some types of ICMPv6 messages. ICMPv6 allows error notification response to be sent to multicast addresses, which can be misused by attacker. An attacker can cause multiple responses targeted at the victim by sending packet to a multicast address.

- Routing Header Misuse Attack

IP options in IPv4 are replaced with extension headers in IPv6. All IPv6 nodes are capable to process routing headers. It is possible that an intruder sends a packet to a publicly accessible address with a routing header containing forbidden address i.e. address of the victim network. Then the publicly accessible host will forward the packet to a destination address stated in the routing header even though that destination address is filtered.

There are several known and unknown security threats which can arise in IP enabled network. Some of the devastating attacks are discussed here. Next subsection describes security threats of the sensor-cloud environment.

4.2 Security Threats of Sensor-Cloud Environment

Sensor-cloud environment can be easily compromised by the adversary because of the absence of centralized control[13]. In this virtual environment, main security issue includes violation of authentication and leakage in communication channel. As sensor-cloud framework is deployed in distributed environment, there may appear several security challenges [14] [15] [16] that comes from the following perspectives. Environment of sensor nodes can be compromised or individual sensor can be vulnerable to attacks, whose solutions are available. Information flows within the cloud can be affected by compromised cloud nodes. The cloud client can be infected by malicious code, which can lead to further security breaches within the environment. The communication channels between sensors and cloud and between client and cloud are vulnerable to different types of attacks. Some of the typical attacks in this environment are SPAM over Internet Telephony (SPIT), where an attacker sends bulk unsolicited calls to an enterprise; The spammer attempts to initiate a voice session and then relays a prerecorded message if the callee answers; Denial of Service (DoS), which is an severe issue for any IP network- based service; Service theft, where an unauthorized users get access to the network that results into authentication violation.

In the next subsection, the main focus is on security risks for the smart objects in IoT.

4.3 Security Risks for Smart Objects Which Form Basis of IoT

In IoT environment, the solutions of some known attacks such as man-in-the-middle attack, eavesdropping, routing attacks are already known. The transmission phase in IoT environment may be vulnerable to man-in-the middle attacks, where it is assumed that no third party is able to sit in between the two communicating entities. In completely automated mechanisms, there is usually no prior knowledge about each other and can not always be able to identify intruder. During transmission between smart objects in a network, it may be susceptible to eavesdropping, either for insufficient protection of communication medium or for use of compromised session key. Routing information in IoT can be spoofed, altered, or replayed. Other known relevant routing attacks include sinkhole attack or blackhole attack, selective forwarding, wormhole attack, sybil attack. Other security threats related to smart sensing objects are as follows [17][8].

- Privacy Threat

Tracking of object's location and usage may give rise to privacy risk for its users. An attacker can gather information about individual object to find out behavioral patterns of the users. Such information can be distributed to interested parties for marketing purposes. A smart object deployed in the ambient environment can easily be captured by an attacker. Such an attacker may then attempt to extract security information to misuse it.

- Firmware Replacement Attack

When a smart object is in operation or maintenance phase, its firmware or software may be updated to allow for new functionality or new features. An attacker may be able to exploit such a firmware upgrade by replacing the object with malicious software, thereby influencing the operational behavior of the object.

- Cloning of Smart Objects by Untrusted Manufacturer

During the manufacturing process of a smart object, an untrusted manufacturer can easily clone the physical characteristics or security configuration of the object. Cloned object may be sold at a cheaper price in the market, and be able to function normally as a genuine object. In the worst case, a cloned object can be used to control a genuine object. An untrusted manufacturer may also change functionality of the cloned object, resulting in degraded functionality with respect to the genuine object. It can also implement additional functionality with the cloned object, such as a backdoor. During the installation of object, a genuine object may be substituted with a similar variant of lower quality without being detected. The main motivation may be cost savings. Genuine objects can be reselled in order to gain further financial benefits. Another motivation may be to damage reputation of competitors.

After going through different security threats from heterogeneous perspective, next section focuses on possible security solutions in IoT.

5 Security Solutions in IoT Environment

Security in IoT environment should address the following main issues.

- Enabling smart and intelligent behavior of networked objects.
- Preservation of privacy for heterogeneous sets of objects.
- Decentralised authentication and trust model.
- Energy efficient security solutions.
- Proper authentication of the objects within the network.
- Security and trust for cloud computing services.
- Data ownership.

Security should ensure accurate implementation of confidentiality, integrity, authentication, non-repudiation and access control. Two of the main security issues in the IoT are privacy and confidentiality. Because of the scale of deployment and mobility, the cloud of "things" is hard to control. Cryptographic techniques are useful to protect confidential information stored in the network and to transfer secure messages from one ubiquitous node to another. But there are some gaps between available techniques and requirements. Implementing cryptography into the network of smart objects creates tremendous challenges. Because traditional mechanisms are slow in speed, large in size and consume more power and may fail to provide necessary protection for sensed data. Lightweight Cryptography[18][19] is a current field of research that can meet the constraints set by the use of smart objects. There is no strict criterion for classifying a cryptographic algorithm as lightweight but the basic concept is that cryptography techniques need to work by using minimum amount of essential resources of target objects. It can be categorized into hardware-oriented and software-oriented. Hardware-oriented techniques are more applicable in areas where main concern is about the size of chip and number of clock cycles required for its execution. Software oriented techniques need to be considered when main focus is on memory (Ram and ROM) requirements, and power consumption. Another type of categorization is symmetric versus asymmetric. Asymmetric cryptographic algorithms offer more security than symmetric, but symmetric technique works well, where authentication and integrity are of prime importance than non-repudiation and confidentiality. This can save additional computational cost and power consumption. Asymmetric ciphers are computationally far more demanding in both hardware and software levels. Finally it is the designer's decision to choose the appropriate techniques based upon the application's requirements along with the constraints carried by them.

In some cases, maximum security can only be achieved by designing an effective intrusion detection system which is not intended to prevent attacks. Instead, their purpose is to provide alert about possible attacks, ideally in time to stop the attack or to mitigate the damage. Researchers have been working for quite some time for designing intrusion detection in wireless sensor network and in IP based network. IoT involves heterogeneous network and thus an integration of techniques is needed. Not much work has been done in the field of IoT environment. There exist some open source IDS systems for IPv4 network. By using software, IDS systems in IPv4 networks, procedure of intrusion detection can be automated. In this, intrusion attempt is recognized and logged by IDS system and alert is generated. There is no freeware IDS software for IPv6 networks. By using packet analyzer tools, an intrusion detection procedure will require an efficient network administrator. Therefore IPv6 supporting IDS system must consider IPv6 protocol specific features.

- IPv6 defines a new header format, which must be properly recognized by IDS.
- IDS must implement support for IPv6 extension headers and to check the order of extension headers. It is recommended for IDS to discard a packet with an undefined "Next Header" value and to record this as incident.
- IDS should be capable of detecting duplicate options of Hop-by-Hop option header or destination options header.
- IDS with IPv6 support should also be able to recognize and analyze IPv6 traffic tunneled in IPv4.

Some of the existing works on intrusion detection in sensor network based IoT are mentioned below. Most of the researches are directed to wireless sensor network which is a main component of IoT. The lack of fixed infrastructures and scarce resource make WSN difficult to collect audit data for the entire network. It is more difficult to distinguish false alarms and real intrusions. In one of the previous papers [9], a survey of recent IDS in sensor network has been presented. In the earlier works, different lightweight hierarchical models [10][11] are proposed for heterogeneous wireless sensor network to detect DoS type attacks. But a sensor node can utilize most of the services offered by traditional IP networks with the help of IP stack. Without a proper security framework, it is not possible to grow towards IoT environment. So our research work currently focuses on this domain. In [20], a dynamic coding mechanism to implement distributed signature based lightweight IDS has been proposed in IP based Ubiquitous Sensor Networks. In [21], intrusion detection and response system has been proposed considering different types of attacks arise from internet hosts, clients and insider. Main module of the IDS resides on the gateway which supports dual stack- one for analyzing packets from Internet (IPA) and another for analyzing packets from USN (UPA) for detecting attacks. In [22], a distributed intrusion detection scheme for IoT has been proposed based on anomaly mining where intrusion semantic is analyzed to distinguish intrusion behaviors from anomalies; since all anomalies are not triggered by malicious intrusion. In [23], to detect the security threats in IoT, artificial immune system is applied where detectors evolve dynamically to detect new IoT attacks. Newly detected attacks are combined with the attack information library to alarm the manager of the IoT. The problem on how to consider new class of security threats in IP-enabled IoT is currently a challenging research issue.

Therefore a need of efficient security mechanism specifically tailored for this purpose is inevitable. In the next section, a pervasive environment based real life application of IoT has been briefly discussed to give idea about security implementation in next generation network.

6 Case Study: Pervasive Environment of Healthcare

A healthcare system based on pervasive environment controls health data in an electronic format EPR (Electronic Patient Record) as compared to the largely paper-based Records[24][25]. As EPRs are kept on networked systems for availability reasons, it is accessible from anywhere and is very easy to copy. Examples include confidential personal data like HIV status, psychiatric records, genetic information etc. The usage of EPRs, imposes new security risks to health data. This may lead to unauthorized access and tampering of sensitive EPRs. The pervasive use of wireless techniques makes it easy for malicious adversaries to launch security attacks. As healthcare systems are designed to assist in medical treatment, security vulnerabilities lead the entire system unreliable, putting patients' lives at risk. Some of the most possible vulnerabilities to this type of healthcare systems include the following.

- In case of emergencies, false alarms may be generated or real alarms may be suppressed by the system.
- Alteration of health data of specific patients, leading to incorrect diagnosis and treatment.

Another concern has been significantly increased over privacy issues, relating to electronic health data. Reliable healthcare systems must ensure same level of privacy policy for electronic data as applicable to paper based patient records. The main idea behind securing it is to preserve patient privacy. To ensure this, care needs to be taken to prevent all unauthorized access to EPRs in the system. An important property of a medical system is that patients have a high level of control over deciding who accesses their health information. Two additional issues associated with pervasive healthcare systems are security of wireless communication and physical security of handheld devices. Pervasive healthcare systems make extensive use of wireless communication technologies to communicate health data collected by sensor networks, which have many security vulnerabilities. Portable handheld devices that are used by both patients and caregivers, may store sensitive health information about the patient and cause a serious privacy breach, if stolen or misplaced. Therefore, physical security of the devices also has to be considered.

Recent development of cloud computing allows systems to store all or selective health data in cloud storage and ensures availability with reduced expenditures. In cloud design, data is stored on multiple third party servers where the storage can be accessed on demand. Migrating health data into the cloud offers enormous convenience to healthcare service providers because they do not have to worry about the complexities of direct hardware management. But user privacy preservation and proper access control of health data are growing concern. A contextual patient-centric access policy may be adopted to classify the roles. Security solutions of pervasive healthcare systems may focus on protecting health data from different aspects. In recent years several promising prototypes for wearable health monitoring have started to emerge. These devices are being used for continuous monitoring of patients for a long time. But effective security implementation for this type of pervasive applications is an unexplored area till date.

7 Conclusion

IoT is the emerging technology of the present age which consists of the analysis of new evolving data from heterogeneous sources for creating a new era of real life applications. In this paper, an effort has been made to study the various security issues coming up in IoT environment. It is obvious that as IoT deals with a large number of objects, the use of wireless sensor network and cloud are almost inevitable. This gives rise to the need for transformation from IPv4 to IPv6 based connectivity to handle the huge number of embedded smart objects. The use of IPv6 increases flexibility at the cost of various known and unknown security challenges. There have been many works on securing the wireless sensor networks already. The main focus in this paper is to study security threats in the new IPv6 enabled network, cloud environment and smart sensing “things” in the context of IoT. Without proper security framework, intelligence in IoT environment may lead to major catastrophe. IoT research and innovation activities need to address security issues to support the growth of a smarter world. The possible solutions such as cryptography or intrusion detection etc. have been studied. The authors have considered a case study of healthcare system as this is definitely going to be maximum benefited in the IoT environment. This field is an emerging domain of research at present. The present survey is aimed at the construction of a secure independent living plan for the aged persons in the society in near future.

References

1. Sundmaeker, H., Guillemin, P., Friess, P.: Vision and Challenges for Realising the Internet of Things (2010) ISBN: 978-92-79-15088-3, doi:10.2759/26127
2. Libelium Unveils the Top 50 Internet of Things Applications, http://www.itwire.com/opinion-and-analysis/beerfiles/54432-libelium-unveils-the-top-50-internet-of-things-applications
3. Internet of Things in 2020: A roadmap for the future, Workshop report by EPoSS (European Technology Platform on Smart Systems Integration), http://www.iot-visitthefuture.eu/fileadmin/documents/researchforeurope270808_IoT_in_2020_Workshop_Report_V1_1.pdf
4. Dash, S.K., Mohapatra, S., Pattnaik, P.K.: A Survey on Applications of Wireless Sensor Network Using Cloud Computing. International Journal of Computer Science & Emerging Technologies 1(4) (December 2010) (E-ISSN: 2044-6004)
5. Ahmed, K., Gregory, M.: Integrating Wireless Sensor Networks with Cloud Computing. In: Proceedigs of the Seventh International Conference on Mobile Ad-hoc and Sensor Networks (2011)
6. The Internet of Things 2012 New Horizons, http://www.internet-of-things-research.eu
7. White Paper: Smart Networked Objects and Internet of Things, http://www.iot-a.eu/public/news/white-paper-smart-networked-objects-and-internet-of-things
8. Roman, R., Najera, P., Lopez, J.: Securing the Internet of Things. IEEE Computer 44(9), 51–58 (2011)
9. Bhattasali, T., Chaki, R.: A Survey of Recent Intrusion Detection Systems for Wireless Sensor Network. In: Wyld, D.C., Wozniak, M., Chaki, N., Meghanathan, N., Nagamalai, D. (eds.) CNSA 2011. CCIS, vol. 196, pp. 268–280. Springer, Heidelberg (2011)
10. Bhattasali, T., Chaki, R.: Lightweight Hierarchical Model for HWSNET. International Journal of Advanced Smart Sensor Network Systems (IJASSN) 1(2), 17–32 (2011), doi:10.5121/ijassn.2011.1202, ISSN: 2231-4482 [Online], 2231-5225 [Print]
11. Bhattasali, T., Chaki, R., Sanyal, S.: Sleep Deprivation Attack Detection in Wireless Sensor Network. International Journal of Computer Applications 40(15), 19–25 (2012), doi:10.5120/5056-7374, ISBN: 978-93-80866-55-8
12. IPv6 and IPv4 Threat Comparison and Best Practice Evaluation, Cisco, http://www.cisco.com/security_services/ciag/documents/v6-v4-threats.pdf
13. Yuriyama, M., Kushida, T.: Sensor-Cloud Infrastructure. In: Proceedings of the 13th International Conference on Network- Based Information Systems (2010)
14. Kapadia, A., Kotz, D., Triandopoulos, N.: Opportunistic Sensing: Security Challenges for the New Paradigm. In: Proceedings of the First International Conference on Communication Systems and Networks (COMSNETS) (January 2009)
15. Kapadia, A., Myvers, S., Wang, X., Fox, G.: Secure Cloud Computing with Brokered Trusted Sensor Networks. In: Proceedings of the International Symposium on Collaborative Technologies and Systems (CTS 2010) (May 2010)
16. Zhang, L., Zhao, K.: Study on Security of Next Generation Network, IEEE, Supported by Open Fund from Resources and Environment Management Lab (2008)
17. Sethi, M.: Security in Smart object Networks (2012), http://nordsecmob.aalto.fi/en/publications/theses_2012/sethi-mohit_thesis.pdf

18. Zhou, S., Xie, Z.: On Cryptographic Approaches to Internet-of-Things Security, http://www.lix.polytechnique.fr/hipercom/.../papers/ZhouSujing.pdf
19. Cirani, S., Ferrari, G., Veltri, L.: Enforcing Security Mechanisms in the IP-Based Internet of Things: An Algorithmic Overview (2013), http://www.mdpi.com/1999-4893/6/2/197
20. Amin, S.O., Siddiqui, M.S., Hong, C.S., Choe, J.: A Novel Coding Scheme to Implement Signature Based IDS in IP Based Sensor Networks. In: Conference Proceedings of International Symposium on Integrated Network Management Workshops (IM 2009), New York (2009) ISBN: 978-1-4244-3923-2
21. Amin, S.O., Yoon, Y.J., Siddiqui, M.S., Hong, C.S.: A Novel Intrusion Detection Framework for IP Based Sensor Networks. In: Conference Proceedings of International Conference on Information Networking, Chiang Mai (2009) ISBN: 978-89-960761-3-1
22. Fu, R., Zheng, K., Zhang, D., Yang, Y.: An Intrusion Detection Scheme Based on Anomaly Mining in Internet of Things. In: Conference Proceedings of the 4th International Conference on Wireless, Mobile and Multimedia Networks (ICWMMN 2011), China (2011)
23. Liu, C., Yang, J., Zhang, Y., Chen, R., Zeng, J.: Research on Immunity-based Intrusion Detection Technology for the Internet of Things. In: Conference Proceedings of the Seventh International Conference on Natural Computation (ICNC), Shanghai (2011) ISBN: 978-1-4244-9950-2
24. Dash, B.: Security in Pervasive Computing (2006), http://web.cs.wpi.edu/~kven/papers/Sec_Pervasive_Healthcare.pdf
25. Varshney, U: Pervasive Healthcare and Wireless Health Monitoring (2007), http://www.cens.ucla.edu/~mhr/cs219/varshney07.pdf

Security Issues of IPv6 Network Autoconfiguration

Maciej Rostański and Taras Mushynskyy

Academy of Business in Dąbrowa Górnicza, Faculty of Computer Science,
Cieplaka 1C, 41-300 Dąbrowa Górnicza, Poland
mrostanski@wsb.edu.pl
http://www.wsb.edu.pl

Abstract. IPv6 is a new version of IP protocol, which was defined in the series of RFC documents at the end of previous century. Although developments and improvements are conducted for many years already, a new standard still did not get such distribution as IPv4. The useful innovation and one of basic advantages of IPv6 protocol is a possibility of automatic assignment of addresses to the network devices. Such mode got the name SLAAC (StateLess Address AutoConfiguration). However, there are tasks, for implementation of which greater control is needed. In this case it is necessary to use the static addressing or DHCPv6 server for IPv6 protocol (stateful autoconfiguration). The aim of this work was to visualize an IPv6 network using stateless and stateful addressing modes and to reveal the features and security issues of the specific configurations. Those security issues need to be reminded to the administrators, as the big IPv6 migration is coming for small and medium businesses.

Keywords: IPv6, Computer Networks, Network Security, SLAAC, DHCPv6.

1 Introduction. On the Verge of the Internet of Things

Networks are growing endlessly and more and more data is being processed every day. For example, only by utilizing the amount of sensors gathering data, the Internet is becoming huge infrastructure for aggregation and delivery of constant data for the end-systems. The concept of this, called The Internet of Things, is a bit futuristic expression created by K. Ashton (see [1]). The idea is that not only typical computers, smartphones or tablets will use network for communication - but any device will. This is very relevant to the IPv6 adoption - as the IPv6 major feature is a huge addressing space and simplified network configuration, supporting massive scalability of networks, at the same time enabling end-to-end communication with devices connected to internetwork. Once enabled, this concept creates endless possibilities - such as the Talking Tree Project[1], a very

[1] Talking Tree Project Website: http://www.talking-tree.com/

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 218–228, 2013.

interesting idea of equipping a tree with sensors, cameras, etc. and utilizing complex software allowing a tree to literally tweet about the weather, noises, wind changes.

Migration to IPv6 protocol with its vast address space is a step forward into those and many other possibilities for innovative services. This is a major IPv6 adoption driver for innovative enterprises, and many of small and medium organizations are considering migration. Although the dynamics of IPv6 protocol deployment is not as high as expected, experts assume it is going to grow for a couple of next years [2]. There is an overall impression, that only legacy applications hold back the migration for many organizations, especially when they learn about many advantages of IPv6, such as simple autoconfiguration mechanisms. This article shows a different perspective on this matter.

1.1 IPv6 Is Not about Bigger Address Space Only

IPv6 protocol, being a successor to the most popular network layer protocol, is thoroughly described in RFCs and in numerous literature, such as [3], [4] or [5]. There are many advantages of IPv6 over IPv4 protocol, which, among many benefits, include:

- multi-addressing, which basically means, the node may have many IPv6 addresses, related to its function and connectivity, as well as address scope
- simplified network configuration, relying on automatic host addressing and routers sending the prefixes in router advertisements,
- directed data flows, utilizing multicast rather than broadcast transmission - in addition, IPv6 header includes Flow Label field for identifying packets within the same flow,
- simplified packet header, meaning more efficient packet processing - for example, there is no IP-level checksum,
- true end-to-end connectivity, restored by eliminating the need for Network Address Translation,
- authentication and privacy capabilities, built into protocol itself.

For many years, there is a discussion still active, whether it is a good time to migrate to IPv6. As years went by, it was becoming clearer that IPv4 is not going to vanish entirely in a fast manner; some even predict that a dual IP protocol coexistence is going to last a very long time. The organizations are reluctant in IPv6 adoption. Arbor Networks study cited in [6] indicates that possible obstacles to adoption include lack of economic incentives, lack of existing IPv6 content and technical and design hurdles.

On the other hand, the IPv4 address space is shrinking rapidly; on September 2012, RIPE NCC ran out of IPv4 addresses[2]. Technical problems diminish: most

[2] Further information: `http://arstechnica.com/information-technology/2012/09/europe-officially-runs-out-of-ipv4-addresses/`

of modern operating systems are fully IPv6 capable, the network equipment is IPv6 ready, and IPv6 knowledge is becoming more and more common.

According to the Author, the best thing about IPv6 protocol is the most surprising one: IPv6 is in fact simpler in configuration and more efficient in deployment than IPv4. Many administrators would disagree and point out the complexity of IPv6 address compared to IPv4, however considering other features, like stateful and stateless autoconfiguration of network nodes, mobility support, mandatory security protocols support, the above statement is quite defendable.

1.2 About This Paper

Myths and benefits of IPv6 deployment are very well described by Van Beijnum in [3]. The ease of configuration creates risks - some administrators would just plug the network equipment in and, while it works, won't bother with security checks. That's why this article is important - it focuses on the presentation of autoconfiguration mechanisms risks in IPv6. The protocols and messages used during autoconfigurations are described in section 2; section no. 3 presents several security issues related to the autoconfiguration protocols. Section 4 presents a real-life case scenario and the results of field testing.

2 About Automatic Address Configuration in IPv6

The useful advantage of IPv6 protocol, most relevant to the subject of this paper, is allowing network nodes to address themselves on their own or with Neighbor Discovery Protocol is called SLAAC (Stateless address autoconfiguration).

Although there are tasks, for implementation of that greater control is needed besides addressing in LAN networks. At that case it is necessary to use the static addressing or DHCP server for IPv6 protocol (DHCPv6).

As the first step, every host configures link-local address on every IPv6-enabled interface. In early IPv6 documentation, the address should almost everytime be derived from layer 2 address (e.g. MAC address) - now, it is not the case (see [7]), for example Microsoft Windows 7 or 8 configures its host address portion randomly. In any case, the IPv6 host is supposed to perform a DAD (Duplicate Address Detection) operation. Once link-local address is configured, the autoconfiguration operation commences. The entire process is described in [8]. Just to provide short overview, the process relies on:

a) IPv6 router sending out RAs (Router Advertisements) periodically and on-demand (as a response to RS message - Router Solicitation),
b) Hosts sending RS messages and obtaining RA information, such as an address prefix and a lease lifetime.

Fig. 1 provides the lifecycle of an autoconfigured IPv6 address.

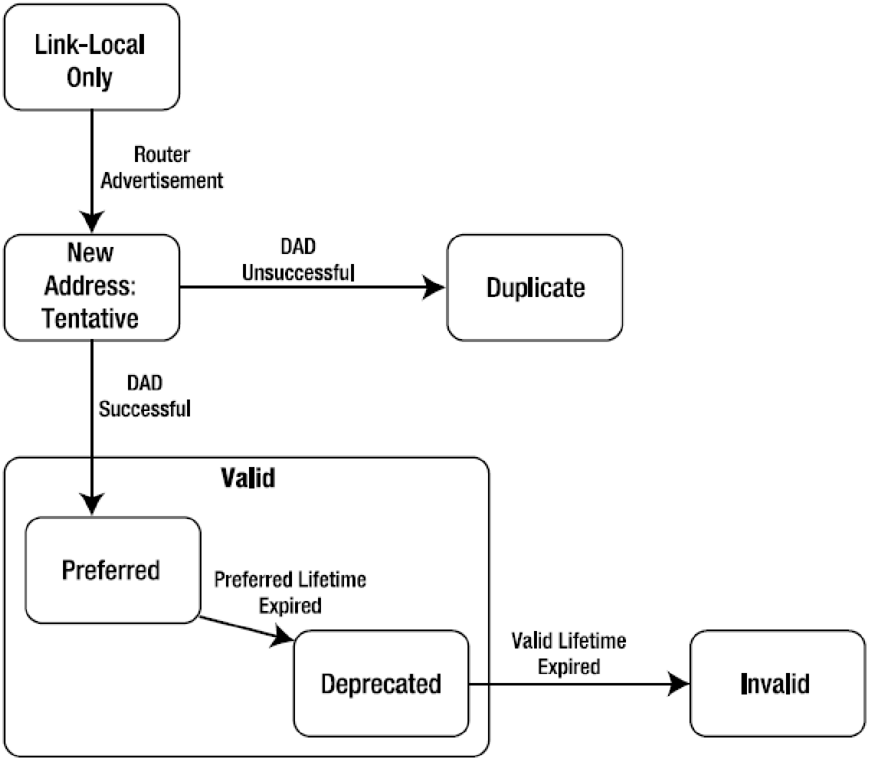


Fig. 1. The state-diagram of the life cycle of an IPv6 address. Source: [3].

In simple terms, stateful configuration utilizes an updated version of DHCP for IPv4. The DHCP protocol for IPv6 (DHCPv6) is slightly different because it relies on the client sending RS messages, thus detecting the presence of the routers on the link. The steps that follow include [9]:

a) If a router is found (RA is received), the RA message is examined for flags indicating whether DHCP can or should be used,
b) If no router is found or DHCP can be used, host sends DHCP Solicit message to All-DHCP-Agents multicast address (this is somewhat deprecated due to lack of default gateway specification, see [10] for details).

The overview of DHCPv6 operation is presented on Fig. 2.

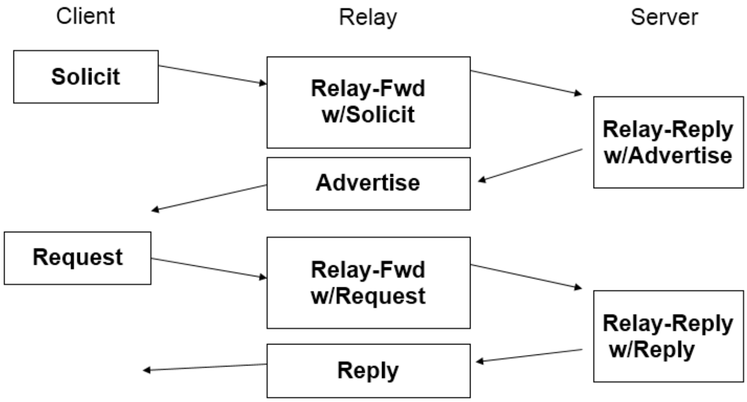


Fig. 2. The state-diagram of DHCPv6 operation. Source: [4].

3 Automatic Configuration Security Issues

Unfortunately, IPv6 configurations may impose a vulnerability threats in many situations, when improperly used. An example of such problems involve using fragmentation for attacks on IPv6 network, such as overlapping fragments, Paxson/Shankar model [11] or Rose attack [12], as well as 6to4 tunneling, quoting [3]: "allowing people to create packets with spoofed IPv6 addresses and encapsulate them in legitimate IPv4 packets, thereby passing anti-spoofing filters that may be in effect". Other examples include using tunneling for attacks, DNS advertising or neighbor discovery. As shown before, the stateless configuration allows any IPv6 device to communicate with other IPv6 devices in the same LAN, by advertising its presence, so it can be located by Neighbor Discovery Protocol. NDP protocol however, cannot be left without supervision; it may allow an attacker to gather network devices data, for example. Further data on this subject is thoroughly described in [13].

3.1 State-of-the-Art

The autoconfiguration mechanisms of an IPv6 protocol are subject to huge research indicating its vulnerabilities, as [14], [7], [15] or [16]. There is also well known literature pointing to techniques of securing network behavior, especially:

- Secure Neighbor Discovery (SEND) [17];
- IPv6 Router Guard [18];
- Autoconfig filtering on Ethernet switches [19];
- Implications for Network Scanning [20].

Such techniques and mechanisms have been recognized as more or less effective, but are also difficult to configure. With autoconfiguration in mind, the common practice (for example when setting up a conference network) is to configure the edge router for IPv6 and let the modern operating systems autoconfigure themselves. The most dangerous fact is, modern operating systems attempt IPv6 autoconfiguration by default and use IPv6 stack with higher priority than IPv4 [21]. This is extremely probable for example if BYOD (Bring Your Own Device) strategy is deployed without sufficient configuration and/or monitoring. The point of this article is to present vulnerability of such scenario and show yet another security flaw - the DHCP 'flapping' on an autoconfigured network.

The malicious behavior scenarios possibilities are discussed in IPv6 related documentation, such as [22] or [17]. Mitigating such risks is by all means possible and doesn't mean that IPv6 features are flawed; the administrator just has got to have knowledge of such threat and be aware of the conditions imposing a vulnerability. Several techniques for securing IPv6 local traffic are considered canonical, such as SEND (SEcure Neighbor Discovery, see [17]), other methods are possible but deployed rarely, e.g. IPv6 Router Advertisement Guard [18]. The problem is, those solutions are, quoting Levy-Abegnoli, 'non-trivial to deploy' ([18]). In case of non-prepared administrators or low budget for security testing, those techniques won't even be considered. This is sadly a typical experience in small and medium enterprises.

In this section, several autoconfiguration security issues for default-configured network devices and hosts are described. An administrator should be able to recognize such conditions in his own network and prepare for those situations accordingly.

3.2 Typical Autocofiguration Scenarios and Associated Threats

Following scenarios are possible if the attacker has access to local network and the IPv6 devices are configured with default security measures. The network doesn't have to be configured for IPv6, all modern devices support IPv6 and the IPv6 stack is turned on by default.

Stateless Autoconfiguration with Rogue Station: This security risk scenario assumes that following conditions are fulfilled:

a) The original network node is autoconfigured using SLAAC mode with address A1,
b) The malicious node is trying to set its address to A1 repeatedly ignoring DAD (Duplicate Address Discovery) messages.

The outcome should be IPv6 stack disabled on the second node. However, because of the malicious nature, should the second node continue to send packets, the original node's operating system operation may vary, depending on its manufacturer and version. This is a situation that should be tested for systems used in organization's network.

Stateless Autoconfiguration with Rogue Router: This security risk scenario assumes that following conditions are fulfilled:

a) The network nodes are using SLAAC autoconfiguration. For this purpose, an original router node is configured to send RAs containing network prefix,
b) Every node is configured with network prefix and default gateway through RA messages,
c) The malicious router is being introduced to the network, sending RAs on his own.

In the outcome, an administrator would want the nodes to ignore other RAs and this is achievable through specific configuration and network security devices introduction; but the question is: how would default configured equipment behave? This is interesting research subject for various operating systems and versions.

Stateful Autoconfiguration with Rogue DHCP Server: Scenario conditions include:

a) Configuring network nodes and DHCPv6 server for a stateful configuration of any device,
b) Introducing a malicious DHCPv6 server, competing with an original one.

Desired network behavior would be ignoring new DHCP server, until lease expiration time. The device should then ask legitimate server for another lease, choosing another DHCP only in the situation of denial or absence of legitimate DHCP server.

Stateful Autoconfiguration with Rogue Router: Scenario conditions include:

a) Configuring network nodes and DHCPv6 server for a stateful configuration of any device,
b) Introducing a malicious IPv6 router, sending RAs on his own.

Desired network behavior would be ignoring malicious router, until lease expiration time. The device should then ask legitimate server for another lease, choosing RA information only in the situation of denial or absence of legitimate DHCP server.

4 Field Study. Rogue SLAAC Server vs. Legitimate DHCP Server Example

For the study of security issues, the last and most interesting scenario was chosen for field testing. The scenario consisted of:

a) Configuring a typical internetwork with DHCP server,

b) Introducing an IPv6 rogue router advertising RAs for stateless autoconfiguration.

The aim of this work was to create an IPv6 network using DHCP addressing mode and revealing the features of the network equipment's work.

4.1 Research Topology

For the project we used the following equipment:

- Routers: Cisco 1841 with IOS operating system,
- Switches: Cisco 2950T with IOS operating system,
- PCs with Microsoft Windows7 Enterprise SP1 operating system,
- PCs with Microsoft Windows Server 2008 R2 Enterprise SP1 operating system.

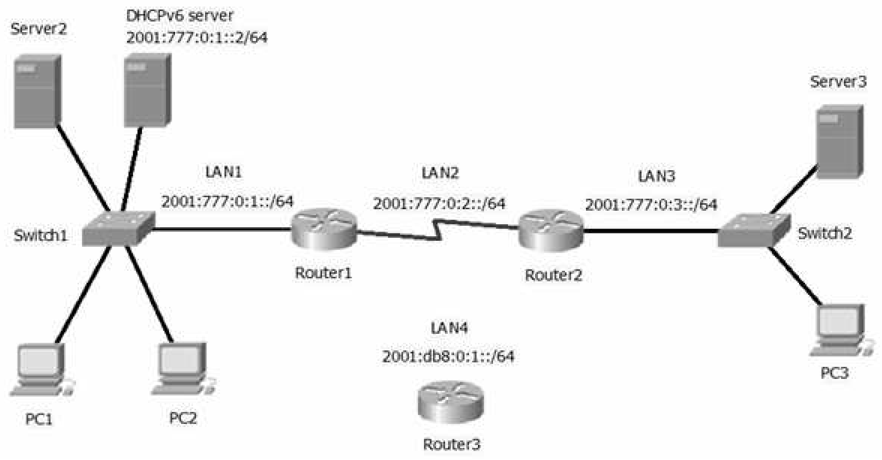


Fig. 3. Test topology. Source: own work.

Schematic representation of the networks and their connection to each other is shown in Figure 3. DHCP server has been deployed on the Windows server 2008 r2 enterprise sp1. In order for a computers to receive addresses from the DHCP server, Router1 and Router2 were configured for the *dhcp-relay* mode with the commands:

```
Router1:
ipv6 dhcp relay destination 2001:777:0:1::2
ipv6 nd managed-config-flag
ipv6 nd prefix 2001:777:0:1::/64 no-advertise
```

```
Router2:
ipv6 dhcp relay destination 2001:777:0:1::2 Se0/0/0
ipv6 nd managed-config-flag
ipv6 nd prefix 2001:777:0:3::/64 no-advertise
```

As a result of configuration, all computers obtained dynamic addresses from DHCP server. DHCP server and routers were assigned static IP addresses. Moreover, the server gave out addresses for both networks LAN1 and LAN2.

4.2 Rogue SLAAC Router Scenario

After verification, it was decided to connect router3 to switch1. Router3 was configured with these commands:

```
ipv6 unicast-routing
ipv6 address 2001:db8:0:1::1/64  (on the interface FastEthernet0/0)
ipv6 enable
```

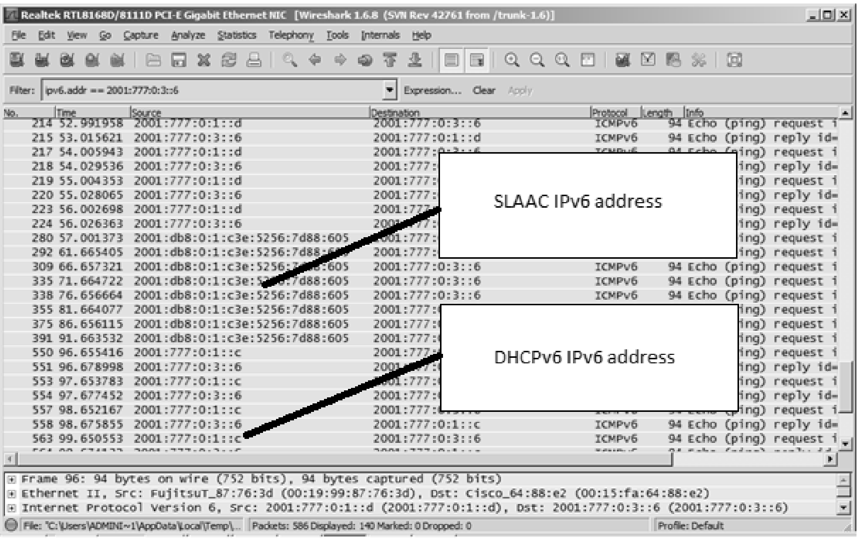
That is, router3 had IPv6 mode enabled and assigned an IPv6 address to the FastEthernet interface. Accordingly, the automatic address assignment SLAAC was activated, which was enabled by default.

Connecting the router3 to switch1 had significant effects on the computers in the network LAN1. First, the computer was running using the address obtained from the DHCP server. Then, after a while, the computer received the SLAAC address from Router3. After a few minutes the computer again received address from a DHCP server, but different from the first one. The address lease time at DHCP server was 8 days by default. On the average, changing of IP address occurred in 6-9 minutes. Fig. 4 shows the result of observation, which was carried out using Wireshark sniffer application.

Fig. 4 shows that the command ”ping 2001:777:0:3::6” was executed on computer with IP address 2001:777:0:1::d. After changing of addressing in SLAAC mode and returning back to the DHCP provisioned address, IP address of the computer became 2001:777:0:1::c.

This change in the DHCP address happened every time, going through all the addresses from scope, thus depleting the available pool. This problem occured on all computers in the network with both Microsoft Windows Server 2008 and Microsoft Windows 7.

A similar experiment was done in the network LAN3 (Router3 was connected to Switch2). The results of computers behavior were the same as in the network LAN1. Routing protocol RIP was used on routers 1 and 2. If RIP protocol was enabled on Router3, then during IP address change, the computer was still able to ping the devices in the other network. If the Router3 was not assigned the static unicast address during configuration phase, then the computers received only link-local addresses from Router3, but everything else followed the same


Realtek RTL8168D/8111D PCI-E Gigabit Ethernet NIC [Wireshark 1.6.8 (SVN Rev 42761 from /trunk-1.6)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Expression... Clear Apply
SLAAC IPv6 address
DHCPv6 IPv6 address

Filter: ipv6.addr == 2001:777:0:3::6

```
No.  Time      Source                              Destination          Protocol Length Info
 214 52.991958 2001:777:0:1::d                     2001:777:0:3::6      ICMPv6     94 Echo (ping) request i
 215 53.015621 2001:777:0:3::6                     2001:777:0:1::d      ICMPv6     94 Echo (ping) reply id=
 217 54.005943 2001:777:0:1::d                     2001:777:                           ing) request i
 218 54.029536 2001:777:0:3::6                     2001:777:                           ing) reply id=
 219 55.004353 2001:777:0:1::d                     2001:777:                           ing) request i
 220 55.028065 2001:777:0:3::6                     2001:777:                           ing) reply id=
 223 56.002698 2001:777:0:1::d                     2001:777:                           ing) request i
 224 56.026363 2001:777:0:3::6                     2001:777:                           ing) reply id=
 280 57.001373 2001:db8:0:1:c3e:5256:7d88:605      2001:777:                           ing) request i
 292 61.665405 2001:db8:0:1:c3e:5256:7d88:605      2001:777:                           ing) request i
 309 66.657321 2001:db8:0:1:c3e:5256:7d88:605      2001:777:0:3::6      ICMPv6     94 Echo (ping) request i
 335 71.664722 2001:db8:0:1:c3e:5256:7d88:605      2001:777:0:3::6      ICMPv6     94 Echo (ping) request i
 338 76.656664 2001:db8:0:1:c3e:5256:7d88:605      2001:777:0:3::6      ICMPv6     94 Echo (ping) request i
 355 81.664077 2001:db8:0:1:c3e:5256:7d88:605      2001:777:                           ing) request i
 375 86.656115 2001:db8:0:1:c3e:5256:7d88:605      2001:777:                           ing) request i
 391 91.663532 2001:db8:0:1:c3e:5256:7d88:605      2001:777:                           ing) request i
 550 96.655416 2001:777:0:1::c                     2001:777:                           ing) request i
 551 96.678998 2001:777:0:3::6                     2001:777:                           ing) reply id=
 553 97.653783 2001:777:0:1::c                     2001:777:                           ing) request i
 554 97.677452 2001:777:0:3::6                     2001:777:                           ing) reply id=
 557 98.652167 2001:777:0:1::c                     2001:777:                           ing) request i
 558 98.675855 2001:777:0:3::6                     2001:777:0:1::c      ICMPv6     94 Echo (ping) reply id=
 563 99.650553 2001:777:0:1::c                     2001:777:0:3::6      ICMPv6     94 Echo (ping) request i
```

```
⊞ Frame 96: 94 bytes on wire (752 bits), 94 bytes captured (752 bits)
⊞ Ethernet II, Src: FujitsuT_87:76:3d (00:19:99:87:76:3d), Dst: Cisco_64:88:e2 (00:15:fa:64:88:e2)
⊞ Internet Protocol Version 6, Src: 2001:777:0:1::d (2001:777:0:1::d), Dst: 2001:777:0:3::6 (2001:777:0:3::6)
File: "C:\Users\ADMINI~1\AppData\Local\Temp\...   Packets: 586 Displayed: 140 Marked: 0 Dropped: 0   Profile: Default
```

Fig. 4. Traffic during experiment - the 'flapping' between stateless and stateful address. Source: own work.

erratic scenario. Rogue router's RAs may result in change of the default route on affected hosts - this is another vulnerability as described in [14].

5 Conclusion

In this article, given security risk scenarios are described as potentially dangerous; field testing confirming the most interesting scenario was presented as case study. As a result of this work it was shown that two autoconfiguration methods (one of which is the stateful mode, and the other is a SLAAC mode) would work in IPv6 network with the same priority and hinder the work of each other. This in turn can lead to serious disturbances in the functioning of DHCP servers - the basic service of corporate networks.

This article shows one of many vulnerabilities of an autoconfigured IPv6 network. Important fact is, given latest BYOD strategies, such autoconfiguration may occur without administrator's knowledge - modern operating systems support IPv6 by default. This is not indication of a flaw - just a reminder, that in IPv6 networks, careful administration of autoconfiguration must (!) be applied, for example using SEND [17].

Acknowledgement. Presented work is a part of ongoing research "IPv6 role in High Availability and Redundant Networks of Small and Medium Organisations", being conducted in Academy of Business in Dabrowa Gornicza.

References

1. Ashton, K.: That 'Internet of Things' Thing. RFID Journal 22 (July 2009)
2. Curtis, S., Niedzielewski, D.: Internet of Things: miliardy urządzeń, czujników i liczników podłączonych do sieci. Networld polish ed. 01/2013, Miller Druk, Warszawa (2013)
3. Van Beijnum, I.: Running IPv6. Apress, New York (2006) ISBN: 1-59059-527-0
4. Odom, W.: CCNP Route 642-902 Official Certification Guide, 4th edn. Cisco Press, Indianapolis (2011)
5. Deering, S., Hinden, R.: Internet Protocol, Version 6 (IPv6) Specification, RFC 2460, IETF 1998 (1998)
6. Burkhalter, M.: Study: IPv6 adoption remaining slow, Perle Industry News (2011), http://www.perle.com/articles/Study-IPv6-adoption-remaining-slow-800490443.shtml
7. Narten, T., et al.: Privacy Extensions for Stateless Address Autoconfiguration in IPV6 (RFC 4941), IETF 2007 (2007)
8. Thomson, S., Narten, T., Jinmei, T.: IPv6 Stateless Address Autoconfiguration (RFC 4862), Draft Standard, IETF 2007 (2007)
9. Droms, R., et al.: Dynamic Host Configuration Protocol for IPv6 (DHCPv6) (RFC 3315), IETF 2003 (2003)
10. Droms, R., Narten, T.: Default Router and Prefix Advertisement Options for DHCPv6, IETF 2009 (2009), http://tools.ietf.org/html/draft-droms-dhc-dhcpv6-default-router-00
11. Novak, J.: Target-Based Fragmentation Reassembly, Sourcefire, Columbia, MD 2005 (2005)
12. Hollis, K.: Rose Attack Explained, http://digital.net/~gandalf/Rose_Frag_Attack_Explained.htm (retrieved: February 16, 2013)
13. Narten, T., et al.: Neighbor Discovery for IP version 6 (IPv6) (RFC 4861), Draft Standard, IETF 2007 (2007)
14. Chown, T., Venaas, S.: IPv6 Router Advertisement Problem Statement (RFC 6104), IETF 2011 (2011)
15. Chown, T.: Dynamic Host Configuration Protocol (DHCP): IPv4 and IPv6 Dual-Stack Issues (RFC 4477), IETF 2006 (2006)
16. Durand, A. et al.: Operational Considerations and Issues with IPv6 DNS (RFC 4472), IETF 2006 (2006)
17. Arkko, J. (ed.): SEcure Neighbor Discovery (SEND) (RFC 3971), IETF 2005 (2005)
18. Levy-Abegnoli, E., et al.: IPv6 Router Advertisement Guard (RFC 6105), IETF 2011 (2011)
19. Ward, N.: IPv6 Autoconfig Filtering on Ethernet Switches, Internet Draft, IETF 2009 (2009)
20. Chown, T.: Implications for Network Scanning (RFC 5157), IETF 2008 (2008)
21. Chown, T.: Default Address Selection for Internet Protocol Version 6 (IPv6) (RFC 6724), IETF 2012 (2012)
22. Nikander, P. (ed.): IPv6 Neighbor Discovery (ND) Trust Models and Threats, RFC 3756, IETF 2004 (2004)
23. Mankin, A.: Threat Models introduced by Mobile IPv6 and Requirements for Security in Mobile IPv6, Internet Draft, IETF 2002 (2002)

Security Aspects of Virtualization in Cloud Computing

Muhammad Kazim, Rahat Masood, Muhammad Awais Shibli,
and Abdul Ghafoor Abbasi

National University of Sciences and Technology,
Sector H-12, Islamabad - 44000, Pakistan
{muhammad.kazim,10msccsmmasood,awais.shibli,abdul.ghafoor}@seecs.edu.pk

Abstract. In Cloud computing, virtualization is the basis of delivering Infrastructure as a Service (IaaS) that separates data, network, applications and machines from hardware constraints. Although Cloud computing has been a focused area of research in the last decade, research on Cloud virtualization security has not been extensive. In this paper, different aspects of Cloud virtualization security have been explored. Specifically, we have identified: i) security requirements for virtualization in Cloud computing which can be used as a step towards securing virtual infrastructure of Cloud, ii) attacks that can be launched on Cloud virtual infrastructure, and iii) security solutions to secure the virtualization environment by overcoming the possible threats and attacks.

Keywords: Cloud computing, Cloud virtualization security, Cloud service provider, Hypervisor, Virtual machines, Disk images.

1 Introduction

Cloud computing is becoming popular among IT businesses due to its agile, flexible and cost effective services being offered at Software, Platform and Infrastructure level. Software as a Service (SaaS) allows users to access applications hosted by different vendors on Cloud via internet. Platform as a Service (PaaS) enables developers to code, test and deploy their applications on IaaS. In Infrastructure as a Service (IaaS) model, Cloud providers offer services such as computing, network, storage and databases via internet. IaaS is the base of all Cloud services with PaaS and SaaS both built upon it. The primary features of IaaS are elasticity and virtualization [1].

Virtualization enables a single system to concurrently run multiple isolated virtual machines (VMs), operating systems or multiple instances of a single operating system (OS). However, there are still open challenges in achieving security for Cloud virtualization. Research has been done to explore major security issues related to virtualization in Cloud. The standard bodies in computing security including National Institute of Standard Technologies (NIST) [2], Cloud Security Alliance (CSA) [3], and Payment Card Industry Data Security Standard (PCI DSS) [4] have issued guidelines on virtualization technologies. These guidelines

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 229–240, 2013.

discuss security issues related to virtualization in Cloud and provide recommendations for secure virtualization environments. However, the holistic view of virtualization security has not been presented in a composed form. Furthermore, there is need to investigate existing virtualization security solutions proposed in literature to mitigate different attacks.

This paper analyzes the security issues of Cloud virtualization from three different aspects including the security requirements, attacks and security solutions of virtualization. Therefore, the contribution of this paper is three-fold. This paper: i) presents general requirements for securing Cloud virtualization environment, ii) describes possible attacks that can be launched on different virtualization components (hypervisor, VMs, images), and iii) describes solutions and architectures to provide protection against these different attacks that lead towards a secure virtualization environment.

This paper is organized as follows: Section 2 reviews the security requirements that must be followed to secure virtualization environment. Section 3 describes the major threats and attacks scenarios related to Cloud, section 4 provides existing security solutions for virtualization. Conclusion is given in section 5.

2 Security Requirements of Virtualization

Different virtualization approaches can be applied to various system layers including hardware, desktop, operating system, software, memory, storage, data and network. Full virtualization is a form of hardware virtualization that involves complete abstraction of underlying hardware and provides better operational efficiency by putting more work load on each physical system [2]. Full virtualization can be categorized into two forms: i) bare metal virtualization and ii) hosted virtualization. Bare metal approach is mostly used for server virtualization in large computing systems like Cloud computing as it provides better performance, more robustness and agility. The architecture of bare metal based virtualization generally used in Cloud is shown in Fig. 1.

The unique characteristics of virtualization along with their benefits also have some drawbacks. Each component of virtualization needs to be secured from the

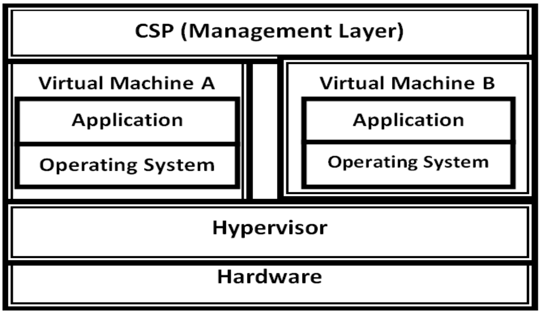


Fig. 1. Bare metal virtualization architecture

possible threats. In general, before planning and implementing security of any system it is important to understand the security requirements of that environment. This section presents general requirements to prevent virtualization layers attacks in Cloud.

2.1 Service Provider Requirements

A report by Alert Logic [5] shows that 50 percent of Cloud users consider service provider security as a major threat. However, the impact of Cloud service provider on Cloud virtualization security has also not been discussed comprehensively in literature.

To secure the virtualization hardware, (Cloud) service provider must limit access of hardware resources to authorized person. Similarly, proper access control should be implemented in the management layer, so that each administrator has access only to its concerned data and software. The service provider also need to provide strong authentication mechanisms to users. Furthermore, security principles for the development of trusted computing system such as economy of mechanism, complete mediation, open design, principle of least privilege, psychological acceptability must also be followed by the service provider.

2.2 Hypervisor Requirements

Hypervisor provides the necessary resource management functions that enable sharing of hardware resources between the VMs. Hypervisor must maintain the isolation between VMs and support multiplexing of multiple VMs on single hardware platform [6]. It must ensure that no application from any VM can directly take control of it as a host to modify the source code of hypervisor and other VMs in the network. Hypervisor should also monitor the guest OS and applications in VMs to detect any suspicious behavior [7].

Programs that control the hypervisor must be secured using similar practices used for security of programs running on servers. Similarly access to the hypervisor must be restricted. Other security measures to secure hypervisor include installing updates to the hypervisor, restricting administrator access to the hypervisors management interfaces and analyzing hypervisors logs to see if it is functioning properly [2].

2.3 Virtual Machine Requirements

Limit on VM resource usage has to be assigned so that malicious VMs can be restricted from consuming extra resources of the system [4]. Moreover, isolation between virtual machines should be provided to ensure that they run independently from each other. To secure the guest OS running in virtual machines, best practices for the security of physical machines must be followed that include updating the OS regularly for patches and updates, using anti-virus software, securing internet and email and monitoring of guest OS regularly [3].

Privileged VM (Dom0) is the first domain started by XEN hypervisor after boot. It is responsible for monitoring the communication between the remote users and guest VMs. Dom0 is also responsible for creating and destroying all guest VMs and providing device drivers to the guest VMs. Dom0 should boot the guest VMs without tampering them. The state of the VM saved as a disk file in Dom0 must remain confidential, and it must not be tampered [8].

2.4 Guest Image Requirements

Hypervisors use disk images (host files used as disk drive for guest OSs) to present guest OSs with virtual hard drives. Guest OS images can be moved and distributed easily, so they must be protected from unauthorized access, tampering and storage. To securely manage the guest OS images they must be examined and updated regularly according to the requirements. Unnecessary images must not be created and if any image is useless it must be removed from system [2]. Whenever VM is migrated from one physical machine to another, images on previous disks should be completely removed. Similarly, data on old broken disks should also be removed before they are discarded. Furthermore, backup of the virtual machines images must be maintained.

VM checkpoint is a feature that allows the users to take snapshot of VM image in the persistent storage. Snapshot records the state of the running image that contains all components of the guest OS. Snapshot is generally captured as a difference between the image and the running state. The major function of checkpoint is to restore VM to its previous state if the VM enters any undesired state. However, the snapshot access should be given to authorized users and checkpoint must be used only to return VM to a stable and non-malicious state [9].

3 Attacks on Virtualization

Each component of virtualization layer can act as an attack vector to launch multiple attacks on the system. Attacks that target different components of virtualization environment may result in security issues such as compromise of complete Cloud infrastructure, stealing of customer data and system hacking. This section discusses different attack scenarios at virtualization environment in Cloud.

3.1 Service Provider Attacks

If the attacker has physical access to the Cloud hardware, he may run malicious application or code in the system to damage the VMs by modifying their source code and changing their functionality. With the help of physical access to system, attackers can also launch cross VM side channel attacks. These attacks include CPU cache leakage to measure the load of other virtual web server on the network [10]. Moreover, if access control is not implemented properly, different administrators such as network admin and virtualization admin might access the customer data that they are not authorized to access. These activities will result in security compromises such as loss of data confidentiality and unauthorized traffic monitoring.

Service provider has to ensure that software deployed on Cloud are built using proper coding practices. Flawed coding can result in web application attacks such as SQL Injection, Cross Site Scripting, Denial of Service and Code Execution etc. Alert Logic [5] report shows web application attacks to be the most common attacks on Cloud environment, impacting almost 52 percent customers.

3.2 Hypervisor Attacks

A Cloud customer can lease a guest VM to install a malicious guest OS, which attacks and compromises the hypervisor by changing its source code in order to gain access to the memory contents (data and code) of VMs present in the system [7]. With more features in hypervisor its increased code size has resulted in design and implementation vulnerabilities. To control the complete virtualization environment malicious hypervisors such as BLUEPILL rootkit, Vitriol and SubVir and are installed on the fly, which give attacker the host privileges to modify and control VMs [11]. This technique used by malicious software to take complete control of the underlying operating system by hiding itself from administrator and security software is called hyperjacking.

Another attack in which program running in one VM can get root access to the host machine is called VM Escape [2]. It is done by crashing the guest OS to get out of it and running an arbitrary code on the host OS. Therefore, such malicious VMs can take complete control of the host OS. Escaping the guest OS allows the VMs to interact with the hypervisor and provides them access to other guest OS on the system as well. Fig. 2 shows that the attacker from his virtual machine (VM 2) is able to escape his VM. VM 2 is used to compromise the hypervisor which is further used to launch attacks on other VMs (VM 1) in the system.

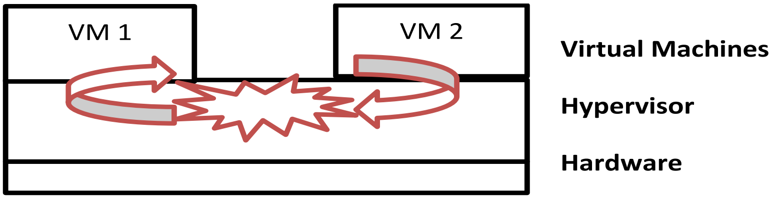


Fig. 2. VM Escape attack (Source: [7])

3.3 Virtual Machine Attacks

Malicious programs in different virtual machines can achieve required access permissions to log keystrokes and screen updates across virtual terminals [12] that can be exploited by attackers to gain sensitive information. If isolation is not properly implemented covert channels can be used for unauthorized communication with other VMs in the system. Attackers can use Trojans, malwares and botnets for traffic monitoring, stealing critical data, and tampering the functionality of guest OS. Conficker, Zeus botnet, command and control botnet communication activity are the examples of such attacks that result in data destruction, information gathering and creation of backdoors for attackers. Attacks through buggy software, viruses and worms can exploit the guest OS in VMs. Furthermore, unpatched VM operating systems can be exploited by zero day attacks.

The privileged host virtual machine Dom0 can be compromised by attacker to either tamper boot process of guest VMs or access all guest VMs including their memory, disk space and network traffic. By controlling Dom0 attacker can create too many virtual machines to consume all resources of the system or destroy any virtual machine containing important data by launching DOS attack at Cloud. Furthermore, the saved state of guest virtual machine as a disk file appears in plaintext to Dom0. Attacker can compromise the integrity and confidentiality of the saved VM state and when restored VM may not function as desired [8].

3.4 Guest Image Attacks

Unnecessary guest OS images in Cloud can result in different security issues if the security of each image is not maintained [2]. If a malicious guest OS image is migrated to another host, it can compromise the other system as well. Furthermore, creating too many images and keeping unnecessary images can consume resources of the system which can be used as a potential attack vector by attacker to compromise the system [2]. When VMs are moved from one physical machine to other, data of VM images might still exist on previous storage disks that attacker can access. Similarly, attackers might also recover some data from old broken disks [3]. The security of image backup is also an issue. By gaining access to the backup images attacker can extract all information and data.

Attacker can access VM checkpoint present in the disk that contain VM physical memory contents and can expose sensitive information of VM state. A new checkpoint can be created by attacker and loaded in system to take VM to any state desired by attacker. If all the checkpoints in storage are accessed, information about previous VM states can be obtained [9].

4 Security Solutions for Virtualization

To cater the attacks on virtualization environment different security solutions have been proposed in literature. This section discusses those security solutions for each component of virtualization architecture. By implementing these security solutions the attacks discussed in section 3 can be mitigated or at least the impact of those attacks on virtualization environment can be minimized.

4.1 Service Provider Security

Unauthorized person should not have physical access to the virtualization hardware of the system. In order to protect VMs from unauthorized access by Cloud administrators, each VM can be assigned access control that can only be set through Hypervisor. The three core principles of access control namely identification, authentication and authorization will restrict admin access from unauthorized data and system components. Moreover, if any administrator is involved in security compromise, access control implemented in Cloud can help identify that person. Web application attacks can be prevented by installing an application layer firewall infront of web facing applications and by having the customer code reviewed for common vulnerabilities [4].

An online identity management community OpenID has been integrated with an open source Cloud platform OpenStack to provide identity management in Cloud [13]. Sandra R. et al. [14] proposed an architecture using SELinux, XEN, IPsec as tools to enforce Mandatory Access Control (MAC) policies at VM, OS and network layers. These MAC policies control the communication between VMs based on application templates that can be configured by administrators dynamically. Furthermore, the security requirements of virtualized environment differ from that of physical system, Cloud service provider must make sure that the security tools for vulnerability assessment also include the virtualization tools used [3].

4.2 Hypervisor Security

Hypersafe is a system that maintains code integrity of the Hypervisor. It extends the hypervisor implementation and prevents its code modification by locking down the write-protected memory pages. It secures the Hypervisor against the control-flow hijacking attacks by protecting its code from unauthorized access [15]. VM Escape attack can only be executed through a local physical environment. Therefore, the physical Cloud environment must be prevented from insider attacks. The interaction between guest machines and host OS must also be properly configured [12].

In order to stop one VM from affecting or communicating with other VMs isolation must be properly implemented and maintained by hypervisor. Moreover, further possible attack vectors on hypervisors can be reduced by hardening the hypervisor [4]. These techniques include separating the duties of administrative functions, restricting the hypervisors administrator access to modify, create or delete hypervisor logs, and monitoring the hypervisor logs regularly.

4.3 Virtual Machine Security

Administrator must deploy a software or application that stops VMs from using extra resources unless authorized. Moreover, a light weight process must run on a virtual machine that collects logs from the VMs and monitors them in real time to fix any tampering of VMs. The guest OS and applications running on it must be hardened by using best security practices. These practices include installing security software such as anti-viruses, anti-spyware, firewall, Host Intrusion Prevention System (HIPS), web application protection, and log monitoring in guest OS [4]. Protection of VMs by different security practices is shown in Fig. 3.

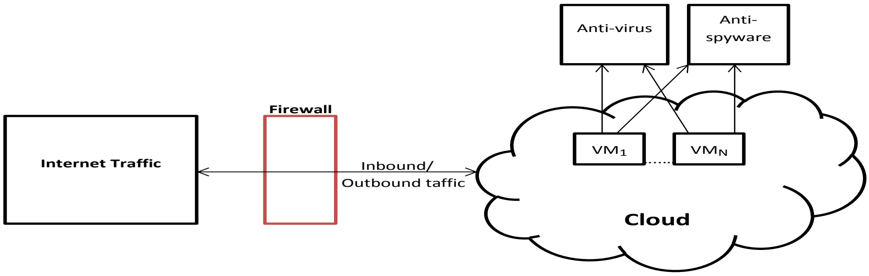


Fig. 3. VM security by firewall, anti-virus and anti-spyware

To identify the faults in guest OS Dan P. et al. [16] proposed a system called "Vigilant". It utilizes virtualization and machine learning methods to monitor VMs through hypervisor without putting any monitoring agent in VMs (out-of-band detection). Flavio L. et al. [17] proposed Advanced Cloud Protection System (ACPS) that monitors and protects the integrity of OS in guest VMs. The periodic monitoring of executable system files is done to check the behavior of Cloud components. It uses virtual introspection techniques to deploy guest monitoring machine in system without being noticed by attacker on guest VM. Hence any suspicious activity on the guest OS can be blocked.

To protect the newly created virtual machines for users (guest VMs) from compromised privileged virtual machine Dom0, a protocol is designed by Jinzhu Kong [8]. Hypervisor generates a pair of secret keys, Kernel and the initrd image are kept encrypted all the time with the secret key Kimg. First the user attests the Cloud server through Trusted Platform Module (TPM), if attestation succeeds then user sends a boot request to the Dom0 which then boots the guest domain. The guest VM executes the wrapping code and requests Hypervisor to decrypt kernel and initrd images. Hypervisor encrypts this request with its private key and asks user for key to decrypt kernel so that a VM can be created. The user sends private key Kimg encrypted under the public key of Hypervisor. Hypervisor decrypts the user message, and the private key Kimg is used to decrypt the kernel, initrd images and to launch the guest virtual machine. In this way the newly created VM is secured from compromised Dom0. The complete workflow is shown in Fig. 4.

To avoid the VM storage attacks, before saving the state of the virtual machine in Dom0 its encryption can be done using AES-256, where key can be any random initialization vector. The hash of the encrypted state can be taken using MD5. When the virtual machines are to be restored, the new hash can be taken to verify the integrity of saved virtual machine. If the hash of the restored state

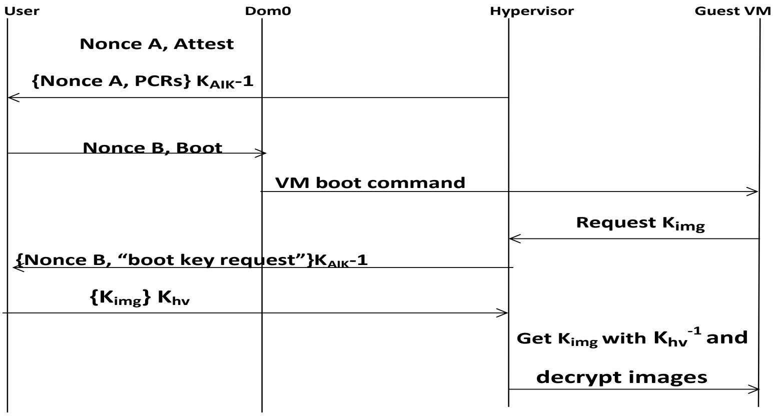


Fig. 4. Secure VM boot protocol (Source: [8])

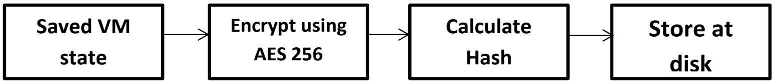


Fig. 5. Securing the saved VM state

and hash of the saved state match it means that the virtual machine state is not altered [8]. Fig. 5 shows the secure storage of saved VM state.

4.4 Guest Image Security

Organizations using virtualization must have a policy to manage the creation, usage, storage and deletion of images. Image files must be scanned for the detecting viruses, worms, spyware and rootkits that hide themselves from security software running in guest OS. J. Wei et al. [18] proposed an image management system to efficiently manage images in Cloud and detect security violations in images. It proposes the use of filters, virus scanners and rootkit detectors to provide protection against potentially compromised images. Nuwa [19] is a tool designed to apply efficient patching to VM images in Cloud. By analyzing patches, Nuwa rewrites the patching scripts so that they can be applied offline. As a result, the installation scripts for online patching can be applied to images when they are offline.

When VMs are to be migrated from one physical machine to another, Cloud admin must recheck and ensure that all data is removed from previous or broken disks. To protect the backup VM images cryptographic techniques such as encryption may be employed to encrypt all backup images. If any VM is deleted then its backup must also be removed from system. Furthermore, to protect VM images from storage attacks, Cloud provider must encrypt the complete VM images when not in use [3].

Checkpoint attacks can be prevented by encrypting the checkpoint files. Another mechanism to provide security to Checkpoints is SPARC. SPARC is a mechanism designed to deal with security and privacy issues resulting from VM checkpoint. SPARC enables users to select applications that they want to checkpoint so sensitive applications and processes cant be checkpointed. Table 1 shows the summary of different security aspects of virtualization discussed in the paper.

Table 1. Summary of security aspects of virtualization described in paper

Category	Requirements	Attacks	Solutions
Service Provider	Limit access to hardware	Malicious code execution	Develop and implement policy to limit access to hardware
	Implement access control	Stealing of customer data through unapproved access	Implement MAC policies at VM, OS and network layers
	Provide strong authentication mechanisms to users	Unauthorized access to Cloud system and data	OpenID integration with OpenStack Cloud to provide secure authentication
Hypervisor	Maintain isolation between VMs	VM Escape attack	Properly configure the interaction between guest machines and host VM
	Hypervisor should monitor functionality of guest VMs	Customers can lease a guest VM to install a malicious guest OS	Encrypt the VMs to protect them from compromised hypervisor and VMs
	Programs controlling the hypervisor must be secured using best software security practices	Malicious hypervisors attacks including BLUEPILL, Vitriol and SubVir	Hypersafe is a system designed to maintain the integrity of Hypervisor
			Use techniques to harden the hypervisor security
Virtual Machines	There must be limit on VMs resource usage	Using a malicious VM to consume extra resources of the system, resulting in DOS attack	Administrator must deploy a software or application that limits VMs from using extra resources unless authorized
	Isolation between virtual machines should be implemented properly	Malicious programs use covert channels to communicate with other VMs in unauthorized way	Vigilant can monitor faults in guest OS of VM
	Update the OS regularly and use anti-virus software, secure internet and restrict remote access	Malicious programs can monitor traffic, steal critical data, and tampering the functionality of VMs	Security features such as firewall, HIPS, log monitoring must be provided in guest OS
	Guest OS must be monitored regularly for updates and errors	Attacks through worms, viruses, botnets can also be used to exploit the VMs	Use anti-viruses, anti-spyware programs in guest OS to detect any suspicious activity
			Advanced Cloud Protection System (ACPS) can monitor and protect the integrity of guest OS
	Securely boot the guest VMs	Attacker can tamper boot process of guest VMs	Security protocol by J. Kong can be to ensure secure boot of guest VMs
	Saved VM state must not be tampered by Dom0	Attacker can compromise the integrity and confidentiality of the saved state of guest virtual machine	Use encryption and hashing of VMs state before saving VM
Guest Images	Snapshot access must be prevented from authorized access	VM checkpoint attacks	Checkpoint attacks can be prevented by encrypting the checkpoints or using SPARC
	Make a policy to remove unnecessary images	Security issues from unnecessary images can compromise system	J. Wei et al. proposed an image management system to manage images in Cloud
	Apply updates and patches to maintain images secure	Old images are vulnerable to zero day attacks	Nuwa is a tool designed to apply efficient patching to VM images in Cloud
	There must be policy to remove images from old disks after VM migration	Attackers can access and recover data from old and broken disks	After VM migration, Cloud admin must ensure that data is removed from old disks
	Backup of the virtual machines images must be maintained	Unauthorized access to the backup data can result in leakage of sensitive information	Backup of VM images must be encrypted. If any VM is removed then its backup must also be removed

5 Conclusion

The security of cloud cannot be maintained unless its virtualization environment is secured. Although different virtualization approaches exist, bare metal virtualization approach is commonly used in large computing systems such as Cloud for server virtualization. This paper presents general architecture of bare metal virtualization and covers security aspects of its different components. Cloud virtualization environment can be compromised by different attacks at service provider, hypervisor, virtual machines, guest operating system and disk images. The attack scenarios at these components are discussed in the paper. To provide security to the virtualization environment, general requirements for virtualization security and different existing security schemes that provide security to virtualization environment have also been discussed. Therefore, the holistic picture of virtualization security in Cloud is provided through structured analysis in which security requirements, attacks and solutions correspond to each other.

Addressing these security aspects will lead towards more extensive research on secure Cloud virtualization environment. In future, an assessment criteria needs to be proposed by which we can analyze the effectiveness of security solutions of virtualization against the specific attacks.

References

1. Orlando, D.: Cloud computing service models, `http://www.ibm.com/developerworks/cloud/library/cl-cloudservices1iaas/cl-cloudservices1iaas-pdf.pdf` (last accessed: October 27, 2012)
2. Hoffman, P., Scarfone, K., Souppaya, M.: Guide to security for full virtualization technologies. National Institute of Standards and Technology (NIST), 800–125 (2011)
3. Brunette, G., Mogull, R., et al.: Security guidance for critical areas of focus in cloud computing v2.1. Cloud Security Alliance, 1–76 (2009)
4. Virtualization Special Interest Group PCI Security Standards Council: Pci dss virtualization guidelines v2.0, pp. 1–39 (2011)
5. ALERTLOGIC: State of cloud security report: Targeted attacks and real world hacks, `http://www.alertlogic.com/resources/cloud-security-report/` (last accessed: April 14, 2013)
6. Szefer, J., Keller, E., Lee, R.B., Rexford, J.: Eliminating the hypervisor attack surface for a more secure cloud. In: Proceedings of the 18th ACM Conference on Computer and Communications Security, pp. 401–412. ACM (2011)
7. Szefer, J., Lee, R.B.: A case for hardware protection of guest vms from compromised hypervisors in cloud computing. In: 2011 31st International Conference on Distributed Computing Systems Workshops (ICDCSW), pp. 248–252. IEEE (2011)
8. Kong, J.: Protecting the confidentiality of virtual machines against untrusted host. In: 2010 International Symposium on Intelligence Information Processing and Trusted Computing (IPTC), pp. 364–368. IEEE (2010)
9. Gofman, M.I., Luo, R., Yang, P., Gopalan, K.: Sparc: a security and privacy aware virtual machinecheckpointing mechanism. In: Proceedings of the 10th Annual ACM Workshop on Privacy in the Electronic Society, pp. 115–124. ACM (2011)
10. Jin, S., Ahn, J., Cha, S., Huh, J.: Architectural support for secure virtualization under a vulnerable hypervisor. In: Proceedings of the 44th Annual IEEE/ACM International Symposium on Microarchitecture, pp. 272–283. ACM (2011)

11. Ibrahim, A.S., Hamlyn-harris, J.H., Grundy, J.: Emerging security challenges of cloud virtual infrastructure (2010)
12. Reuben, J.S.: A survey on virtual machine security. Helsinki University of Technology (2007)
13. Khan, R.H., Ylitalo, J., Ahmed, A.S.: Openid authentication as a service in openstack. In: 2011 7th International Conference on Information Assurance and Security (IAS), pp. 372–377. IEEE (2011)
14. Rueda, S., Sreenivasan, Y., Jaeger, T.: Flexible security configuration for virtual machines. In: Proceedings of the 2nd ACM Workshop on Computer Security Architectures, pp. 35–44. ACM (2008)
15. Wang, Z., Jiang, X.: Hypersafe: A lightweight approach to provide lifetime hypervisor control-flow integrity. In: 2010 IEEE Symposium on Security and Privacy (SP), pp. 380–395. IEEE (2010)
16. Pelleg, D., Ben-Yehuda, M., Harper, R., Spainhower, L., Adeshiyan, T.: Vigilant–out-of-band detection of failures in virtual machines. Operating Systems Review 42(1), 26 (2008)
17. Lombardi, F., Di Pietro, R.: Secure virtualization for cloud computing. Journal of Network and Computer Applications 34(4), 1113–1122 (2011)
18. Wei, J., Zhang, X., Ammons, G., Bala, V., Ning, P.: Managing security of virtual machine images in a cloud environment. In: Proceedings of the 2009 ACM Workshop on Cloud Computing Security, pp. 91–96. ACM (2009)
19. Zhou, W., Ning, P., Zhang, X., Ammons, G., Wang, R., Bala, V.: Always up-to-date: scalable offline patching of vm images in a compute cloud. In: Proceedings of the 26th Annual Computer Security Applications Conference, pp. 377–386. ACM (2010)

Threshold Method of Detecting Long-Time TPM Synchronization

Michał Dolecki[1] and Ryszard Kozera[1,2]

[1] The John Paul II Catholic University of Lublin
ul. Konstantynów 1 H; 20-708 Lublin, Poland
michal.dolecki@kul.pl
[2] Warsaw University of Life Sciences – SGGW
ul. Nowoursynowska 159, 02-776 Warsaw, Poland
ryszard.kozera@gmail.com, ryszard_kozera@sggw.pl

Abstract. The phenomenon of neural networks synchronization by mutual learning can be used to construct key exchange protocol on an open channel. For security of this protocol it is important to minimize knowledge about synchronizing networks available to the potential attacker. The method presented herein permits evaluating the level of synchronization before it terminates. Subsequently, this research enables to assess the synchronizations, which are likely to be considered as long-time synchronizations. Once that occurs, it is preferable to launch another synchronization with the new selected weights as there is a high probability (as previously shown) that a new synchronization belongs to the short one.

Keywords: neural networks, Tree Parity Machine, key exchange protocol.

1 Introduction

One of the most important aspects of information security is ensuring the confidentiality of communications. Two parties wish to exchange certain information in such a way, that unauthorized persons have no opportunity to guess the content of the communication. Confidentiality is implemented mainly through data encryption with usage of various cryptographic algorithms [1], [2]. The cryptographic keys are additional input data used in the algorithms to encrypt and decrypt transmitted messages. Due to the keys used, the cryptography can be classified into two types, i.e. symmetric and asymmetric cryptography [1]. Asymmetric cryptography requires from both parties to maintain two keys, one public and the second a private one. Public key is used to encrypt messages, and private one, to decrypt them. Symmetric cryptography uses the same key to both of these operations. In this scheme the security of key distribution and its storage becomes a vital issue. This leads to a paradox: on one hand if one strives to build a secure communication channel, one must have a secure key distribution channel first. On the other hand, if one has secure channel, then why not use it to the communication. To cope with this problem, the key exchange protocol exploiting an open channel can be used.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 241–252, 2013.

One of the most popular algorithms used for such key agreement is the Diffie-Hellman algorithm [1], [2], which is based on computationally difficult problem of calculating discrete logarithms in cyclic groups. Such kind of problems cannot be solved in a reasonable amount of time [3]. More specifically there is no proposed algorithm to solve these problems which operates in polynomial time but there may be an algorithm with greater computational complexity. Results presented by three physicists: E. Kanter, W. Kinzel and I. Kanter [4] permit constructing the relatively secure [5-8] key exchange protocol using the phenomenon of artificial neural networks' synchronization by mutual learning.

The idea of applying neural networks in cryptology has emerged relatively recently [9], [10]. This approach represents an interesting alternative to the currently used algorithms based on number theory [11]. In the classical setting, artificial neural networks are built of interconnected layers of neurons. The input network's layer receives input signals and sends them in turn to the first hidden layer's neurons. The output values of neurons of one layer form the inputs for the next neurons' layer etc. Finally, the output of the last layer is the output of the entire network. Each impulse sent to the single neuron is modified by a certain weight. For each neuron, if the sum of weighted impulses exceeds a given threshold level, then the neuron transmits a signal (due to the specifically tuned up activation function). Network classifiers [12-14] receive impulses which belong to one of several selected classes. The main characteristic feature of neural networks is their ability to learn from the presented examples. Such learning process consists of modifying the weights of the neurons to establish the most accurate classification of input impulses. The network is considered as a trained one, if the amount of misclassified input vectors is lower than the prescribed a priori specific acceptance level. For weights, that are real numbers, there are infinitely many different values ensuring a proper output of the given network. The specific and proper use of the network, in practice requires the additional conditions imposed on their structure (i.e. its topology) and neuron activation functions [4-7], [11], [15].

In this paper we present the idea of key exchange protocol using neural networks, where the corresponding weights upon pertinent iterative procedure are set to be equal. We discuss first two methods measuring an overlap of both weight vectors by calculating first the cosine of the angle between them and then alternatively by using the Euclidian metric. Evidently, two approaches mentioned above rely on the knowledge of both networks weights' values which need to be sent via open channel. This, however is impossible in practical key exchange protocol as it would result in security breach. In order to make the weights publicly unknown we analyze the corresponding frequencies of publicly known common outputs of both networks (called here TPMs). Results presented in this paper indicate strong correlation between calculated frequencies and other methods for evaluating TPM's synchronization state. The latter lays foundation to formulate condition for classification of synchronization duration for a given time classes, especially to detect at early stage a long-term synchronization.

2 Tree Parity Machine Synchronization

The mentioned above, proposed by three physicists Kanter, Kinzel and Kanter cryptographic key exchange protocol in an open communication channel which resorts to the so-called Tree Parity Machine network (abbreviated here to TPM). The key exchange procedure abides to the following pattern: two sides of the communication i.e. A and B entities create a special TPM network with the same structure. Both networks start with randomly chosen initial weight vectors w^A and w^B, this initial state is kept secret. Both partners A and B apply a common and publicly available input vector and calculate next the results of their networks. In the next step they exchange their network's results on an open channel. Each party treats the result of the network of another side as the expected result for himself and teaches its own network accordingly (by selecting one of the agreed before learning scheme). The next step is to choose randomly new input vector and the above procedure is subsequently repeated. Upon performing some number of such cycles, both networks' weights coincide and they can be used as cryptographic keys for the established communication. This bidirectional interaction of two synchronizing networks leads to reaching much faster equal weights vectors, than unidirectional one, which potential attacker could also do.

TPM defines a multilayer, feed-forward network with characteristic tree-like structure created by non-overlapping receptive fields. Figure 1 shows the structure of the typical TPM network.

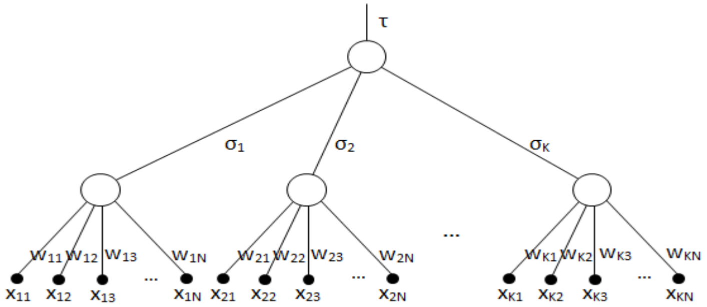


Fig. 1. TPM structure

The hidden layer of this network contains K neurons, each of which has N input signals. The input $x_{ij} \in \{-1,1\}$, where $1 \le i \le K$ and $1 \le j \le N$. Thus, the entire network is made up of the KN inputs. Each input signal is multiplied by the corresponding weight $w_{ij} \in \{-L, -L+1, \ldots, L-1, L\}$. Hidden layer neurons are equipped with the bipolar, step activation function, given by the following formula:

$$\sigma_i = \begin{cases} -1, if \ \sum_{j=1}^{N} x_{ij} w_{ij} \le 0, \\ \ \ 1, if \ \sum_{j=1}^{N} x_{ij} w_{ij} > 0. \end{cases} \tag{1}$$

Commonly in the literature, the TPM network is shortly described with the aid of three parameters: *K-N-L*. The last layer neuron multiplies hidden layer neurons' outputs and the result of its action is deemed as outcome of the entire network. This operation is given by following formula:

$$\tau = \prod_{i=1}^{K} \sigma_i. \tag{2}$$

TPM network has *KN* weights being integers within the range from *-L* to *L*. Thus, at each synchronization step, it may take one of $(2L+1)^{KN}$ states. For example, the weights of the network with parameters 3-101-3, takes one of $7^{3\cdot 101} \approx 1{,}16 \cdot 10^{256}$ states. The cryptographic key generated using weights would be $3 \cdot 3 \cdot 101 = 909$ bits long.

The network is considered as trained only if its result is equal to the expected one, like in classical Hebbian method [7], [12]. In mutual learning, the expected values are generated by the other network. Hence our two networks are trained only if they both have equals outputs. Each network involved is trained in accordance with one of three methods:

1. Anti-Hebbian learning rule:

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} - x_{ij}\sigma_i. \tag{3}$$

2. Hebbian learning rule:

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} + x_{ij}\sigma_i. \tag{4}$$

3. Random-Walk learning rule:

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} + x_{ij}. \tag{5}$$

If the new value of the weight is greater than *L*, it is replaced by *L*, and analogously, if the value is less than *–L*, it is replaced with *–L*, respectively. In the first learning rule weights are modified once the results of both networks are different, and the process leads to the network synchronization with opposite vectors *w*. The remaining two methods lead to synchronization with the same weight values.

Each pair of neurons form synchronizing TPMs in each learning step perform one of three admissible weight change named in [16] *quiet, attractive* or *repulsive* step. Let *i* denote the index of neuron in both networks *A* and *B*.

1. $\tau^A \neq \tau^B$ or $\tau^A = \tau^B \wedge \sigma_i^A \neq \tau^A \wedge \sigma_i^B \neq \tau^B$: then *A* and *B* do not change the weights' values at all or don't change the weights of *i*-th neuron. In both these cases it is the so-called "quiet step".
2. $\tau^A = \tau^B \wedge \sigma_i^A = \tau^A \wedge \sigma_i^B = \tau^B$: then the weight vectors w_i^A and w_i^B are modified according to one of the listed above learning rules. The modification applied depends on the common input vector x_i and equal neurons output, so that the change is coordinated, because all these values are the same for both networks. Weight vectors change in the same direction, and this case is called "attractive step".

3. $\tau^A = \tau^B \wedge [(\sigma_i^A = \tau^A \ \wedge \ \sigma_i^B \neq \tau^B) \vee (\sigma_i^A \neq \tau^A \ \wedge \ \sigma_i^B = \tau^B)]$: this means, that the outputs of A and B are the same and one of the following events takes place:
   (a) the analyzed neuron of network A has the same output as the result of a whole network and the corresponding neuron in network B generates an output different than whole network B,
   (b) the i-th neuron of network A has different output, than the whole network A, and in network B the corresponding neuron has output equal to whole network B output.

   In both events, only one neuron, which output is equal to the whole network output, modifies its weights, while the weights of the second network's neuron remain unchanged. In general, this procedure results in a reduction of weight vectors compatibility, and this step has been called "repulsive step".

Synchronized TPM networks end up with the same weight vectors and remain synchronized regardless of the time of further learning. For each input vector both networks return the same result, so in each next learning step both network pass through the learning procedure and almost every time networks modify their weights. If K is odd, one changes here the weight of at least one neuron in both networks. For K even there is only one case in which the weights of entire network are not modified. Indeed the latter occurs when all of the hidden layer neurons return -1 and the entire network return 1. For such specific case no neuron will change its weight. In opposite, any other combinations of hidden layer outputs yield at least one neuron changing its weights. Summing-up, synchronized networks typically change the weight during next learning steps which every time yields the same changes in two networks and thus both networks remain synchronized.

The condition enforcing the termination of network's synchronization is to obtain exchange of consistent results of both networks in a sufficiently long period of time. As examined experimentally (see section Results of this paper), in 15000 analyzed synchronizations of the network with the structure 3-101-3 the longest exchange of consistent results by not synchronized network is 147 steps. Thus to be sure that both networks are already synchronized, they should exchange over 147 consistent outputs.

Synchronization is a stochastic process, and the time required to achieve the consistent weights values depends on the initial weights and the input signals generated at each step. The synchronization times of network with a given structure, creates histograms as shown in figure 2 [11]. Such histogram is created after 15000 synchronizations of the network 3-101-3, where the Random Walk learning rule is applied. The number of classes in the histogram is given by the Hunstberger formula $k = 1 + 3{,}32 \cdot \log N$ (see [17]), where here $N = 15000$ and is doubled subsequently for better readability of the chart. Along X-axis the number of learning cycles needed to achieve networks full synchronization is given. On the other hand along Y-axis the probability of finding TPM synchronized in a given number of cycles is specified.

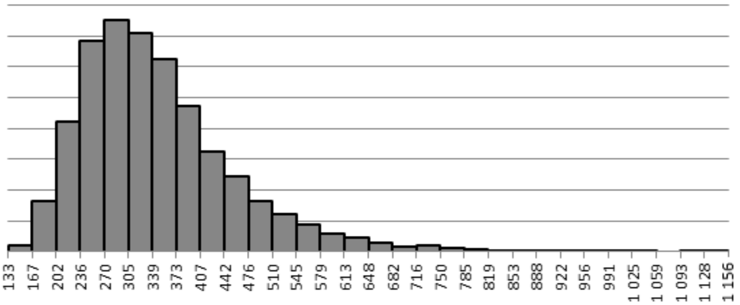


Fig. 2. Times of TPM 3-101-3 synchronization

3 Results

In this paper we analyze the synchronizations of the examined networks with the structure 3-101-3. Such networks enable to agree on key length of 909 bits. The first parameter K specifying the number of neurons in a hidden layer is set to 3, which is a typical value used commonly in TPM networks. The parameter $L=3$ sets the minimum and maximum weight value -L and L, respectively. Each weight must therefore be the integer from interval $[-3,3]$, yielding 7 possible values, which is close to the power of 2. This choice of parameter allows to easily generate a fairly uniform distribution of bits with values 0 and 1 in a generated key. The number of bits needed to store the value of the weights is given by $\log_2(2L+1)$ but in fact one has to use $\lceil\log_2(2L+1)\rceil$ bits (here symbol $\lceil\ \rceil$ denote the standard ceiling function). Therefore, it is important to minimize the difference between these values and select L which minimizes the following cost function

$$b(L) = \lceil\log_2(2L+1)\rceil - \log_2(2L+1). \tag{6}$$

Figure 3 shows the function b depending on the selection of the parameter L and taking the values of redundancy in bit representation of weight. It is transparent from the picture that the corresponding value of L is given by $2^n - 1$ for some integer n.

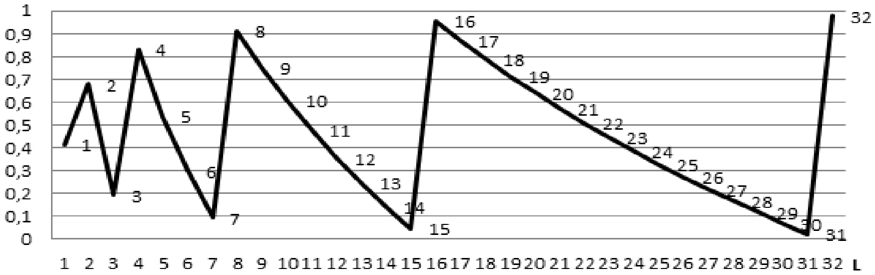


Fig. 3. Function b

Our discussed network is synchronized 15000 times. The fastest synchronization is completed after 133 cycles, while the longest lasted for 1156 cycles. The average synchronization time is in turn 346 cycles. Due to the shape of the histogram (see figure 2) it is interesting to observe that the third quartile value reads as 398, which is slightly above 34% of longest synchronization time. The latter means that 75% of the networks synchronize relatively fast, but there is a group of cases with longer synchronization pattern [18].

Previous research on the synchronization dynamic [7], [16] is based on weights' vectors mutual overlap for each hidden layer neurons. This measurement tool is determined by the cosine of the angle between the vectors of weights (corresponding to the respective neurons in both networks A and B) according to the following formula:

$$\rho_i^{AB} = \frac{w_i^A \circ w_i^B}{\sqrt{w_i^A \circ w_i^A} \cdot \sqrt{w_i^B \circ w_i^B}}. \tag{7}$$

At the initial synchronization phase the cosine renders a value close to 0, while in the end, when TPMs have the same weights, its value is equal to 1. In [19] different approach is presented. Namely, one analyzes the weights' vectors of all neurons and uses the Euclidean distance of these vectors

$$dist(A,B) = \|w^A - w^B\| = \sqrt{\sum_{k=1}^{KN}(w_k^A - w_k^B)^2}. \tag{8}$$

This distance is connected with previously used mutual overlap by cosine theorem for unitary spaces.

This parameter changes during the synchronization from relatively high values for non-synchronized network to 0 for synchronized one. Therefore, it has to be normalized and reversed to preserve the same characteristic as cosine. The normalization operation is accomplished by applying the formula:

$$dist(A,B)_t = 1 - \frac{dist(A,B)_t - \min_{1\le j\le t_{synch}} dist(A,B)_j}{\max_{1\le j\le t_{synch}} dist(A,B)_j - \min_{1\le j\le t_{synch}} dist(A,B)_j}. \tag{9}$$

In figure 4 we present the performance of the Modified Euclidean distance and the cosine measurement for the network 3-101-3 that synchronize within the time equal to the average synchronization time from the all analyzed network's time. The horizontal axis represents TPM's learning time and the vertical one, represents weights compatibility expressed as modified Euclidean distance and cosine.

Reversed normalized distance and cosine have significant linear correlation. Further analysis of this case (see table 3 and 4 below) with the mean square error between the cosine and normalized distance shows that both of these parameters are well-suited to describe the dynamics of synchronization process. In the example mentioned above the mean square error is around 0.016.

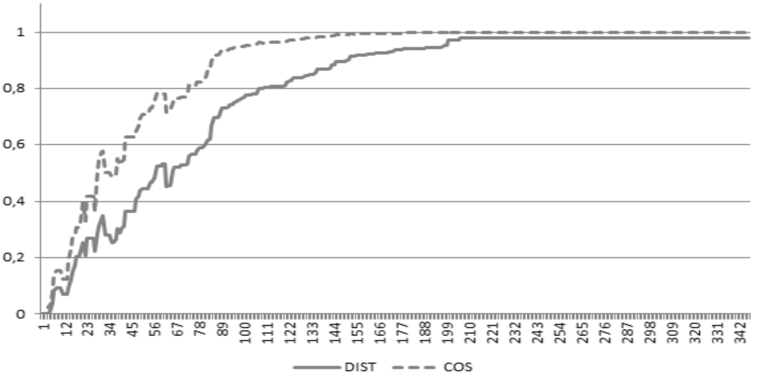


Fig. 4. Modified Euclidean distance and cosine for TPM 3-101-3

The presented above method for modifying distance measurements yield results that stay in good linear correlation with frequency of common output of synchronizing TPMs. Let the (a_n) be a sequence, and $a_n = 1$ if $\tau_n^A = \tau_n^B$ and $\mathrm{a_n} = 0$ if $\tau_n^A \neq \tau_n^B$, $\mathrm{n} = 1,2,\dots,\mathrm{t_{synch}}$. The sequence $(\mathrm{b_n})$ is defined as an average of s elements from $(\mathrm{a_n})$ with indices $\mathrm{n}, \mathrm{n}-1, \dots, \mathrm{n}-\mathrm{s}+1$. This sequence is given by formula:

$$b_n = \frac{1}{s}\sum_{j=n-l+1}^{n} a_n, \tag{10}$$

where $l = \min(n, s)$.

During the synchronization process, the weight's vectors of both networks are generally closer, resulting in achieving equal values. The dynamics of these changes can be described by the cosine of the angle between the weights' vectors or by using normalized and reverse Euclidean distance. In both cases, one must know the weights' vectors of the two networks, which is impossible for practical cryptographic key exchange protocol. Linear dependence of the network synchronization level and frequency of the same results exchanged by both networks, permits to treat such frequencies as a tool to determinate how quickly the network will complete the synchronization process. Analyzing these frequencies indicates the threshold at which networks have compatibility level good enough to finish synchronization process in a very short time.

Given 15000 synchronizations of networks with parameters 3-101-3, 5 networks are selected that synchronized after 200 and after 900 cycles, respectively. Visibly, the first group includes the networks that synchronized relatively quickly, while the second one contains networks with the longest observed synchronization time. For each of tested networks we analyzed frequencies of compatible output of the network in 25, 50, 75, ..., 125 previous steps. Next we verified if any of these frequencies exceeds the threshold value 0.7, 0.6, 0.65, ..., 0.85. The tables 1 and 2 illustrate the correlation coefficient between the reversed normalized Euclidean distance and the cosine (see the first column). In addition, they present the correlation coefficient between this distance and frequency calculated for given number of previous steps (the next columns). The green's intensity of the background indicates the best results. Table 1 lists generated herein results for networks synchronized fast in 200 cycles, whereas table 2 presents the same results for networks with synchronization time above 900 cycles. It is transparent once inspecting both tables, that the best fit is obtained for frequencies calculated on the basis of the latest 50-100 steps.

Table 1. Correlation coefficient for TPM synchronized in 200 cycles

	dist, cos	dist, fr 25	dist, fr 50	dist, fr 75	dist, fr 100	dist, fr 125
TPM 1	0,9675	0,9208	0,9547	0,9726	0,9736	0,9637
TPM 2	0,9463	0,9773	0,9851	0,9809	0,9758	0,9688
TPM 3	0,9452	0,9588	0,9820	0,9790	0,9673	0,9508
TPM 4	0,9543	0,9515	0,9579	0,9652	0,9702	0,9683
TPM 5	0,9583	0,9740	0,9818	0,9722	0,9510	0,9239

Table 2. Correlation coefficient for TPM synchronized in 900 and more cycles

	dist, cos	dist, fr 25	dist, fr 50	dist, fr 75	dist, fr 100	dist, fr 125
TPM 1	0,9171	0,8560	0,8857	0,8900	0,8875	0,8730
TPM 2	0,9349	0,8804	0,9092	0,9057	0,8998	0,8896
TPM 3	0,9403	0,9084	0,9193	0,9103	0,8887	0,8672
TPM 4	0,9513	0,9215	0,9519	0,9391	0,9227	0,9109
TPM 5	0,9359	0,8479	0,8844	0,8796	0,8544	0,8278

In addition for acquiring high linear relationship between distance and frequency it is also important to determine the compatibility of the above results. In order to achieve the latter we calculated the mean square error for the analyzed network. The corresponding results are demonstrated below in tables 3 and 4. The outcomes from these tables refer to both networks synchronized either in 200 or above 900 cycles, respectively. The obtained output structure is analogous to the previous tables. Again the green's intensity of the background indicates the best results. As previously the best results are obtained for frequencies of 50-100.

Table 3. Mean square error for TPM synchronized in 200 cycles

	dist, cos	dist, fr 25	dist, fr 50	dist, fr 75	dist, fr 100	dist, fr 125
TPM 1	0,0278	0,0376	0,0188	0,0111	0,0145	0,0256
TPM 2	0,0363	0,0113	0,0047	0,0074	0,0151	0,0273
TPM 3	0,0268	0,0089	0,0079	0,0187	0,0368	0,0598
TPM 4	0,0368	0,0152	0,0114	0,0130	0,0199	0,0319
TPM 5	0,0189	0,0071	0,0057	0,0164	0,0363	0,0624

Table 4. Mean square error for TPM synchronized in 900 and more cycles

	dist, cos	dist, fr 25	dist, fr 50	dist, fr 75	dist, fr 100	dist, fr 125
TPM 1	0,0347	0,0121	0,0095	0,0089	0,0092	0,0108
TPM 2	0,0395	0,0189	0,0148	0,0134	0,0125	0,0124
TPM 3	0,0514	0,0265	0,0232	0,0219	0,0222	0,0229
TPM 4	0,0350	0,0141	0,0098	0,0097	0,0105	0,0112
TPM 5	0,0365	0,0217	0,0169	0,0157	0,0167	0,0180

Figures 5 and 6 illustrate the reverse normalized Euclidean distance calculated from the knowledge of the weights' vectors for both networks and the frequency of reaching the common results by two networks in 50 and 75 previous cycles. There is one network synchronization in 200 and one in over 900 cycles, respectively.

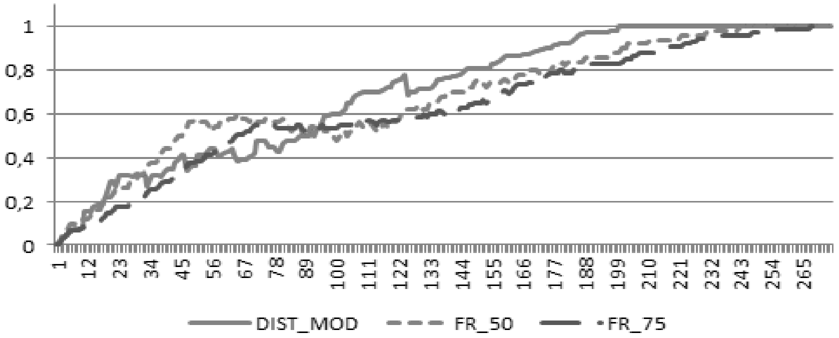


Fig. 5. Dynamics of short synchronization

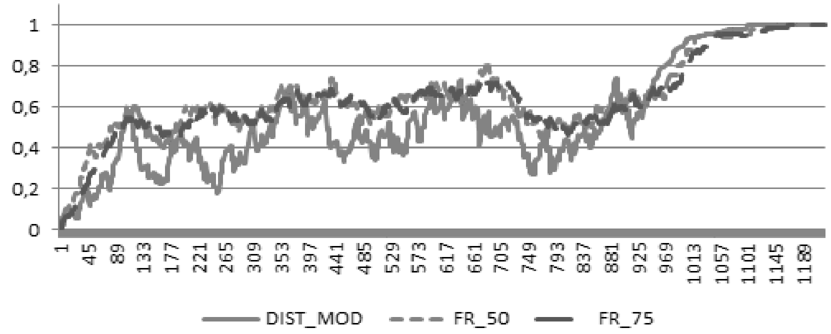


Fig. 6. Dynamics of long synchronization

As indicated in figures 5 and 6 in a long synchronization mode, there exists a threshold value close to 0.8 which, when exceeded in a some number of subsequent cycles, results in fast synchronization. In the case of networks that synchronizes quickly, calculated frequencies exceed this threshold also shortly before full synchronization. The experimental analysis of different threshold values ranging within 0.6 to 0.85 for all

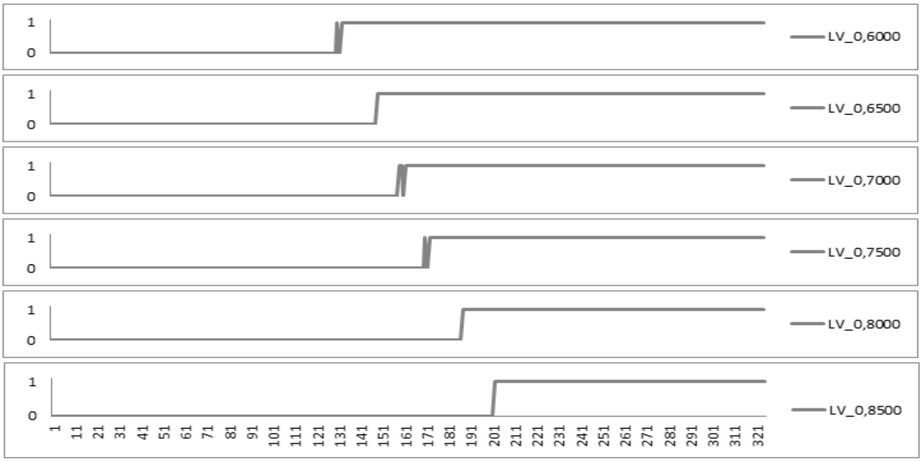


Fig. 7. Crossing the threshold for TPM with synchronization time 200

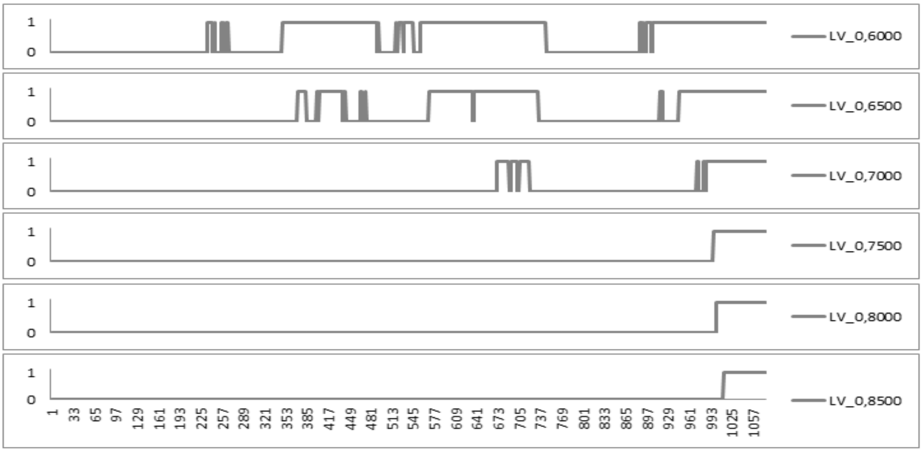


Fig. 8. Crossing the threshold for TPM with synchronization time over 900

analyzed frequencies is also conducted. The fact of passing over a given threshold can be presented in the following schemes for frequencies calculated in 75 previous steps (figures 7 and 8):

It is visible that frequencies generally increase during the network learning. This is consistent with an increase of compatibility of weights' vectors of the synchronizing networks. Thus the bigger threshold values are crossed by calculated frequencies after longer networks learning.

In the long synchronization only threshold 0.6 is exceeded before average learning time, which is 346 cycles for this TPM. Thus the use of the threshold set to 0.65 in this case indicates that there will be a high-speed synchronization. If this threshold is not exceeded in average or in third quartile time, it will be better to start synchronization with new, random weights.

4 Conclusions

The method of frequency analysis for exchange of equal results in TPM networks can be used to assess the synchronization level of both networks. As experimentally shown in this paper the calculated frequencies are not only strongly correlated with the distance between the weights' vectors but also are close to the values of modified distance. The latter is obtained by the small mean square error analysis. The charts containing the results of experiments presented herein shows that selecting the proper range for counting frequency and threshold (to be permanently exceeded) permits to specify whether one deals with either a short or a long synchronization.

References

1. Menezes, A., Vanstone, S., Van Oorschot, P.: Handbook of Applied Cryptography. CRC Press (1996)
2. Stokłosa, J., Bilski, T., Pankowski, T.: Data Security in Informatical Systems, PWN (2001) (in Polish)
3. Stinson, D.R.: Cryptography, Theory and Practice. CRC Press (1995)
4. Kanter, I., Kinzel, W., Kanter, E.: Secure Exchange of Information by Synchronization of Neural Networks. Europhys. Lett. 57, 141–147 (2002)
5. Klein, E., Mislovaty, R., Kanter, I., Ruttor, A., Kinzel, W.: Synchronization of Neural Networks by Mutual Learning and its Application to Cryptography. In: Advances in Neural Information Processing Systems, vol. 17, pp. 689–696. MIT Press, Cambridge (2005)
6. Klimov, A., Mityagin, A., Shamir, A.: Analysis of Neural Cryptography. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 288–298. Springer, Heidelberg (2002)
7. Ruttor, A.: Neural Synchronization and Cryptography, Ph.D. thesis, Würzburg (2006)
8. Ruttor, A., Kinzel, W., Naeh, R., Kanter, I.: Genetic Attack on Neural Cryptography. Phys. Rev. E 73(3), 036121–036129 (2006)
9. Dourlens, S.: Neuro-Cryptography. MSc Thesis, Dept. of Microcomputers and Microelectronics, University of Paris, France (1995)
10. Kotlarz, P., Kotulski, Z.: On Application of Neural Networks for S-Boxes Design. In: Szczepaniak, P.S., Kacprzyk, J., Niewiadomski, A. (eds.) AWIC 2005. LNCS (LNAI), vol. 3528, pp. 243–248. Springer, Heidelberg (2005)
11. Kanter, I., Kinzel, W.: Neural Cryptography, cond-mat/0208453 (2002)
12. Hassoun, M.: Fundamentals of Artificial Neural Networks. MIT Press (1995)
13. Osowski, S.: Neural Networks in Algorithmic Approach, WNT (1996) (in Polish)
14. Rutkowski, L.: Methods and Technics of Artificial Intelligence, PWN, Warsaw (2006) (in Polish)
15. Kanter, I., Kinzel, W.: The Theory of Neural Networks and Cryptography. In: Proceeding of the XXII Solvay Conference on Physics, The Physics of Communication, pp. 631–644 (2003)
16. Rosen-Zvi, M., Klein, E., Kanter, I., Kinzel, W.: Mutual Learning in a Tree Parity Machine and its Application to Cryptography. Phys. Rev. E 66, 066135 (2002)
17. Huntsberger, D.V.: Elements of Statistical Inference. Allyn and Bacon (1961)
18. Dolecki, M.: Tree Parity Machine Synchronization Time – Statistical Analysis. Mathematics, Physics and Informatics Series, Minsk 6(153), 149–151 (2012)
19. Dolecki, M., Kozera, R., Lenik, K.: The Evaluation of the TPM Synchronization on the Basis of their Outputs. Journal of Achievements in Materials and Manufacturing Engineering 57(2), 91–98 (2013)

The Removal of False Detections from Foreground Regions Extracted Using Adaptive Background Modelling for a Visual Surveillance System

Dariusz Frejlichowski[1], Katarzyna Gościewska[1,2], Paweł Forczmański[1], Adam Nowosielski[1], and Radosław Hofman[2]

[1] West Pomeranian University of Technology, Szczecin
Faculty of Computer Science and Information Technology
Żołnierska 52, 71-210, Szczecin, Poland
{dfrejlichowski,pforczmanski,anowosielski}@wi.zut.edu.pl
[2] Smart Monitor, sp. z o.o.
Niemierzyńska 17a, 71-441, Szczecin, Poland
{katarzyna.gosciewska,radekh}@smartmonitor.pl

Abstract. For recent surveillance systems, the false detection removal process is an important step which succeeds the extraction of foreground regions and precedes the classification of object silhouettes. This paper describes the false object removal process when applied to the 'SmartMonitor' system — i.e. an innovative monitoring system based on video content analysis that is currently being developed to ensure the safety of people and assets within small areas. This paper firstly briefly describes the basic characteristics and advantages of the system. A description of the methods used for background modelling and foreground extraction is also given. The paper then goes on to explain the artefacts removal process using various background models. Finally the paper presents some experimental results alongside a concise explanation of them.

Keywords: 'SmartMonitor', visual surveillance system, video content analysis.

1 Introduction

The 'SmartMonitor' system is being developed to combine the advantages of small closed–circuit television systems (CCTV) and visual content analysis algorithms (VCA). It aims to increase the safety of individual clients and their assets — i.e. houses, apartments, shops, enterprises, etc. — and help to ensure security in their surroundings. Through customisation of system parameters, all users will be able to set individual safety rules and adjust the system sensitivity level to suit their actual needs. Such customizability is one of the most important advantages. The system will be an affordable solution and will utilize only commonly available hardware, i.e. a personal computer and digital camera(s). Human

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 253–264, 2013.

participation will be reduced to the minimum and only the calibration process will require user interaction. 'SmartMonitor', as an innovative surveillance system, will utilize specific algorithms to perform video analysis and to react to particular events in pre-specified ways. These algorithms will be integrated into six main system modules responsible for background modelling, object tracking, artefacts removal, object classification, event detection and system response.

The system will operate under four independent scenarios, which concern home/surroundings protection against unauthorized intrusions (Scenario A), supervision of a person who is ill (Scenario B), crime detection (Scenario C) and smoke/fire detection (Scenario D). Each scenario can be characterized by several possible actions and conditions, such as movement detection, object tracking, region limitations, object classification, object size limitations, object feature changes, weather conditions, and work time. The conditions which the system will finally work under are very important. For example, changes in lighting or the appearance of either shadows or reflections in a scene can significantly influence the background model and result in the foreground image being affected by false objects. Therefore, it is crucial to appropriately eliminate false detections in order to extract only the actual object of interest (OOI). Each OOI should be a coherent region of a pre-specified minimum size. More detailed system description is provided in [1]. 'SmartMonitor' has been described in [2] and [3] as well.

In this paper the process of false object removal for the 'SmartMonitor' system is presented. Firstly, it briefly describes the method used for background modelling and foreground extraction, which directly precedes the elimination of falsely detected regions. Additionally, the types of false detections used are summarised. Secondly, the artefacts removal process as applied to two background models using additional operations is described. The paper then presents some experimental results and a concise discussion of them. The database utilized in the process of artefacts removal is very specific and was prepared only for the experiments concerning SmartMonitor system. The final section contains summary and conclusions.

2 The Process of Background Subtraction and Types of False Detections

The background modelling, foreground localization and artefacts removal processes are elements of the background subtraction process, which is presented as a flowchart in Fig. 1. The first stage, image pre-processing, utilizes algorithms that are mainly for colour conversion or image enhancement and helps facilitate image processing during the subsequent stages. The background models are built adaptively using a Gaussian Mixture Model (GMM), based on various colour components. The foreground image is the result of the subtraction of the background image from a pre-processed video frame. During the data validation stage, the system verifies whether the foreground regions correspond to real or false objects. This is done during the false detection removal process and the final foreground binary mask should only be created for the regions that the system is interested in.

Building an adequate background model is very important since the model influences both the obtained foreground regions and the number of false detections. The background model must be both sensitive enough to localise any real moving objects and sufficiently robust to particular environmental changes in the scene. In this case it cannot be static [4] or averaged in time [5] because real scenes are very changeable in time. Therefore, the GMM method (e.g. [6–8]) has been selected. It allows the adaptive modelling of each pixel as a mixture of Gaussians. This method has been evaluated to be a reliable real-time tracker that can quickly adapt any changes appearing within the scene and can deal with slow variations in illumination and repeated disturbances from unexpected motion in the background clutter [9].

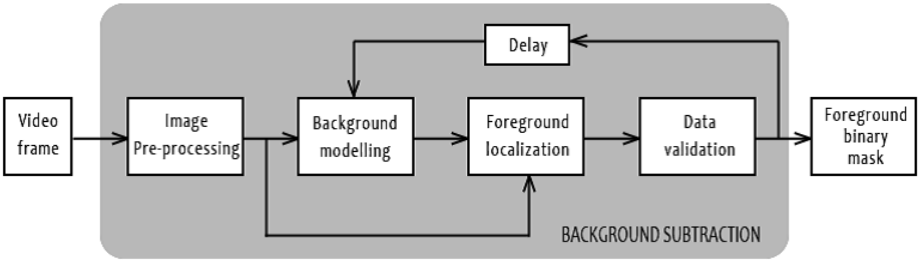


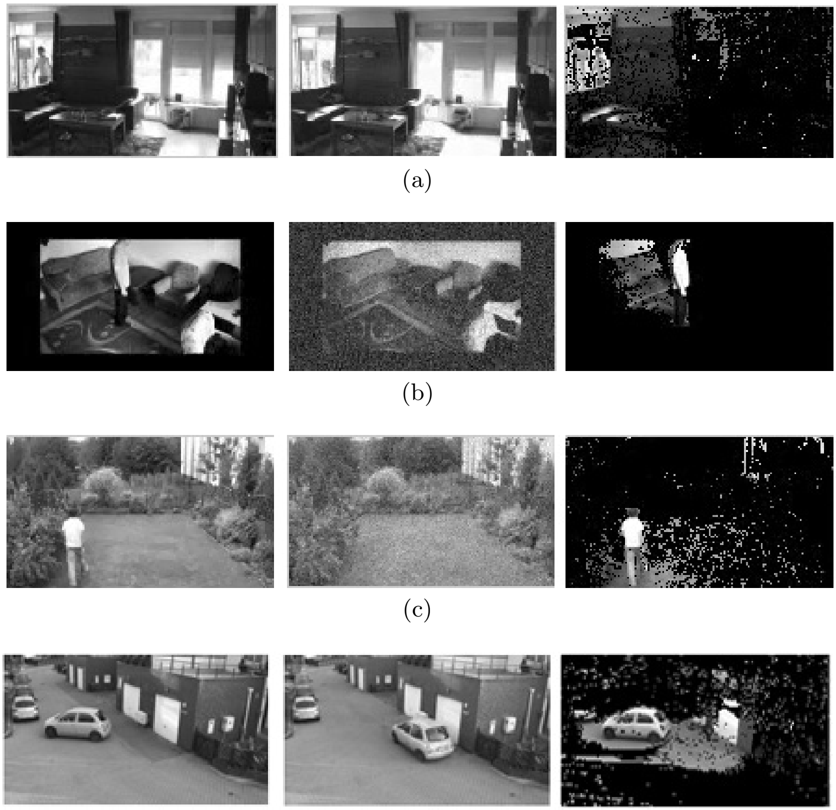
Fig. 1. Flowchart of the background subtraction algorithm (based on [10])

Despite the advantages, the method has some drawbacks that are mainly associated with the utilized colour information and the type of the environment which is present in the scene. False detections can take various forms, such as large coherent regions, isolated pixels or small groups consisting of several pixels. The reasons for artefacts occurrence are [11]:

- sudden changes in illumination, that can completely change the background colours and increase the difference between the model and the current frame;
- shadows of moving objects, that cause illumination changes and might be classified as foreground regions;
- background movements — i.e. the relocation of a part of the background, which causes a change within two regions, namely the newly acquired and previous positions. Both become considered as part of the foreground whereas only the previous position should;
- background initialization in the presence of moving objects — moving objects that belong to the foreground are mistakenly incorporated into the background and partly occlude it.

Fig. 2 shows examples of various types of false detections obtained for a background model with 256 grayscale values. Each row contains a sample video frame, a background model and a foreground image respectively. It is evident that the resulting foreground regions are larger than they should be, with the area of each moving object being surrounded by redundant pixels. Fig. 2 illustrates the results of:

(a) a sudden illumination change caused by sunrays passing through a window. Other possible causes are: turning the light on and off, a camera flash or changing weather conditions such as a partially cloudy sky with sun shining through the clouds;
(b) the appearance of a moving object shadow area;
(c) the movement of small background elements, mainly tree leaves and grass. As a result the foreground region is very noisy;
(d) model initialization with a frame containing a moving object which occludes part of the background. Because there is no information about the occluded background region, it is considered as part of the foreground.



(a)

(b)

(c)

(d)

Fig. 2. Examples of false detections for the intensity model

In summary, the foreground image can include pixels that correspond to both moving objects and false detections. Artefacts can take the form of larger coherent regions, small groups of pixels, or single isolated pixels. False detections are often connected with the actual object of interest, which makes them more difficult to eliminate. Areas of noise can also result from image compression or low data transmission quality.

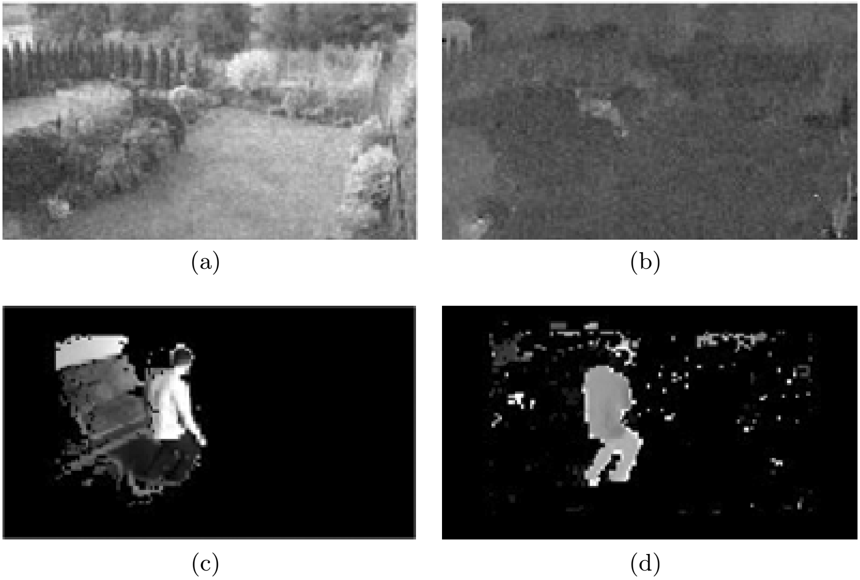
3 The Process of Artefacts Removal

For the 'SmartMonitor' system artefacts removal process, three key issues were taken into consideration and needed to be eliminated, namely the influence of a moving object that was visible in the scene during background model initialization, shadow areas and noisy regions. First of all, two separate background models based on the Y component (intensity) of the YIQ colour scheme and the H component (hue) of the HSV colour scheme were built. Various background models (see Fig. 3(a) and Fig. 3(b) for examples) allowed different information about possible ways to simplify the artefacts removal process to be obtained. Moreover, additional operations and modifications were introduced in order to eliminate the aforementioned false detections.

The first problem, i.e. the presence of a moving object in the background model, occurs mainly when a model is initialized using the first frame obtained from the video stream. However, it is often impossible to obtain an image without moving objects in it, especially from busy environments. Therefore, the initial background image pixel values should be random. On the one hand, this will cause the model to require more time to adapt to the current situation of the scene, and on the other, the resultant background image will be more accurate and the influence of moving objects will be reduced.

The next issue is associated with shadow detection and removal. Here, the utilization of two different background models is discussed. It was stated in [9] that by building the background model using intensity values it is impossible to distinguish between moving objects and moving shadows. The authors of [12] proposed the utilization of a colour component to detect moving shadows and to reduce the computation time. According to [13], shadows affect only intensity values and not the hue of shaded and open regions to a significant extent. Therefore, the comparison of two different foreground images, obtained using intensity and hue background models, allows for shadow exclusion. Two exemplary foregrounds are illustrated in Fig. 3(c) and Fig. 3(d).

The last problem to be solved is that of the elimination of noisy areas. This is done in two stages and each foreground image is processed separately before shadow exclusion is performed. Firstly, image thresholding is performed in which non-zero pixel values are changed to 1. This results in an image with white foreground areas and a black background. Secondly, two morphological operations are applied — i.e. erosion and dilation. Erosion allows for the elimination of isolated groups of several pixels, hence resulting in reduction of the foreground region. Dilation then fills–in the holes and completes the pixels removed from the object region. Afterwards, the process of shadow elimination takes place. Here, two foreground binary images without noisy areas are multiplied using an entrywise product — i.e. when two pixels of the same coordinates are considered to be part of the foreground, their respective pixel in the final binary mask is white, otherwise, the zero value is assigned. A pictorial representation of the particular stages of the artefacts removal process is shown in Fig. 4.


(a)

(b)

(c)

(d)

Fig. 3. Examples of: (a) an intensity background model, (b) a hue background model, (c) an intensity foreground image and (d) a hue foreground image

As can be seen from Fig. 4, both the shadow area and the majority of noisy regions were eliminated. The obtained foreground mask contains a moving object shape and several smaller regions that are not taken into consideration since the system only analyses objects of a pre-specified minimum size.

4 Experimental Conditions and Results

The experiments were performed using a test database containing a set of video sequences which were suitable for pre-planned system scenarios with specified parameters. The main goal was to investigate the effectiveness of the false object removal process when applied to the 'SmartMonitor' system. Each experiment was carried out in the same way. The background model was first initialized with random pixel values and then iteratively adapted to each subsequently processed frame. The subsequent stages of the background subtraction process for an individual video frame are shown in Fig. 5.

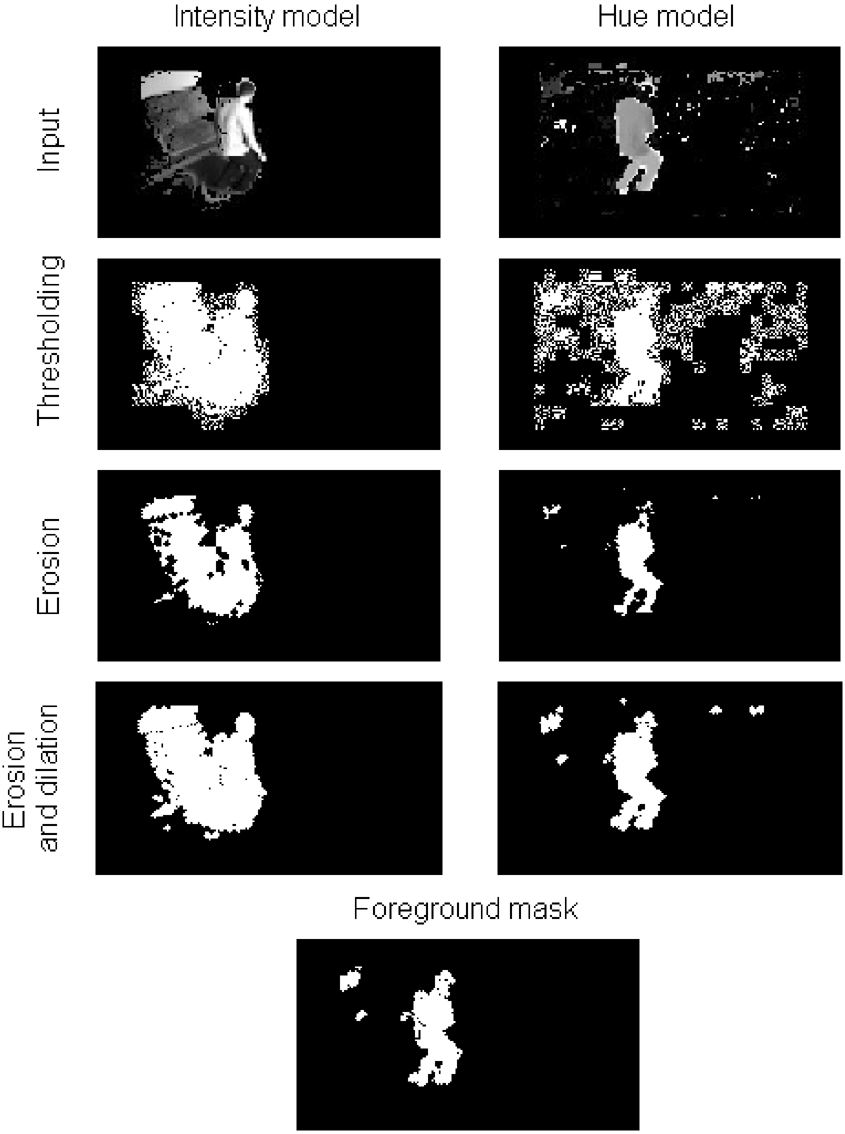


Fig. 4. The illustration of the consecutive steps of the artefact removal process: input image thresholding, erosion, dilation and foreground mask extraction

Fig. 6–9 present some experimental results for various system working conditions and different environments — i.e. scenes recorded both inside and outside a building, with changing weather or lighting conditions, and background movements present. Each figure contains: (a) a sample frame, (b) a foreground image of the intensity model, (c) a foreground image of the hue model and (d) a foreground binary mask. A sample frame from the video sequence fulfilling the conditions of Scenario A, which shows a person walking in a garden, is depicted in Fig. 6(a). A shadow area was detected in the intensity model (Fig. 6(b)) and the foreground (Fig. 6(c)) was affected by noisy areas resulting from background movements and video compression. However, the moving object shape was extracted as expected (Fig. 6(d)). Fig. 7 shows the results for Scenario B. Here, the sample frame shows a person who fell over and is lying still. Noise and shadow areas were smaller than in the previous example and were completely removed. A similar situation is visible in Fig. 8, which concerns Scenario C, with sample frame presenting a person with hands raised. The moving shadow visible in Fig. 9(a) was also eliminated and so the final binary mask (Fig. 9(d)) does not contain any foreground areas.

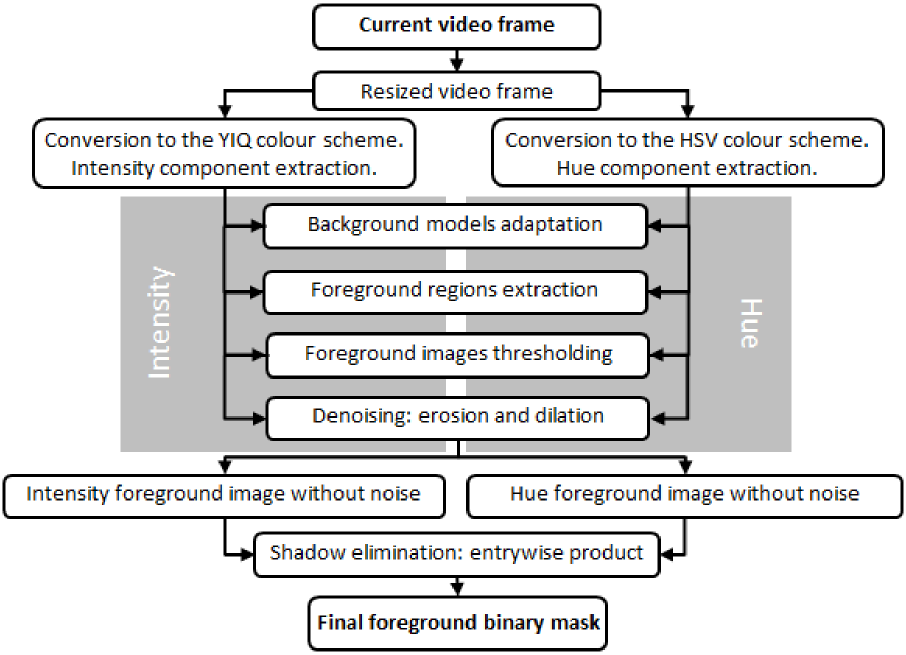

Current video frame
Resized video frame
Conversion to the YIQ colour scheme. Intensity component extraction.
Conversion to the HSV colour scheme. Hue component extraction.
Intensity
Hue
Background models adaptation
Foreground regions extraction
Foreground images thresholding
Denoising: erosion and dilation
Intensity foreground image without noise
Hue foreground image without noise
Shadow elimination: entrywise product
Final foreground binary mask

Fig. 5. The diagram of background subtraction process with artefacts removal stages

Experimental results were evaluated visually by the users and showed that the majority of falsely detected objects have been successfully removed. This proves the effectiveness of the false object removal process in application to the 'SmartMonitor' system. There is no available data that could be compared with the results as a ground–truth — the utilized database is very specific and adapted to system scenarios.

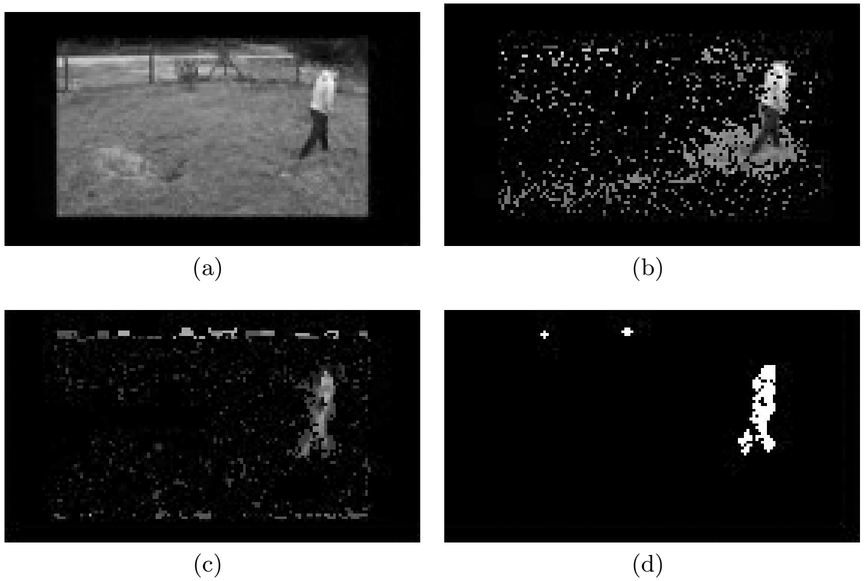

(a)
(b)
(c)
(d)

Fig. 6. Exemplary results of the artefacts removal process for scenario A

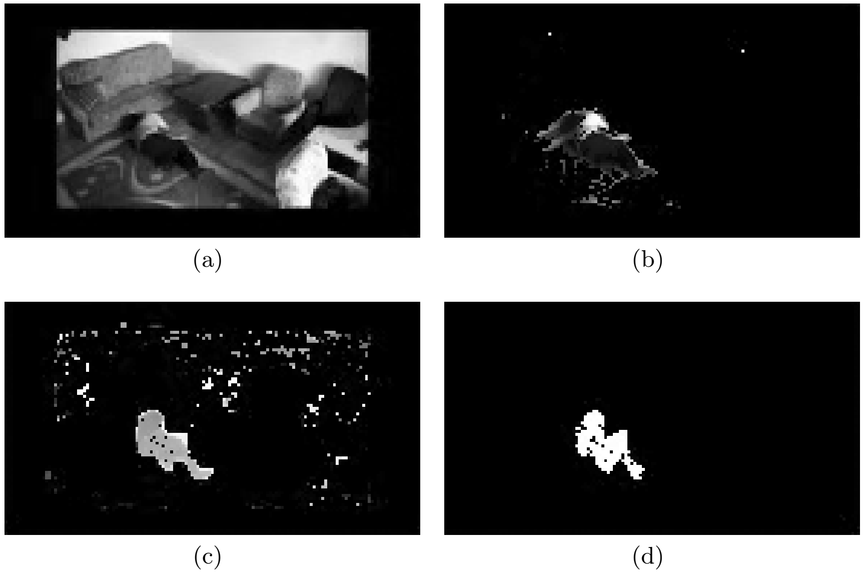

(a)
(b)
(c)
(d)

Fig. 7. Exemplary results of the artefacts removal process for scenario B

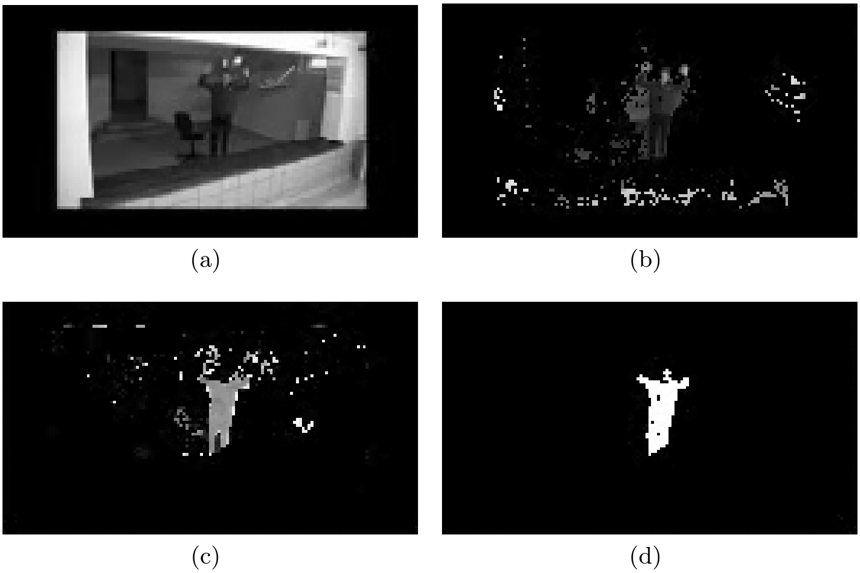


Fig. 8. Exemplary results of the artefacts removal process for scenario C

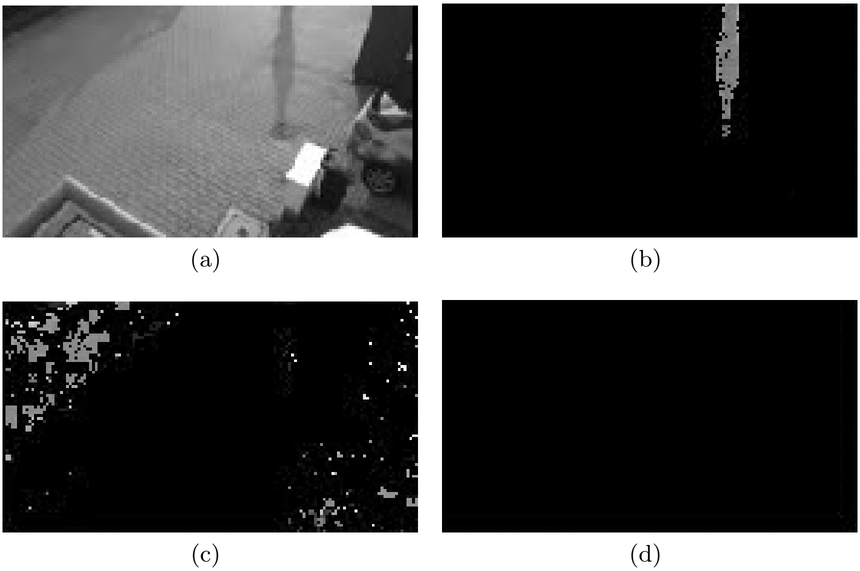


Fig. 9. Exemplary results of the artefacts removal process for shadow removal

5 Summary and Conclusions

In the paper, the application of the artefacts removal process to the 'SmartMonitor' system was presented. Firstly, it introduced the main characteristics of the system and some basic information about the background subtraction process. The method for eliminating false detections within two background models was then described. Despite the fact that the system is currently under development, some experimental results were also provided and discussed.

Different background models allow various foreground images to be obtained. Therefore, we focused on the intensity component of the YIQ colour scheme and the hue component of the HSV colour scheme in order to exclude shadow areas. Moreover, we introduced additional operations and modifications in order to eliminate false detections. These were morphological operations for noise reduction, and the use of an initial background model with random pixel values to decrease the influence of moving objects present during initialization. The experiments showed that the proposed method gave satisfactory results. However, several additional issues, such as camera placement or system sensitivity level, could influence both the number of artefacts and number of false alarms.

Acknowledgements. The project *'Innovative security system based on image analysis — "SmartMonitor" prototype construction'* (original title: *Budowa prototypu innowacyjnego systemu bezpieczeństwa opartego o analize obrazu — "SmartMonitor"*) is a project co-founded by the European Union (project number PL: UDA-POIG.01.04.00-32-008/10-01, Value: 9.996.604 PLN, EU contribution: 5.848.800 PLN, realization period: 07.2011-04.2013). *European Funds — for the development of an innovative economy* (*Fundusze Europejskie — dla rozwoju innowacyjnej gospodarki*).

References

1. Frejlichowski, D., Forczmański, P., Nowosielski, A., Gościewska, K., Hofman, R.: SmartMonitor: An Approach to Simple, Intelligent and Affordable Visual Surveillance System. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) ICCVG 2012. LNCS, vol. 7594, pp. 726–734. Springer, Heidelberg (2012)
2. Forczmański, P., Frejlichowski, D., Nowosielski, A., Hofman, R.: Current trends in developing of intelligent visual monitoring systems (in Polish). Methods of Applied Computer Science 4/2011(29), 19–32 (2011)
3. Frejlichowski, D., Gościewska, K., Forczmański, P., Nowosielski, A., Hofman, R.: SmartMonitor: recent progress in the development of an innovative visual surveillance system. Journal of Theoretical and Applied Computer Science 6(3), 28–35 (2012)
4. Horprasert, T., Harwood, D., Davis, L.S.: A robust background subtraction and shadow detection. In: Proceedings of the Asian Conference on Computer Vision (2000)
5. Frejlichowski, D.: Automatic Localisation of Moving Vehicles in Image Sequences Using Morphological Operations. In: Proceedings of the 1st IEEE International Conference on Information Technology, Gdańsk 2008, pp. 439–442 (2008)

6. Wang, W., Chen, D., Gao, W., Yang, J.: Modeling Background from Compressed Video. IEEE Transactions on Circuits and Systems for Video Technology 5, 670–681 (2008)
7. Piccardi, M.: Background Subtraction Techniques: A Review. In: IEEE International Conference on Systems, Man and Cybernetics, vol. 4, pp. 3099–3104 (2005)
8. Zivkovic, Z.: Improved Adaptive Gaussian Mixture Model for Background Subtraction. In: Proceedings of the 17th International Conference on Pattern Recognition, vol. 2, pp. 28–31 (2004)
9. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2–252 (1999)
10. Sen-Ching, S.C.S., Kamath, C.: Robust Techniques for Background Subtraction in Urban Traffic Video. In: Bhaskaran, V., Panchanathan, S. (eds.) Visual Communications and Image Processing, vol. 5308, pp. 881–892 (2004)
11. Javed, O., Shafique, K., Shah, M.: A Hierarchical Approach to Robust Background Subtraction Using Color and Gradient Information. In: Workshop on Motion and Video Computing, pp. 22–27 (2002)
12. Kaewtrakulpong, P., Bowden, R.: An Improved Adaptive Background Mixture Model for Real-Time Tracking with Shadow Detection. In: Proceedings of the 2nd European Workshop on Advanced Video Based Surveillance Systems, Computer Vision and Distributed Processing (2001)
13. Forczmański, P., Seweryn, M.: Surveillance Video Stream Analysis Using Adaptive Background Model and Object Recognition. In: Bolc, L., Tadeusiewicz, R., Chmielewski, L.J., Wojciechowski, K. (eds.) ICCVG 2010, Part I. LNCS, vol. 6374, pp. 114–121. Springer, Heidelberg (2010)

Using Backward Induction Techniques in (Timed) Security Protocols Verification[*]

Mirosław Kurkowski, Olga Siedlecka-Lamch, and Paweł Dudek

Institute of Computer and Information Sciences
Częstochowa University of Technology,
Dąbrowskiego 73, 42-201 Częstochowa, Poland
{mkurkowski,olga.siedlecka,pdudek}@icis.pcz.pl

Abstract. This paper shows a new way of automatic verification of properties of untimed and timed security protocols. To do this we use a modified version of previously introduced formal model based on a network of synchronized (timed) automata that expreses behaviour and distributed knowledge of users during protocol executions. In our new approach we will use the backward induction method for searching of a tree of all real executions of an investigated protocol. Our approach uses additionally the boolean encoding of constructed structures and SAT solvers for searching answers to the questions about investigated properties which are expressed as reachability or unreachability of undesired states in a considered model. We exemplify all our notions and formalisms on the well known NSPK, and show experimental results for checking authentication and security properties of a few untimed and timed protocols.

Keywords: Security Protocols Verification, Backward Induction.

1 Introduction

The modern people process tens of gigabytes of information a day, most of which is transferred electronically. "Consumption" of information is a must, it gives people the knowledge, capabilities, and does not allow to fall outside the margins of society. The transfer of information must remain undisturbed, consistent with reality and not threatened by outside intruders. Such communication should be guaranteed by a number of communication protocols, secured by modern cryptographic techniques. The heart of every communication protocol is a security protocol that meets the following requirements: mutual authentication of communicating entities, confidentiality of transmitted information, integrity of this information and a new key-session distribution.

Security protocols used in practice satisfy one or more of the above requirements using either symmetric or asymmetric cryptography. Many of the developed security protocols proved to be vulnerable to the attack by malicious Intruders [21, 22], and investigators undertook the challenge of formal methods of protocol verification.

[*] The second and the third author acknowledge support from EU Project (European Social Fund.) PO KL *Information technologies: Research and their interdisciplinary applications*, Agreement UDA***/*-*/*POKL.04.01.01-00-051/10-00.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 265–276, 2013.

The process of creating the protocol is difficult, but it seems more difficult to verify. The first step is to create a valid, full formal description. The description should include all parties, states and actions occurring in the protocol: participants (sender and receiver), the information about keys that encrypt the transmitted data, actions of generating, encoding, sending, receiving and decrypting information. The Common Language is a commonly used protocol specification language, which unfortunately does not allow for a full description. It shows only the scheme of sending messages and their construction. In this case there is a need for using more complicated languages, for example presented in the papers [19, 18, 17].

The precisely specified protocol can be automatically verified in many possible ways. The field of automatic verification has been exploited for many years by both academic and commercial institutions. Basically, there are two main ways of verification: testing real and virtual systems (simulations), or modeling and formal verification. Of course after testing the already implemented systems, we can be sure that the system has worked properly only so far. Formal modeling and verification is based on building adequate mathematical structures, showing the actions that occur during the protocol execution. Of course these methods are succesfully used to model and verify various kinds of computer systems. Good examples are [11, 16]. There are numerous approaches to the verification of security protocols associated sometimes with appropriate tools [1, 2, 14, 3, 8, 9, 12, 5].

The algorithmic approach is based predominantly on model checking, but we can distinguish among formal methods also those inductive [26, 4] or deductive [16, 6, 20]. Intuitively, model checking of a protocol proceeds in checking whether a formal model of protocol executions contains an execution or a reachable state that represents an attack upon the protocol. Comparing to standard model checking techniques for communication protocols or for distributed systems, the main difficulty is caused by the need to model the Intruder which is responsible for generating attacks as well as changes of knowledge (about keys, nonces, etc.) of the participants. The Intruder can be modeled in several ways, although the Dolev-Yao model and its versions seems to be the most adequate [10]. In mentioned model the Intruder has access to all the transmitted information, can collect and use it with a precision according to its skills and resources, and is able to send it somehow and sometime through the network. Of course, the Intruder does not need to comply with the requirements of using the protocols and, in particular, does not have to use fresh information or to meet time requirements.

Using model checking we face another serious problem - the exponential explosion of states, which depends on the number of participants, sessions, or messages. The model should predict all possible scenarios with a huge amount of information generated and sent by the Intruder. In considered case the computational complexity of model checking algorithms is typically exponential in parameters of the verified protocols, particularly in the number of participants and execution steps. This problem calls for the modification of existing models and algorithms, as well as finding new ones.

So far, our studies in this area were directed at the two models: a network of synchronized automata for modeling separately the executions of a protocol and the knowledge of the participants like in [19, 18, 17] or using the chains of states of protocol execution, see [27]. In the first case, the constructed networks of automata were translated to the propositional boolean formulas. The security property was firstly expressed as a property of reachability of certain global states in the automata network, and subsequently as a satisfiability of the formula. SAT-solver answered the question whether satisfying valuation exists, which was equivalent to the existence of an attack or an undesirable situation. By contrast, if the formula turned out to be unsatisfiable, it meant that no attack exists.

For the searching of models of executions of concurrent systems, the backward induction technique is sometimes used. Generally speaking the method consists in constructing a tree of executions in accordance to the reversed relation to that which determines the dependence between the states in the basic model. In some situations, for example in the case of testing the reachability of certain states, this causes the reduction of size of the constructed trees. In the case of protocols this approach was sucessfully used in a few works, for example in [15].

In this article we propose the approach of the backward induction technique for the investigation of properties of protocols by building a network of automata encoding the tree of protocol execution, where the runs take place inversely to the runs proposed in [19, 18, 17]. The automata built in this approach are slightly bigger in size, which is why the experimental results confirm greater efficiency of the proposed approach in the case of protocols with a small number of steps. It is interesting that this method may also be used in the case of time-dependent protocols, and the networks of time automata which model them.

2 The Needham-Schroeder Public Key Protocol

The designing of the communication protocols is a very hard task which is connected with the possibility of appearing many problems with later use of a made protocol. As the essential example the NSPK protocol can be presented [25].

In the notation of this protocol there are designations $i(A)$ and $i(B)$, which express identifiers of the participants of the communication, respectively the A and B participants. The expression $\langle X\rangle_{K_A}$ means the X message encrypted by the public key of A participant. Similarly, the message encrypted by the public key of B participant - $\langle X\rangle_{K_B}$.

The A participant has a very important role – starts the run of the protocol. The aim of the execution of this protocol is to achieve the mutual confirmation of the identity (the authentication) between the communicating participants. The designation $A \rightarrow B : X$ refers to sending the X message from the A participant to B participant. We assume that sending the message causes the operation of receiving it by the suitable person. The concatenation of the elements in the message was determined by the operator "·".

Example 1. The scheme of NSPK protocol proposed in [25] is as follows:
α_1 $A \rightarrow B$: $\langle N_A \cdot i(A)\rangle_{K_B}$,
α_2 $B \rightarrow A$: $\langle N_A \cdot N_B\rangle_{K_A}$,
α_3 $A \rightarrow B$: $\langle N_B\rangle_{K_B}$.

The A participant in first step of the protocol execution generates the random number (nonce) N_A and sends it to the B participant with A's identifier encrypted by the public key of the B user. The next step of protocol execution starts from generating by the B

participant its own random number N_B. Next it creates the message consisting of random numbers of both communicating participants, encrypts this message by the public key of the A participant and sends it to the A. In the next stage of the execution, A runs decrypting operation of the received message and the operation of comparing the N_A number, which got from B with the number N_A, which was prepared by him. If both numbers are equal, A considers B as authenticated. In the last step of the protocol execution A sends to B its random number N_B encrypted by the K_B key. After decrypting and comparing the proper numbers, the B participant can consider the A participant as authenticated. This protocol execution should guarantee both participants the identity of the communicating persons.

Example 2. Figure 1 presents the automata based model of honest execution of NSPK Protocol due to formal definitions from [19, 18, 17]. In the presented networks synchronisation of component automata is used, it consists in the fact that the global transition (in the network) labeled by the label α is enabled, if in all components if there exists transitions labeled by α, then at least one of them in each automaton is enable. It is assumed further that all states are accepting ones. As it can be seen the model consists of automata modeling the execution of external actions of the protocol (sending messages) $\mathcal{A}$ and automata that model distributed knowledge of participants ($\mathcal{A}^A_{N_A}, \mathcal{A}^A_{N_B}, \mathcal{A}^B_{N_A}, \mathcal{A}^B_{N_B}$). For example, the automaton $\mathcal{A}^A_{N_A}$ models the knowledge of the user A about the random number N_A. Executions of additional actions constituting the protocol are represented by labels on automata transitions. Transitions labeled by α_1 model execution of the first step of the protocol. During execution of that step, users A and B acquire knowledge about the number N_A. Transitions labeled α_2 model the performance of the second action of the protocol, in which users gain knowledge about the number N_B. We should pay attention on the loop in automata $\mathcal{A}^B_{N_A}$, labeled also by α_2, it models the condition to possess knowledge about number N_A by B, to execute the second step of the protocol. Accordingly, the transitions labeled α_3, model the execution of the third step of the protocol.

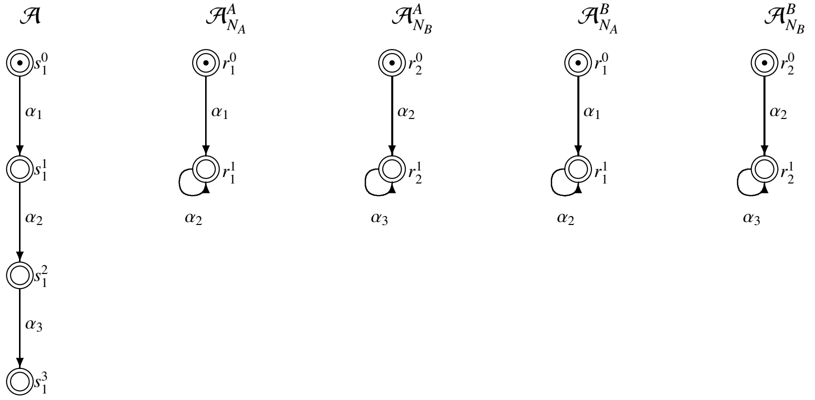


Fig. 1. Automata model of execution of the NSPK Protocol

The NSPK protocol was used through 17 years. In 1995 Gavin Lowe discovered the version of the protocol execution which consisted a possible attack, showing that the

NSPK protocol is susceptible to a break-in [21]. The Intruder is the additional participant with its own identifier and the keys, determined by ι symbol. The Intruder does not do the protocol according to the scheme but uses own ways, impersonates other users and cheats them.

Example 3. The attack presented by Gavin Lowe is as follows:

$\alpha_1^1\ A \rightarrow \iota\ :\ \langle N_A \cdot i(A)\rangle_{K_\iota},$

$\quad \alpha_1^2\ \iota(A) \rightarrow B\ :\ \langle N_A \cdot i(A)\rangle_{K_B},$

$\quad \alpha_2^2\ B \rightarrow \iota(A)\ :\ \langle N_A \cdot N_B\rangle_{K_A},$

$\alpha_2^1\ \iota \rightarrow A\ :\ \langle N_A \cdot N_B\rangle_{K_A},$

$\alpha_3^1\ A \rightarrow \iota\ :\ \langle N_B\rangle_{K_\iota},$

$\quad \alpha_3^2\ \iota(A) \rightarrow B\ :\ \langle N_B\rangle_{K_B}.$

The Lowe scheme shows two simultaneously NSPK protocol executions. The α^1 execution refers to the communication between the A participant and the Intruder. The α^2 execution refers to the situation where the Intruder pretends to be the A participant and in his name communicates with B. To have correct protocol which cannot be broken it is enough to add the identifier of the sender to sent message in the second step of the original NSPK protocol execution: $\langle N_A \cdot N_B \cdot i(B)\rangle_{K_A}$. All known verification methods and tools confirm safety of this version of NSPK.

3 Backward Induction in Modeling and Verification of Untimed Protocols

As mentioned earlier, in the verification of systems modeled by transitional structures, backward induction is sometimes used. This takes place when there is hope that it is easier to build an execution tree of a program or system by constructing it, so to say, from the end, moving backwards from a certain state in accordance to a specific relation of a passage in the structure. Examples indicate that sometimes this method is very efficient, especially in the case of stating unreachability of certain states (see [15]).

In order to verify security protocols using the backward induction method, we will now define the product automaton which models inversely the runs which take place in the product automaton constructed in previous sections. An appropriate theorem of the adequacy of the calculations done in both product automata will allow the searching of the tree using the backward induction method.

Let $\mathfrak{A}$ be a family of automata $\mathcal{A}_i$, $i \in I$ meeting the following conditions. Let $\mathcal{A}_i = \{Q_i, \Sigma_i, \delta_i, s_i^0, F_i\}$ be an automaton, where:

- $Q_i = \{s_i^0, s_i^1, \ldots, s_i^t\}$, for all $i \in I$ and some $t \in N$ is the set of states of automata $\mathcal{A}_i$,
- $\Sigma_i = \{k_i^1, k_i^2, \ldots, k_i^s\}$ is the set of labels, where ($\Sigma_i \cap \Sigma_j = \emptyset$, dla $i \neq j$) for all automata from $\mathfrak{A}$,
- $\delta_i \subseteq Q_i \times \Sigma_i \times Q_i$ is a transition relation that meets the following conditions:
  - $(s_i^l, k, s_i^j) \in \delta_i$ iff $l = j - 1$, for any $l = 0, 1, \ldots, s-1$ oraz $j = 1, 2, \ldots, s$,
  - if $(s_i^l, k, s_i^{l+1}) \in \delta_i$ and $(s_i^p, k, s_i^{p+1}) \in \delta_i$, then $l = p$.
- s_i^0 is an initial state, $F_i = Q_i$ is the set of finite states.

It is noticeable that this family defines the automata which have the same structure as automata modeling the execution of protocols defined in [17–19]. The following conditions have been met:

- the sets of states of all the automata are equally numerous.
- the relation of the passage in each automaton is determined only on a pair of states directly following one another.
- labels do not duplicate in different automata, nor do they duplicate within the scope of one automaton,
- all states of the automata are accepting ones.

We denote by Σ the set of all labels from automata from the family $\mathfrak{A}$ ($\Sigma = \bigcup_{i \in I} \Sigma_i$).

Now the family of automata corresponding to the knowledge automata will be defined.

Let $\mathfrak{B}$ be a family of automata $\mathcal{B}_j$, $j \in J$ that meets the following conditions. $\mathcal{B}_j = \{\mathcal{R}_j, \Sigma, \rho_j, r_j^0, T_j\}$ where:

- $\mathcal{R}_j = \{r_j^0, r_j^1\}$, for all $j \in J$,
- Σ is the set of labels constructed before,
- $\rho_j \subseteq \mathcal{R}_j \times \Sigma \times \mathcal{R}_j$ is the transition relation that meets the condition: $(r_j^l, k, r_j^p) \in \rho_j$ iff $l = 0 \wedge p = 1$ or $l = p = 1$,
- r_j^0 is an initial state, $T_j = \mathcal{R}_j$ is a set of finite states.

It is noticeable here that these automata have the same structure as knowledge automata constructed in the previous section:

- each have two states,
- transitions are defined only from the first state to the second, and from the second state to itself,
- labels are taken from the set of all labels from the automata of the $\mathfrak{A}$ family,
- all states are accepting ones.

By $\underline{\mathfrak{A}\mathfrak{B}}$ we denote the product automaton (network of automata) defined over the family $\mathfrak{A} \cup \mathfrak{B}$ due to definition of product automaton given in [17].

We will now define the product automaton modeling the runs of automaton $\underline{\mathfrak{A}\mathfrak{B}}$ executed inversely.

For all automata $\mathcal{A}_i \in \mathfrak{A}$ let $\overline{\mathcal{A}_i}$ be a smallest automaton that meets the following conditions:

$\overline{\mathcal{A}_i} = \{Q_i, \Sigma_i, \overline{\delta_i}, s_i^t, \{s_i^t, s_i^0\}\}$ where:

- $\overline{\delta_i} \subseteq Q_i \times \Sigma_i \times Q_i$ is a transition relation that meets the condition: if $(s_i^l, k, s_i^{l+1}) \in \delta_i$, then $(s_i^{l+1}, k, s_i^l) \in \overline{\delta_i}$ and $(s_i^{l+1}, k, s_i^0) \in \overline{\delta_i}$.
- s_i^t is an initial state, $\{s_i^t, s_i^0\}$ is a set of accepting states.

One should notice that the sets of states and labels are the same as in corresponding automata of the family $\mathfrak{A}$. By $\overline{\mathfrak{A}}$ we define the family of all such constructed automata.

Moreover for all automata $\mathcal{B}_j \in \mathfrak{B}$ we define automata $\overline{\mathcal{B}_j}$ as the smallest automata that meet for every $j \in J$:

$\overline{\mathcal{B}_j} = \{\overline{\mathcal{R}_j}, \Sigma, \overline{\rho_j}, r_j^2, \{r_j^0, r_j^2\}\}$ where:

- $\overline{\mathcal{R}_j} = \mathcal{R}_j \cup \{r_j^2\} = \{r_j^0, r_j^1, r_j^2\}$,
- Σ is the set of labels constructed before,
- $\overline{\rho_j} \subseteq \overline{\mathcal{R}_j} \times \Sigma \times \overline{\mathcal{R}_j}$ is the transition relation that meets the following conditions: if $(r_j^l, k, r_j^p) \in \rho_j$, then $(r_j^p, k, r_j^l) \in \overline{\rho_j}$ and $(r_j^2, k, r_j^l) \in \overline{\rho_j}$.
- r_j^2 is an initial state, $\{r_j^0, r_j^2\}$ is a set of accepting states.

By $\overline{\mathfrak{B}}$ we denote the family of all automata constructed before.

Example 4. Figures 2 and 3 present the intuition of the constructions implemented above. According to the above mentioned definitions of the beginning states of automata $\mathcal{A}, \mathcal{B}_1, \mathcal{B}_2$ are accordingly: s_1^0, r_1^0, r_2^0, whereas accepting ones are all states of the automata. In the case of automata $\overline{\mathcal{A}}, \overline{\mathcal{B}_1}, \overline{\mathcal{B}_2}$, beginning states are accordingly: s_1^3, r_1^2, r_2^2, whereas accepting ones are states $s_1^3, s_1^0, r_1^2, r_1^0, r_2^2, r_2^0$. One should notice that in the first network runs over the words: q_1, q_1q_2 and $q_1q_2q_4$ exist. In the second product automaton there are accordingly runs over the words: q_1, q_2q_1 and $q_4q_2q_1$.

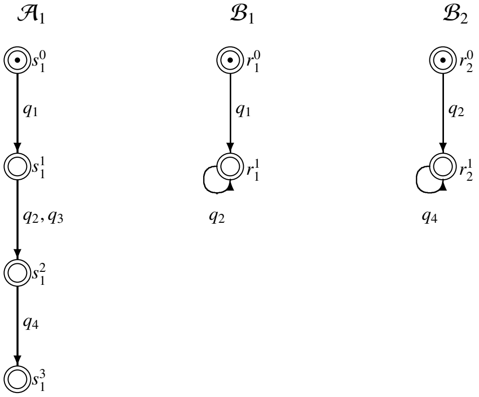


Fig. 2. Direct automata model

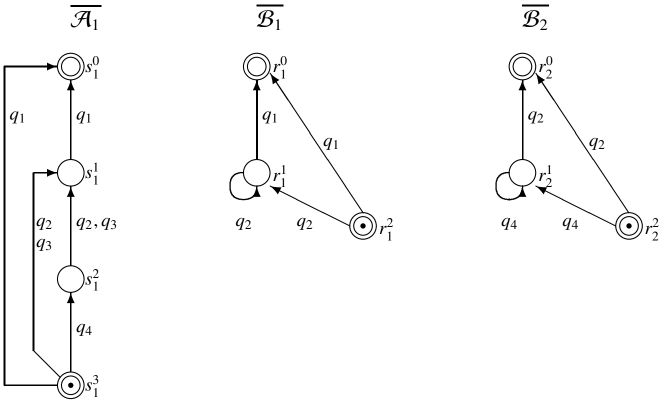


Fig. 3. Reverse automata model

One can see from the previous example that suitable accepting runs exist in the product automaton of the first type only if in the product automaton of the second type exist accepting runs which are mirror-reversed. The suitable example of network of automata that models reverse execution of NSPK is presented in Example 5.

Example 5. Figure 4 show the reverse automata model for execution of NSPK Protocol given in *Example 2*.

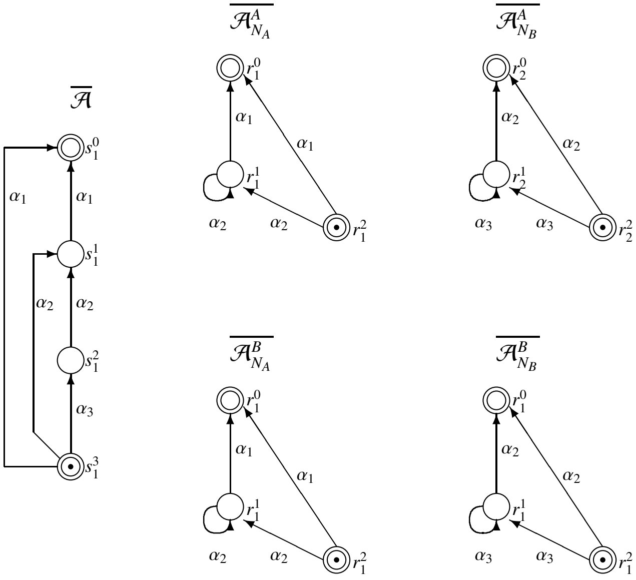


Fig. 4. Reverse automata model for NSPK execution

As done so previously, let's consider the product automaton $\overline{\mathfrak{AB}}$. For such constructed families of automata the following theorem holds.

Theorem 1. *In the product automaton* $\underline{\mathfrak{AB}}$ *there exists the run over the word* $q = q_1, q_2, \dots, q_w$ *iff in the product automaton* $\overline{\mathfrak{AB}}$ *there exists the run over the word* $\overline{q} = q_w, \dots, q_2, q_1$.

Proof of Theorem 1 proceeds by induction considering the length of the word q.

Proof of the right implication. In order to prove the inductive base, one should notice that if in product automaton $\underline{\mathfrak{AB}}$ there exists a run over the word $q = q_1$, then there exists exactly one automaton from the family $\mathfrak{A}$ in which the transition labelled q_1 is the transition from the beginning state to its successor. Therefore it is fulfilled that for a given $i \in I$ we have $(s_i^0, q_1, s_i^1) \in \delta_i$. According to the definition of the automaton $\overline{\mathfrak{AB}}$ in automaton $\overline{\mathcal{A}_i}$ there exists the transition $(s_i^t, q_1, s_i^0) \in \overline{\delta_i}$. It is noticeable that according to the definitions of knowledge automaton from the family $\underline{\mathfrak{AB}}$, in which any transition is labelled with label q_1, then there also exists an appropriate transition labelled q_1 leading from the beginning state r_j^2 to state r_j^0. For automata where in their transitions the label

$\underline{q_1}$ does not appear, an apropriate definition of ending states of the product automaton $\mathfrak{AB}$ should be reminded.

In order to prove the inductive step, it is assumed that the right implication is true for words of length k, therefore it is true that if in the product automaton $\underline{\mathfrak{AB}}$ there is a run over the word $q = q_1, q_2, \ldots, q_k$, then in the product automaton $\overline{\mathfrak{AB}}$ there is a run over the word $\overline{q} = q_k, \ldots, q_2, q_1$. It now should be proved that the implication is true for words of length $k + 1$.

Any given word $q = q_1, q_2, \ldots, q_k, q_{k+1}$ should now be considered. According to the definition of the product automaton $\underline{\mathfrak{AB}}$ there exists exactly one automaton from the family $\mathfrak{A}$ in which there exists a transition labelled q_{k+1}. So it is fulfilled that for a given $i \in I, (s_i^j, q_{k+1}, s_i^{j+1}) \in \delta_i$. According to the definition of automaton $\overline{\mathfrak{AB}}$ in automaton $\overline{\mathcal{A}_i}$ there exists a transition $(s_i^t, q_1, s_i^j) \in \overline{\delta_i}$. It is also noticeable that according to definitions of knowledge automata from the family $\underline{\mathfrak{AB}}$, in which any transition is labelled with label q_{k+1}, there also exists an appropriate transition labelled q_{k+1}. Proof of left implication is analogous.

It should also be noticed that the same property concerns the accepting runs. The reason for this results directly from the above statement and in the definition of accepting states of the components of families of both types.

4 Backward Induction in Modeling and Verification of Timed Protocols

Just like in the previous section, a model automata of time-dependent protocol executions which allows verification using the backward induction method will now be presented. Because the knowledge automata in this chapter do not include time aspects, only the construction of timed automata representing protocol execution will be introduced.

Let C be a set of time constraints defined in [19, 18]. Aditionally let $\mathfrak{TA}$ be a family of timed automata $\mathcal{TA}_i$ ($i \in I$ for some indices from I) that meet the following conditions.

Let $\mathcal{TA}_i = \{Q_i, \varSigma_i, \delta_i, s_i^0, F_i, \mathcal{X}_i\}$ be a timed automaton, where:

- let $Q_i = \{s_i^0, s_i^1, \ldots, s_i^t\}$, for all $i \in I$ and some $t \in N$ be a set of states in $\mathcal{A}_i$,
- let $\varSigma_i = \{k_i^1, k_i^2, \ldots, k_i^s\}$ be the set of labels where $\mathfrak{A}$ ($\varSigma_i \cap \varSigma_j = \emptyset$, dla $i \neq j$),
- $\delta_i \subseteq Q_i \times \varSigma_i \times C \times 2^{\mathcal{X}_i} \times Q_i$ be a transition relation that satisfies:
    - $(s_i^l, k, \mathfrak{cc}, X, s_i^j) \in \delta_i$ iff $l = j - 1$, for all $l = 0, 1, \ldots, s - 1$ and $j = 1, 2, \ldots, s$,
    - if $(s_i^l, k, \mathfrak{cc}, X, s_i^{l+1}) \in \delta_i$ and $(s_i^p, k, \mathfrak{cc}, X, s_i^{p+1}) \in \delta_i$, then $l = p$.
- s_i^0 is an initial state,
- $F_i = Q_i$ is the set of accepting states,
- $\mathcal{X}_i$ is the set of clocks.

This family defines automata which have the same structure as those defined in [19, 18] timed automata modeling the execution of time-dependent protocols. A timed automaton modeling reverse protocol execution will now be constructed.

For each automaton $\mathcal{TA}_i \in \mathfrak{A}$ let $\overline{\mathcal{TA}_i}$ be a smallest automaton that satisfies the following conditions:
$\overline{\mathcal{A}_i} = \{Q_i, \Sigma_i, \overline{\delta_i}, s_i^t, \{s_i^t, s_i^0\}, \mathcal{X}_i\}$ where:

- $\overline{\delta_i} \subseteq Q_i \times \Sigma_i \times C \times 2^{\mathcal{X}_i} \times Q_i$ be a transition relation that satisfies: if $(s_i^l, k, \mathfrak{cc}, X, s_i^{l+1}) \in \delta_i$, then $(s_i^{l+1}, k, \mathfrak{cc}, X, s_i^l) \in \overline{\delta_i}$ and $(s_i^{l+1}, k, \mathfrak{cc}, X, s_i^0) \in \overline{\delta_i}$.
- s_i^t is an initial state,
- $\{s_i^t, s_i^0\}$ is a set of accepting states.

In the case presented below, there holds a proper and adequate theorem concerning the existence of accepting runs in product automata including families execution automata as well as knowledge automata. The proof is analogous like in the previous one. For it should be noticed that only execution automata include nonempty time constraints, and these are accordingly modeled in a reverse product automata.

5 Experimental Results

In this section the results of experiments performed according to methodology discussed above will be presented and described. The results were obtained with the original implementation done in C++. For testing we used untimed and timed versions of four of well known protocols: NSPK (mentioned above), Lowe's version of NSPK (NSPKL), Wide Mouth Frog (WMF), and CCITT. The implementation generates for a given (in ProToc language [17, 18]) protocol and defined in it the space of considerations, different interpretations - executions of the tested protocol. These executions, in turn, are automatically translated to the network of untimed or timed automata. Then automata are translated to the boolean propositional formula in CNF (conjunction normal form) format. Formulas are optimized and tested using SAT-solver MiniSAT. The existence of a valuation satisfying the formula is equivalent to the existence of an attack on the protocol. Lack of satisfying valuation shows that there was no attack in the considered finite space. The module enables testing the previously described attacks. The performed calculations found the known attacks upon protocols. The tables presented below mainly focus on verification parameters: memory used by the CPU and computation time.

The verification method that uses the backward induction approach gave better results in the case of the WMF and CCITT protocols in both cases of untimed and timed versions. This may be caused by a small number of protocol steps, resulting in slight differences in the constructed model and the size of the formula. In the case of protocols with a larger number of steps automata model was a little biger and encoding gave larger formula. In this case searching by the solver took a bit more time than in the direct method.

Table 1 and 2 show a comparison of the direct verification and the backward induction technique in the case of untimed and time dependent protocols. After conducting a series of calculations for these protocols and their analyzis, the following conclusions can be made: the backward induction method gave somewhat better results in the case of protocols with a small number of steps. In the case of long protocols the method did not quicken obtained verification results.

Table 1. Comparison of direct and backward methods of untimed protocols verification

	Direct	Direct	Backward	Backward
Protocol	Memory (MB)	Time (s.)	Memory (MB)	Time (s.)
WMF$_{Untimed}$	8,56	0,004	7,78	0,002
CCITT$_{Untimed}$	9,13	0,041	8,84	0,031
NSPK	9,57	0,102	13,5	0,137
NSPKL	11,3	0,156	15,61	0,171

Table 2. Comparison of direct and backward methods for time dependent protocols

	Direct	Direct	Backward	Backward
Protocol	Memory (MB)	Time (s.)	Memory (MB)	Time (s.)
WMF	10,32	0,004	9,41	0,002
CCITT	10,89	0,063	10,24	0,045
NSPK$_{Timed}$	11,56	0,110	16,32	0,166
NSPKL$_{Timed}$	13,72	0,210	18,77	0,231

6 Summary

Research carried out by various international teams show that there is no single, "ideal" method of verification for all protocols. Often we must select the method for the given protocol. During the searching for the next models and methods we developed and implemented the backward induction method for constructing automata that model executions of security protocols. After a series of computation and their analysis we can draw the following conclusions: the backward induction method in the case of automata based modeling of executions and verification of security protocols due to methodology given in [17, 18] may give better results only in the case of protocols with a small number of steps and there it can be successfully used. For long protocol method does not speed up the computations.

References

1. Amnell, T., et al.: UPPAAL - Now, Next, and Future. In: Cassez, F., Jard, C., Rozoy, B., Dermot, M. (eds.) MOVEP 2000. LNCS, vol. 2067, pp. 99–124. Springer, Heidelberg (2001)
2. Armando, A., et al.: The AVISPA Tool for the Automated Validation of Internet Security Protocols and Applications. In: Etessami, K., Rajamani, S.K. (eds.) CAV 2005. LNCS, vol. 3576, pp. 281–285. Springer, Heidelberg (2005)
3. Armando, A., Compagna, L.: Sat-based model-checking for security protocols analysis. International Journal of Information Security 7(1), 3–32 (2008)
4. Bella, G., Massacci, F., Paulson, L.C.: Verifying the set registration protocols. IEEE Journal on Selected Areas in Communications 20(1), 77–87 (2003)
5. Bella, G., Paulson, L.C.: Using Isabelle to prove properties of the kerberos authentication system. In: Orman, H., Meadows, C. (eds.) Proc. of the DIMACS Workshop on Design and Formal Verification of Security Protocols (1997)
6. Burrows, M., Abadi, M., Needham, R.M.: A logic of authentication. ACM Trans. Comput. Syst. 8(1), 18–36 (1990)
7. Cohen, E.: TAPS: A first-order verifier for cryptographic protocols. In: CSFW 2000: Proceedings of the 13th IEEE Computer Security Foundations Workshop (CSFW 2000), p. 144. IEEE Computer Society, Washington, DC (2000)

8. Corin, R., Etalle, S., Hartel, P.H., Mader, A.: Timed model checking of security protocols. In: Proc. of the 2004 ACM Workshop FMSE, pp. 23–32. ACM (2004)
9. Delzanno, G., Ganty, P.: Automatic verification of time sensitive cryptographic protocols. In: Jensen, K., Podelski, A. (eds.) TACAS 2004. LNCS, vol. 2988, pp. 342–356. Springer, Heidelberg (2004)
10. Dolev, D., Yao, A.: On the security of public key protocols. IEEE Transactions on Information Theory 29(2), 198–207 (1983)
11. El Fray, I., Kurkowski, M., Pejaś, J., Maćków, W.: A New Mathematical Model for the Analytical Risk Assessment and Prediction in IT Systems. Control & Cybernetics 41, 241–268 (2012); Systems Research Institute PAS, Warsaw
12. Evans, N., Schneider, S.: Analysing time dependent security properties in CSP using PVS. In: Cuppens, F., Deswarte, Y., Gollmann, D., Waidner, M. (eds.) ESORICS 2000. LNCS, vol. 1895. Springer, Heidelberg (2000)
13. Jakubowska, G., Penczek, W., Srebrny, M.: Verifying security protocols with timestamps via translation to timed automata. In: Proc. of the International Workshop on Concurrency, Specification and Programming (CS&P 2005), pp. 100–115. Warsaw University (2005)
14. Kacprzak, M., et al.: Verics 2007 - a model checker for knowledge and real-time. Fundam. Inform. 85(1-4), 313–328 (2008)
15. Kurkowski, M., Maćków, W.: Using Backward Strategy to the Needham-Schroeder Public Key Protocol Verification. In: Artificial Intelligence and Security in Computing Systems, pp. 249–260. Kluwer Academic Publishers, Boston (2003)
16. Kurkowski, M., Pejaś, J.: A Propositional Logic for Access Control Policy in Distributed Systems. In: Artificial Intelligence and Security in Computing Systems, pp. 157–191. Kluwer Academic Publishers, Boston (2003)
17. Kurkowski, M., Penczek, W.: Verifying Security Protocols Modeled by Networks of Automata. Fund. Inform. 79(3-4), 453–471 (2007)
18. Kurkowski, M., Penczek, W.: Verifying Timed Security Protocols via Translation to Timed Automata. Fund. Inform. 93(1-3), 245–259 (2009)
19. Kurkowski, M., Penczek, W.: Applying Timed Automata to Model Checking of Security Protocols. In: Wang, J. (ed.) Handbook of Finite State Based Models and Applications, pp. 223–254. Chapman&Hall/CRC Press, Boca Raton (2012)
20. Kurkowski, M., Srebrny, M.: A Quantifier-free First-order Knowledge Logic of Authentication. Fund. Inform. 72, 263–282 (2006)
21. Lowe, G.: Breaking and Fixing the Needham-Schroeder Public-key Protocol Using fdr. In: Margaria, T., Steffen, B. (eds.) TACAS 1996. LNCS, vol. 1055, pp. 147–166. Springer, Heidelberg (1996)
22. Lowe, G.: Some new attacks upon security protocols. In: Proceedings of the Computer Security Foundations Workshop VIII. IEEE Computer Society Press (1996)
23. Meadows, C.: The NRL protocol analyzer: An overview. Journal of Logic Programming 26(2), 13–131 (1996)
24. Moore, J.H.: Protocol failures in cryptosystems. Proceedings of the IEEE 76(5) (1988)
25. Needham, R.M., Schroeder, M.D.: Using encryption for authentication in large networks of computers. Commun. ACM 21(12), 993–999 (1978)
26. Paulson, L.C.: Inductive analysis of the internet protocol tls. ACM Trans. Inf. Syst. Secur. 2(3), 332–351 (1999)
27. Siedlecka-Lamch, O., Kurkowski, M., Piech, H.: A New Effective Approach for Modeling and Verification of Security Protocols. In: Proceedings of 21st International Workshop on Concurrency, Specification and Programming (CS&P 2012), pp. 191–202. Humboldt University Press, Berlin (2012)

Telecommunications Networks Risk Assessment with Bayesian Networks

Marcin Szpyrka[1], Bartosz Jasiul[2], Konrad Wrona[3], and Filip Dziedzic[1]

[1] AGH University of Science and Technology
Department of Applied Computer Science
Al. Mickiewicza 30, 30-059 Kraków, Poland
mszpyrka@agh.edu.pl, filipdz@student.agh.edu.pl

[2] Military Communication Institute
ul. Warszawska 22a, 05-130 Zegrze, Poland
b.jasiul@wil.waw.pl

[3] NATO Communications and Information Agency
Oude Waalsdorperweg 61, 2597AK Den Haag, The Netherlands
Konrad.Wrona@ncia.nato.int

Abstract. We propose a solution which provides a system operator with valuation of security risk introduced by various components of the communication and information system. This risk signature of the system enables the operator to make an informed decision about which network elements shall be used in order to provide a service requested by the user while minimising security risk related to service execution. In considered scenario transmitted data can be intercepted, modified or dropped by an attacker. Each network component and path can be potentially used to compromise information, since an adversary is able to utilise various vulnerabilities of network elements in order to perform an attack. The impact and probability of such successful attacks can be assessed by analysing the severity of the vulnerabilities and the difficulty of exploiting them, including the required equipment and knowledge. In consequence, each possible service work-flow can be assigned a security risk signature.

Keywords: telecommunications networks, risk assessment, Bayesian networks.

1 Introduction

Modern network infrastructure enables transport of information between a sender and a receiver by using different paths composed of multiple network links and nodes. It is a heterogeneous environment which involves various protocols, operating systems, software and hardware [30]. Moreover, a network consists of separate administrative domains, managed by different authorities. In such complex environment software bugs, protocol flaws, non-installed patches, obsolete network components, etc. introduce a risk of unauthorised disclosure or modification of transmitted information.

The diversity of telecommunication devices and protocols must be taken into account when sensitive information is transmitted. It is important not only to protect information by cryptographic measures including confidentiality, integrity and authenticity, but to take into account availability of the information as well. The operator must be made

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 277–288, 2013.

aware of potential risk of information being compromised when a specific distribution path is chosen. Naive solution consisting of stopping information exchange in case of any non-negligible risk is not viable because existence of any, even smallest, applicable and not mitigated threat would lead to stopping all communication. The solution proposed in this paper focuses on delivering a measurable indicator of risk, so-called risk signature. The risk signature takes the form of quantitative values that can be compared during the decision process, thus offering an advantage over qualitative risk indicators, such as low, medium and high.

2 Motivation

The work presented in this paper has been motivated by challenges related to development of the Information Exchange Gateway (IEG) concept. The North Atlantic Treaty Organization (NATO) and national forces have defined the concept of an IEG to facilitate secure communications between different security and management domains [11]. Related to IEG is the concept of Content-based Protection and Release (CPR) [36]. CPR aims to improve timely sharing of information in the NATO Network Enabled Capability (NNEC) and the Future Mission Network (FMN) [15] environments. Potential implementation scenario for the IEGs using CPR policies is communication between NATO domains and non-NATO organisations, such as United Nations, International Committee of the Red Cross and other non-government organisations.

In current operations timely sharing of information is hampered due to a number of limitations that are inherent to the traditional use of security markings. CPR overcomes these limitations by enforcing access control based on the content properties of an information object instead of a security marking. In CPR the decision to release information object to a user is based on a protection and release policy that expresses requirements the user and the operational environment (e.g. user's terminal) must meet in order to access an information object with a given set of content properties. The requirements are translated to user and terminal attributes; as such CPR is an extension to Attribute Based Access Control (ABAC) [16].

Of particular importance for the approach presented in our contribution is that the CPR policy can also take into account specific environmental attributes such as the risk signature of the network. This enable CPR to provide risk-based adaptive access control. In our proof-of-concept implementation based on XACML 3.0 architecture [27] such risk signature can be obtained by the Policy Information Point from a Dynamic Risk Assessment system [20]. Current IEG solution assumes that surroundings of the protected domain are hostile and their the main source of threats to the information exchange. However the state of the protected domain can also influence a level of risk involved in information exchange process. For example, if some software used on nodes of protected domain has some unpatched vulnerabilities, a communication channel between untrusted domain and protected domain may introduce a possible attack path for outsider attacker. Ability to perform such attack may depend on IEG accepting mediation of particular content type or protocols. If one cannot eliminate risk, in order to control the flow of information between the domains the level of risk should influence the decision whether the information will be sent or dropped. Thus IEG should be able to selectively and dynamically modify its release policy, depending on a level of risk existing in the internal system. In this paper we present an approach for dynamic risk assessment which can be used in order to support risk-adaptable access control [23].

3 Related Work

The presented approach is in accord with recent research in the network security area, where a number of approaches (including formal ones) to the analysis and design of security systems has been proposed. For example, a formal graph model of a computer network with a variable topology is consider in [22]. The model is used for the verification of security constraints. An extension of attack graphs called multiple-prerequisite graphs is considered in [17]. A computer software based on it is used to classify vulnerabilities and recommend actions to improve network security. Anticipation game framework with the so-called strategy objectives is considered in [13]. This extension of attack graphs is used to coupe with both the financial and temporal aspects of attacks. Some authors focuses on formal approach to the design of network elements e.g. firewalls. For example, the design of a network firewall with a formalised rule-based framework called XTT2 [26] is proposed in [25]. Another rule-based approach that uses RTCP-nets [31] is considered in [32] and [34]. Moreover, exclusion rule-based systems are proposed in [33]. Due to such frameworks possible anomalies in the security policy can be eliminated during the design stage.

There are several methodologies used to asses the risk. One of the most popular is CCTA Risk Analysis and Management Method (CRAMM) [1], which is used in the UK, Denmark and Czech Republic. NATO uses a modified version of CRAMM, compliant with NATO security policy, supporting directives and guidance. CRAMM analyses vulnerabilities of the system and potential threats that may have impact on loss of assets and functionality. It is a qualitative methodology that covers identified aspects that may affect the risk: personnel security, physical security and security of information. It utilises a large database with detailed questions and it is compliant with ISO/IEC standard [18]. A week point of this technique is a lack of possibility to analyse algorithm used to calculate the risk.

Pilar [3] is a risk assessment tool based on Methodology for Information Systems Risk Analysis and Management (MAGERIT) [24] developed in Spain. This methodology provides a qualitative risk assessment. Risk is represented as the impact of the threat weighted by its rate (or expectation) of occurrence. A customised version of Pilar is used also within NATO.

The Expression of Needs and Identification of Security Objectives (EBIOS) [9] methodology has been created by the French National Security Agency. The methodology covers 5 steps: context, security needs, threats analysis, identification of security objectives and identification of security requirements. It allows to assess, communicate, and choose appropriate mitigation measures for risks related to information security.

Method for Harmonized Analysis of Risk (MEHARI) is another French information risk assessment method, developed by association of information security professionals and designed for an analysis of risk situations described through scenarios [4].

The Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE) method [10] is a framework for identifying and managing information security risks developed by Computer Emergency Response Team. It also is a suite of tools, techniques, and methods for risk-based information security strategic assessment and planning. The method is used to: (i) identify critical assets and the threats to those assets, (ii) identify the vulnerabilities, both organisational and technological, that expose those threats, creating risk to the organisation, and (iii) develop a protection strategy and risk mitigation plans to support the organisation's mission and priorities.

4 Case Study

As an illustrative example for the proposed approach and calculation of risk we will use hypothetical protected network shown in Fig. 1. The network provides an access to confidential data by using two redundant database servers. The only way to collect the data is to send a request to one of the two web servers. A web server sends a request to a database server and then provides the user with the received answer. A firewall is used to isolate the network from the Internet. The security policy allows users from other domains to send requests to the services located in the protected domain. Due to different transmission media, the possible access routes differ in the level of security [35].

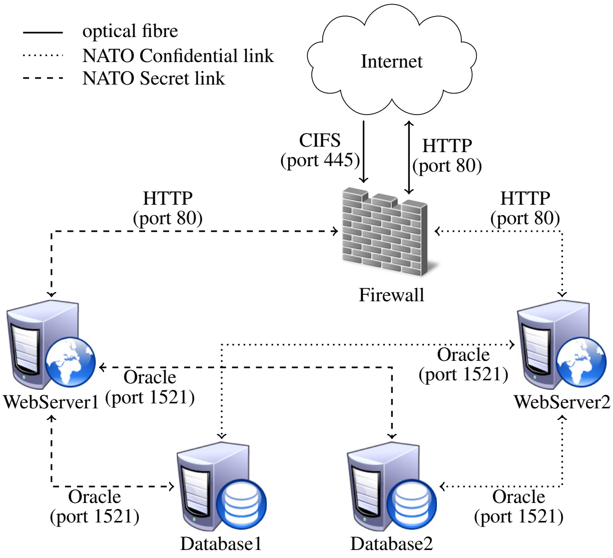


Fig. 1. Network topology

While estimating the risk of information retrieval, it is necessary to take under consideration different confidentiality protection levels for the used transmission media. Moreover, there are additional factors affecting the security of depicted network. These are various services offered in the domain and software installed on devices. Usually, the more equipment, protocols and software are used in the network, the more potential vulnerabilities are present. These vulnerabilities might be used for taking control of either a device or a service and can lead to revealing sensitive information to adversary.

Scenario presented in Fig. 1 is partially related to current NATO works on providing information exchange capability for information domains with different classification levels, e.g., NATO Secret, NATO Confidential, and Mission Secret. In the past military networks with different confidentiality level had to be completely separated. Currently, in order to facilitate information sharing, collaboration and reduce duplication of data, there is a trend to provide controlled and secured interconnection of networks operating on different security levels. Risk assessment in this case is identified as a crucial element in taking decision whether to send (or not) the information.

5 Proposed Solution

The proposed risk analysis method is composed of two main components. The first component (see: box 1 in Fig. 2) is dedicated to evaluation of a risk level of individual network elements and the network as whole. The second one (see: box 2 in Fig. 2) is to assess the influence of the risk introduce by the individual network elements involved in execution of a service offered to the end user on the total risk signature of this service.

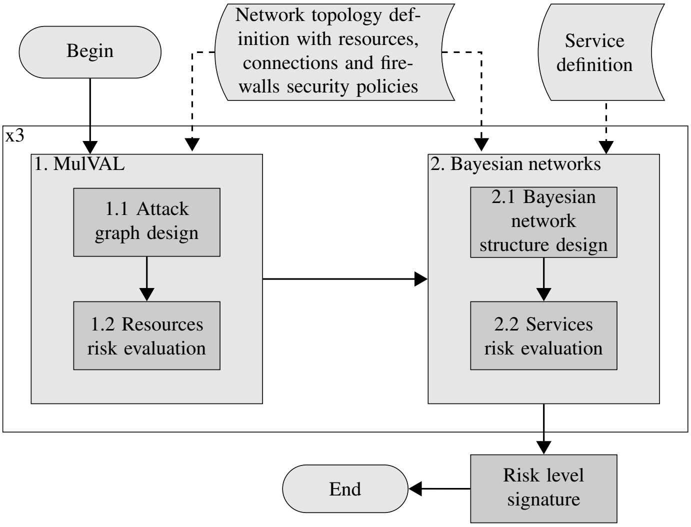

Begin
Network topology definition with resources, connections and firewalls security policies
Service definition
x3
1. MulVAL
1.1 Attack graph design
1.2 Resources risk evaluation
2. Bayesian networks
2.1 Bayesian network structure design
2.2 Services risk evaluation
Risk level signature
End

Fig. 2. The proposed approach scheme

The first part of risk assessment method is based on MulVAL tool [28] (see also Sec. 5.2). Attack trees [29] were chosen to assess likelihood of successful attacks on assets vulnerabilities. Attack trees (see Sec. 3) provide a formal, methodical way of describing structure of attack on specific goal. Precisely, the root node represents the object of an attack and leafs are the steps that must be done to achieve the goal. For instance, if a malicious user of a corporate network wants to read a document stored at his supervisor's computer, he needs to get access to this computer. He can trick the supervisor into running specially prepared code that will send documents to him via e-mail (social engineering attack). Alternatively, he can brake into supervisor's office, then start the computer and guess the password. At last, he can recognise the infrastructure of the network, identify the computer he wants to take control over, compromise the personal firewall, which was not updated, and inject a code that will allow him to download the document by the FTP protocol.

The second part of our method takes into account that the specific services provided to the users usually rely only on a subset of all network elements. Therefore, the fact that some high risks elements are present in the network does not necessarily mean that the particular service is exposed to high risk as well. For example, the fact that an email server is exposed in a specific network configuration to a high security risk might be irrelevant for a service which provides end user with an access to geographic information stored in a database. Thus, in order to provide a meaningful assessment of risk level which a particular service is exposed to and therefore estimate risk which is transferred to the end user and his operations the risk assessment has to take into account a specific set of network links and nodes which are utilised by the service [35]. Every service can be described as a work-flow, involving different network elements, such as servers, routers, and links. We model the dependencies between steps of the work-flow as Bayesian network (also called belief network) [12,14] (see also Sec. 5.3).

5.1 System Description

In the example considered in Sec. 4, it is assumed that the detailed network topology with its resources and connections among them must be described and provided to MulVAL tool. Vulnerabilities of network elements, software and protocols are obtained from National Vulnerability Database (NVD) [6]. This database, widely used by various security vendors and computer emergency response teams, includes information on specific threats. A record from NVD, called CVE (*Common Vulnerabilities and Exposures* [2]), contains:

- CVE Identifier (eg. CVE-2012-5689),
- description (*short information on vulnerability*),
- asset impacted (*some software, protocol*),
- version infected,
- severity,
- impact,
- score (*value representing potential harmfulness of vulnerability*).

5.2 MulVAL

The system description is used by MulVAL tool to calculate the risk of every affected network element. In fact, MulVAL is used three times in order to assess the risk for confidentiality (C), integrity (I) and availability (A). MulVAL (Multi-host, Multi-stage Vulnerability Analysis Language) [28] is a research tool used to manage the configuration of an enterprise network such that the security risks are appropriately controlled. MulVAL provides a reasoning engine for automatically identifying vulnerabilities in an enterprise network. It uses Datalog facts to represent configuration information and

Datalog rules to represent attack techniques and OS security semantics. After analysing the configuration of a network, the MulVAL reasoning engine outputs a logical attack graph. An example of such a graph (borrowed from [5]) is shown in Fig. 3.

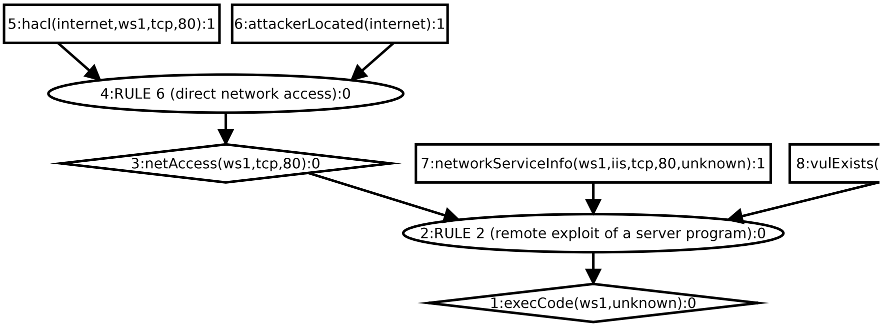

5:hacl(internet,ws1,tcp,80):1
6:attackerLocated(internet):1
4:RULE 6 (direct network access):0
3:netAccess(ws1,tcp,80):0
7:networkServiceInfo(ws1,iis,tcp,80,unknown):1
8:vulExists(
2:RULE 2 (remote exploit of a server program):0
1:execCode(ws1,unknown):0

Fig. 3. Example of MulVAL attack tree [5]

A logical attack graph directly encodes the logical causality relationship among configuration settings and potential attacker privileges. Its key goal is to show *why an attack can happen*. There are three kinds of vertices in such a graph: rectangle vertices represent facts, elliptic vertices represent reasoning rules and diamond vertices represent privileges an attacker can obtain through exploiting the vulnerabilities in the considered system. The part of a logical attack graph shown in Fig. 3 refers to estimating the probability of a successful database server attack.

At the first glance, the value of the risk obtained from MulVAL can be perceived as sufficient. In fact, this method delivers assessment of risk for the network. However, the obtained value does not take into account that some network components might be irrelevant for execution of a particular service, thus even if they are high risk assets, they may not have much influence on risk level for delivery of the service to the end user. In fact, there can be various routes more often used to exchange of information. Thus, additional method is required in order to amend the value obtained from the first assessment and to present the actual risk for the particular service or mission objective supported by the system.

5.3 Bayesian Networks

Bayesian networks [19] were identified as a suitable method to accomplish this goal. Bayesian networks allow for inverse representations of the probabilities concerning two events. For instance, event B is a consequence of event A, what can be described as $A \to B$. It means that A is the reason for the consequence B. However, the question often is: *What is the probability of* A *if* B *was observed?* The answer can be calculated from the statement of Bayes' theory:

$$P(A|B) = \frac{P(A \cup B)}{P(B)} = \frac{P(B|A)P(A)}{P(B)}, \text{ where } P(B) > 0 \tag{1}$$

As noted earlier, we model the dependencies between steps of the service workflow as Bayesian network. The network elements are used by the service in a specific order which has to be captured in the model. Some of the elements might appear in these workflows several times (e.g. a particular link can be used for transmitting both a request to the database and an answer to the query). Similarly, the specific steps of the service workflow can be implemented by using several different network elements. For example, there might be several independent communication links available between a Web server and a database and the database might be replicated on several separate servers for sake of performance and redundancy. In order to deal with possible cycles, it was necessary to represent some of the nodes multiple times in the resulting Bayesian network. However, this multiplication does not interfere with the calculation of the final risk values, as the risk introduced by the particular network element is taken into account only once in the overall calculation. Every network element is assigned three independent risk values related to confidentiality, integrity and availability. For the network nodes, these risk values are calculated by using the information about relevant vulnerabilities, provided in the NVD, and known applicable attacks modeled using MulVal methodology. For the network links, a simplified model has been used, assigning static risk values in the C-I-A dimensions, depending on the characteristic of the links.

The reasoning in the Bayesian network describing service dependencies has been performed by clustering algorithm as described in [21]. This algorithm consists of two stages. In the first stage directed graph representing dependencies between vulnerabilities and attacks is decomposed to a tree. In the second stage the appropriate inference on the obtained tree is conducted. The overall risk level is calculated from a perspective of the user accessing the service and consists of three values, describing risk to confidentiality, integrity and availability of the service. The results and the proof of concept are shown in Sec. 6.

6 Proof of Concept

In order to validate the proposed approach in practice, a prototype software called *Network risk assessment tool* has been developed. The application uses a client-server architecture, with a front-end based on a Web browser supporting JavaScript. The software has been developed by using Java and Play Framework [7]. All the calculations for Bayesian networks were performed with the aid of the SMILE library [8] and for Bayesian networks visualization the GeNIe development environment has been used. The back-end application uses the PostgreSQL database server to provide all data which is necessary for risk calculations.

The developed tool takes under consideration different network devices such as hosts, servers etc. and connections among them. The MulVAL software enable us to take into consideration firewalls configuration and dependencies among resources. Dynamically constructed Bayesian networks make it possible to assess the risk of functioning of any particular service. In order to provide reliable results, we have used real information about vulnerabilities obtained from the US National Vulnerability Database [6].

The application has been tested successfully. One of the test cases based on the network shown in Fig. 1 is presented below. Suppose, the possible communication routes

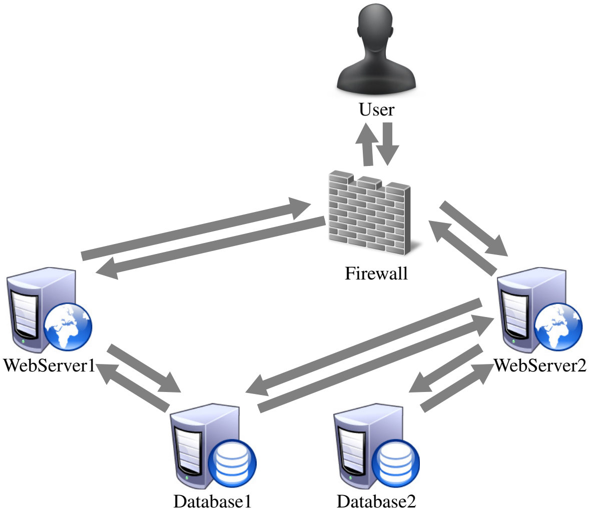


Fig. 4. Possible communication routes

Table 1. Description of the test case scenario

No	Step description	Risk		
		C	I	A
1	Vulnerability: non-identified. Risk assessed on the basis of historical data.	0.07108	0.07108	0.01783
2	Vulnerability: HTTP IIS Server at WebServer 1, monitoring of port 80 TCP. The vulnerable configuration is seen very rarely in practice (difficult utilization). CVE Access Complexity: High. Result: Insignificant increase of confidentiality risk	0.08448	0.07108	0.01783
3	Vulnerability: none - exclusion of HTTP IIS Server vulnerability from point 2. Risk reset to initial values from Step 1.	0.07108	0.07108	0.01783
4	Vulnerability: HTTP IIS Server misconfiguration at Web Server 1. CVE Access Complexity: Medium. Accessible for medium-skilled intruders. Significant change of risk for confidentiality	0.34257	0.07108	0.01783
5	Vulnerability: new HTTP infection at WebServer 2. CVE Access Complexity: Medium. Vulnerability affects all security measures.	0.36549	0.19174	0.02558
6	Vulnerabilities: 1) new infection at Database 1; 2) software misconfiguration at whole network. CVE Access Complexity: Medium. Vulnerability no. 1 affects integrity and availability. Vulnerability no. 2 affects all security measures. Risk: significant increase of all values. Due to redundant routes information is still available, however it is exposed to disclosure and modification.	0.36549	0.25429	0.22009

are defined as shown in Fig. 4. In such a case, we have redundant routes used to access data stored in the database servers. The aim of the computation is to assess the risk for the service that guarantee access to the data.

The redundant routes decrease the risk of loss of access to the data significantly. On the other hand, the greater the number of devices in a network is, the greater the risk of data being compromised. It is a result of potential vulnerabilities introduced by every additional node or link. These vulnerabilities might be exploited by the adversary to get an access to information or to disrupt communication.

The considered test scenario illustrates also the impact of vulnerabilities on the risk for three security dimensions: confidentiality, integrity and availability. Different vulnerabilities have different influence on the assessed risk signature. This is due to variations in difficulty of vulnerability exploitation, harmfulness, impact on the whole communication, etc. As a part of validation scenarios, we have also simulated that some vulnerabilities were detected and the related risks mitigated, in order to observe how this process influences the overall risk signatures. Values of estimated risk are shown in Table. 1.

7 Conclusions

The approach presented in this paper is the first step to assess the risk of compromising services and information provided by a heterogeneous networking environment. Presented solution assesses the risk derived from vulnerabilities of both network links and network nodes, including the vulnerabilities introduced by software and hardware configuration. Current implementation takes into account network elements that are vulnerable to attacks resulting in leakage, modification or unavailability of transmitted data as well as static technical security countermeasures, which can be used to reduce the applicable attack surface. One possible extension of our model is the possibility of modeling of dynamic deployment of technical security countermeasures that reduce risk. Such technical countermeasures include, e.g. intrusion detection systems, network guards and dynamically changing security policy. Obviously, it is impossible to fully exclude the risk involved in the operation of telecommunication networks due to various malware, vulnerabilities of operating systems and network components. To imagine the number of threats, one can observe how often the software (including anti-virus tools) installed on his machine are updated. Therefore information stakeholders must deal with the risk and they need to know the risk signature of their system. The proposed solution provides them with a risk valuation, which can be used by network nodes to find the best route for information exchange and can help network administrators to take appropriate decision about which network links should be considered safe or risky.

The novelty of our solution is in two-step risk assessment. The first step relies on evaluation of attack graphs and uses vulnerabilities description (based on NVD) and network configuration information to assess risk signature for individual network assets. Our approach take into account both risk self-induced by particular assets and the risk induced by the operational environment. The second step involves calculation of risk signature for a particular service offered by the system to end users. Calculation of risk is performed on the basis of Bayesian networks, which allow us not only to obtain a detailed risk signature, but also to analyze reasons and consequences of risk. Our approach is comprehensive and takes into account all system components and network communication links required to provide user with a service. The calculated risk signature covers all three dimensions of security, i.e. confidentiality, integrity and availability. Our solution is more of a quantitative than a qualitative methodology when compared to some of the widely used methods, such as CRAMM or Pilar. Thus, we believe that our proposal offers broader and more detailed approach to risk assessment for system objectives and user services.

Acknowledgements. Work has been partially financed by the European Regional Development Fund the Innovative Economy Operational Programme, INSIGMA project no. 01.01.02-00-062/09.

References

1. CCTA Risk Analysis and Management Method, `http://www.cramm.com/`
2. Common Vulnerabilities and Exposures, `http://cve.mitre.org/`
3. EAR/Pilar - Risk Analysis Environment, `https://www.ccn-cert.cni.es/`
4. MEHARI - Method for Harmonized Analysis of Risk, `http://www.clusif.asso.fr/`
5. MulVAL Attack Paths Engine, `http://forge.fi-ware.eu/plugins/mediawiki/wiki/fiware/index.php/MulVAL_At tack_Paths_Engine_-_User_and_Programmer_Guide`
6. National Vulnerability Database, `http://nvd.nist.gov/`
7. Play Framework, `http://www.playframework.org/`
8. SMILE Documentation, `http://genie.sis.pitt.edu/wiki/SMILE_Documentation`
9. Agence nationale de la sécurité des systèmes d'information: Expression des Besoins et Identification des Objectifs de Sécurité (EBIOS) (2010)
10. Alberts, C.J., Behrens, S.G., Pethia, R.D., Wilson, W.R.: Operationally Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE) Framework, version 1.0 (1999)
11. Apiecionek, Ł., Romantowski, M., Śliwa, J., Jasiul, B., Goniacz, R.: Safe exchange of information for civil-military operations. In: Military Communications and Information Technology: A Comprehensive Approach Enabler, pp. 39–50. WAT Publishing (2010)
12. Barber, D.: Bayesian Reasoning and Machine Learning. Cambrdge University Press (2013)
13. Bursztein, E., Mitchell, J.C.: Using strategy objectives for network security analysis. In: Bao, F., Yung, M., Lin, D., Jing, J. (eds.) Inscrypt 2009. LNCS, vol. 6151, pp. 337–349. Springer, Heidelberg (2010)
14. Darwiche, A.: Modeling and reasoning with Bayesian networks. Cambridge Univ. (2009)
15. Domingo, A., Wietgrefe, H.: A NNEC-compliant approach for a Future Mission Network. In: Proc. of the Military Communications Conference, MILCOM (2012)
16. Hu, V.C., Ferraiolo, D., Kuhn, R., Friedman, A.R., Lang, A.J., Cogdell, M.M., Schnitzer, A., Sandlin, K., Miller, R., Scarfone, K.: Guide to Attribute Based Access Control Definition and Considerations, Draft. NIST Special Publication 800-162, Gaithersburg (2013)
17. Ingols, K., Lippmann, R., Piwowarski, K.: Practical attack graph generation for network defense. In: Proc. of ACSAC Conf. 2006, pp. 121–130. IEEE Computer Society (2006)
18. ISO/IEC: ISO/IEC 27001:2005 Information technology – Security techniques – Information security management systems – Requirements (2008)
19. Kjaerulff, U., Madsen, A.: Bayesian Networks and Influence Diagrams: A Guide to Construction and Analysis. Springer (2008)
20. Lagadec, P., Dandurand, L., Bouillon, E., Wrona, K., Torrente, S.: Cyber Defence Situational Awareness and Dynamic Risk Assessment. In: NATO Research and Technology Organisation Symposium on Information Assurance and Cyber Defence, Tallin, Estonia (2010)
21. Lauritzen, S., Spiegelhalter, D.J.: Local computations with probabilities on graphical structures and their application to expert systems. Journal of the Royal Statistical Society series B 50, 157–224 (1988)

22. Matousek, P., Ráb, J., Rysavy, O., Svéda, M.: A Formal Model for Network-Wide Security Analysis. In: Proceedings of the 15th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems, pp. 171–181. IEEE Comp. Soc. (2008)
23. McGraw, R.: Risk-adaptable access control (radac). In: NIST Privilege (Access) Management Workshop (2009)
24. Ministerio de Administraciones Públicas: MAGERIT version 2, Methodology for Information Systems Risk Analysis and Management, Book I The Method (2006)
25. Nalepa, G.J., Ligęza, A.: Designing reliable Web security systems using rule-based systems approach. In: Menasalvas, E., Segovia, J., Szczepaniak, P.S. (eds.) AWIC 2003. LNCS (LNAI), vol. 2663, pp. 124–133. Springer, Heidelberg (2003)
26. Nalepa, G.J., Ligęza, A., Kaczor, K.: Formalization and modeling of rules using the XTT2 method. International Journal on Artificial Intelligence Tools 20(6), 1107–1125 (2011)
27. OASIS: eXtensible Access Control Markup Language ver. 3.0. Tech. Rep. (August 2010)
28. Ou, X., Govindavajhala, S., Appel, A.: MulVAL: A logic-based network security analyzer. In: Proc. of 14th USENIX Security Symposium, Baltimore, Maryland, USA (2005)
29. Schneier, B.: Attack trees: Modeling security threats. Dr. Dobbs' Journal (1999)
30. Sliwa, J., Gleba, K., Chmiel, W., Szwed, P., Glowacz, A.: IOEM - Ontology Engineering Methodology for Large Systems. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part I. LNCS, vol. 6922, pp. 602–611. Springer, Heidelberg (2011)
31. Szpyrka, M.: Analysis of VME-Bus communication protocol – RTCP-net approach. Real-Time Systems 35(1), 91–108 (2007)
32. Szpyrka, M.: Design and analysis of rule-based systems with Adder Designer. In: Cotta, C., Reich, S., Schaefer, R., Ligéza, A. (eds.) Knowledge-Driven Computing. SCI, vol. 102, pp. 255–271. Springer, Heidelberg (2008)
33. Szpyrka, M.: Exclusion rule-based systems – case study. In: International Multiconference on Computer Science and Information Technology, Wisła, Poland, vol. 3, pp. 237–242 (2008)
34. Szpyrka, M., Szmuc, T.: Decision tables in Petri net models. In: Kryszkiewicz, M., Peters, J.F., Rybiński, H., Skowron, A. (eds.) RSEISP 2007. LNCS (LNAI), vol. 4585, pp. 648–657. Springer, Heidelberg (2007)
35. Wrona, K., Hallingstad, G.: Real-time automated risk assessment in protected core networking. Telecommunication Systems 45(2-3), 205–214 (2010)
36. Wrona, K., Hallingstad, G.: Controlled information sharing in NATO operations. In: IEEE Military Communications Conference (MILCOM), pp. 1285–1290. IEEE (2011)

Power Aware Cluster Based Routing (PACBR) Protocol for Wireless Sensor Network

Ayan Kumar Das[1], Rituparna Chaki[2], and Atreyee Biswas[3]

[1] Department of Information Technology, Calcutta Institute of Engineering and Management,
Kolkata, India
ayandas24114057@yahoo.co.in

[2] A.K. Chowdhury School of IT, University of Calcutta,
Kolkata, India
rituchaki@gmail.com

[3] Department of Natural Science, West Bengal University of Technology,
Kolkata, India
atreyee11@gmail.com

Abstract. Energy efficiency is the main challenge of wireless sensor network (WSN). Many routing algorithms have been designed to meet the demand. Hierarchical clustering is one of the effective techniques for WSN to reduce the load in the network by data aggregation, thus saving energy. In this approach cluster heads take the responsibility to gather information of different events from its neighbors and send that to the sink node. Thus cluster heads loose its energy very early, which causes selection of new cluster head before starting of every round. This paper presents a multilevel cluster based protocol for energy efficient data communication. There is a cluster head selection phase followed by sensing of data, data gathering, aggregation and finally sending the aggregated data to the base station. Simulation results show that the performance of the proposed protocol is better than Energy Efficient Clustering Algorithm (EECA) for data aggregation in WSNs.

Keywords: Wireless Sensor Network, Data aggregation, Network Lifetime, Energy Efficiency, Cluster Head, Heterogeneous.

1 Introduction

A wireless sensor network consists of many small sensor nodes for sensing events in a particular area and sending that information to a sink node. The sensors are very small devices, with limited battery power, and often, with no source of recharge. This makes energy efficiency the main challenge for the researchers. Most of the power aware routing algorithms aim to find the best probable path to send the information between source and base station. Often, the sensor nodes are deployed very densely so that a number of nodes sense the same event. All these nodes try to send the redundant data to the sink node using multiple paths. This may cause a huge amount of energy drainage. Selection of a head node for a region or cluster can reduce the energy drainage in such situations. All the cluster members send the sensed data to the

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 289–300, 2013.

cluster head. The cluster head collects data from the members, reduce those redundant data and aggregate all the collected data. The aggregated data will be sent to the base station only by the cluster head, thus saving energy. The proposed algorithm selects a cluster head in different clusters before starting every round of aggregation and sends data to base station. The energy of the cluster head reduces, thus making the selection of new cluster head necessary after completion of every round. The energy consumption for communication between two nodes is governed by the Inverse Square Law of energy. Further the network is divided into different levels and every level has multiple clusters. All the aggregated packets coming from different cluster heads of higher level will be further aggregated by a leader node of lower level which is selected among the cluster heads of that level. Leader node will be selected depending on the distance from the base station and its residual energy. Finally the leader node will send the total aggregated packet to the leader node of next lower level and ultimately to the base station.

The remaining part of this paper is organized as follows: Section 2 deals with the review of state of the art routing topologies, section 3 gives a description of the proposed methodology, section 4 contains the simulation reports and section 5 is the concluding part.

2 Review Works

Energy Efficiency and increasing network longevity is the main research area in wireless sensor network for the last few years. To reduce energy drainage many algorithms have been designed to form clusters. A cluster head is selected to aggregate collected data and send that to the base station. Energy Efficient Clustering Algorithm for data aggregation in WSN [3] is one of the examples of clustering algorithm. It includes two phases of clustering. One is the formation of cluster heads. In this phase every node broadcast their radius, residual energy and co-ordinates to the neighbor nodes. Then the nodes will calculate competition bids to select the cluster head. The other phase is data aggregation and tree construction, which includes calculation of weight values for cluster heads depending on the distance from the base station and remaining residual energy. These weight values help to select the leader node among the cluster heads. The aggregated data will be sent to base station only by leader node which leads to uniform energy dissipation and long network longevity.

Another algorithm Energy Efficient Heterogeneous Clustered scheme for Wireless Sensor Network [5] has assumed that a percentage of sensor nodes are equipped with more energy and are immobile with known geographical locations. The introduction of computational heterogeneity includes more powerful microprocessor, more energy, complex data processing ability, which adds a lot of advantages to this model. The Link heterogeneity is introduced with the inclusion of high bandwidth and long distance network transceiver to prolong the lifetime of the network together with reliable data transmission. The Energy heterogeneity brought about the energy efficiency to the network, however increasing the implementation cost.

Low Energy Adaptive Clustering Hierarchy (LEACH) [1,2,11,20] is also cluster-based protocol, which includes distributed cluster formation. LEACH randomly selects a few sensor nodes as cluster heads (CHs). Then it rotates this role to evenly distribute the energy consumption among the sensors in the network. The cluster head (CH) compress the collected data from different nodes that belong to the respective cluster. At last that the cluster head sends that aggregated packet to the base station in order to reduce the amount of information that must be transmitted to the base station. This protocol is most appropriate when there is a need for constant monitoring by the sensor network. A user may not need all the data immediately. Hence, periodic data transmissions are unnecessary which may drain the limited energy of the sensor nodes. After a given interval of time, a randomized rotation of the role of the CH is conducted so that uniform energy dissipation in the sensor network is obtained.

The Centralized approach on the other hand includes protocols such as LEACH-C [1,2] which proposes the sending of location awareness of the nodes to the base station which in turn would select the cluster head on the basis of remaining energy. Such process is though effective on the basis of total energy dissipation, formation of correct number of cluster heads, the main disadvantage of this process lies in the overhead caused due to continuous involvement of the base station.

Hybrid Energy-Efficient Distributed (HEED) routing [8] is also a clustering approach, which is one of the most recognized energy-efficient clustering protocols. It extends the basic scheme of LEACH by using residual energy and node degree or density. In HEED, the initial probability for each node to become a tentative cluster head depends on its residual energy, and final heads are selected according to the intra-cluster communication cost. The clustering process is divided into a number of iterations, and terminates with in a constant number of iterations. HEED achieves fairly uniform distribution of cluster heads across the network.

PEGASIS [10] includes arranging of nodes into chains so that they can communicate only with their closest neighbor, thereby minimizing the power requirement for data transmission per second and reducing overhead. However failure of any intermediate node can cut off the link between other nodes and thus can detach a portion of the network.

Clustering and multi-hop routing with power control in Wireless Sensor Network provides clustering together with power control by means of algorithms like QFR and DCNPE [6], which are power efficient and thus prolongs the network lifetime. These algorithms state that the process of clustering is completed in two different phases, namely, THE SET UP PHASE, where the node with the maximum residual energy and maximum intra cluster broadcast power, is chosen as the cluster head; and THE STEADY STATE PHASE, where packets of request are send by the Base station through a proper power level, confirming that those packets never appear twice for a node. On receiving the request the sensor nodes send sensed data to the cluster head in a single hop; which in turn follows a reverse way to the path through which request came to it from the base station, to send the aggregated data to the Base station. The intra cluster communication process is same as LEACH whereas inter cluster communication takes place only between two cluster heads. The base station follows a power efficient path to send requests to the cluster head. Cluster heads follow the reverse path, by which energy of the nodes involved is reduced considerably.

QOS supporting and optimal energy allocation for a cluster based Wireless Sensor Network [7] states that together with energy efficiency, The Quality of Service, which includes source to link delay, data pass rate, data loss rate etc. must also be taken under consideration. The algorithm states that each cluster is controlled by a cluster head having a finite capacity called SINGLE FIXED RATE. The relaying of traffic from cluster to cluster till the sink to minimize the data congestion and increase network lifetime, makes the cluster heads depend on total relaying data rate from its own cluster as well as other clusters.

A novel hierarchical routing protocol algorithm for Wireless Sensor Networks [9] considers energy usage and packet latency together with security against node consideration attack. The simulation result proves that NHPRA can resist more against node consideration attack than LEACH due to the level maintained hierarchy adopted by them. The communication process among the sensors is same as GSPR, however the network is prone to outside node attack.

Energy Efficient Clustering under the joint Routing and coverage constraint [4] addresses optimal planning of the different states of sensors, providing energy efficient scheduling of the states, energy efficient routing, clustering and data aggregation. The algorithm formulates the problem as an ILP model and implementation of TABU search algorithm to manage exponentially increasing computation times. It mentions four different states of sensor node such as Transmit, Receive, Idle and Sleep. A subset of the total number of nodes will remain active at a time to save energy and reduce redundancy. The cluster heads are chosen dynamically on the basis of residual energy and distance from the neighbors and a spanning tree connects all cluster heads which are only capable of routing and thus send data to the sink. It is stated that all nodes have same sensing range and transmission range and the cluster heads are dynamically selected from the nodes.

3 Proposed Work

Most of the existing clustering algorithms send the aggregated packet to the base station from cluster head directly. As a result the energy of cluster head node goes down very quickly and it becomes non-functioning very soon. Some other algorithm like EECA [3] constructs a data aggregation tree where every sub cluster head sends the data to its parent node. The parent node contents of more weight value than its child. The weight value will be calculated depending on the remaining energy and distance from the base station. The problem with this algorithm is the child cluster head may send the aggregated data to its immediate parent node which belongs to the same level, instead of sending it to the next level directly. This leads to wastage of energy, delay and also reduces network lifetime. The proposed algorithm PACBR will divide the network into different levels, many clusters with cluster heads can be there in a same level, and sends the aggregated data to the base station in an energy efficient way and increases network lifetime also.

3.1 Basic Methodology

The Power Aware Cluster Based Routing (PACBR) protocol for Wireless Sensor Network is proposed to send the information of devastating events from source node to base station. The algorithm clusterizes the sensor nodes and chooses a cluster head for every cluster. Every node will send the sensed data to its cluster head and it will take the responsibility to aggregate those data and send that to the base station via other cluster heads. The algorithm assumes that the network is a static network, that is, after deployment the nodes are motionless and contents of same initial energy. Base station or sink node will be of high configuration, i.e. contents of high energy, enough memory etc., deployed in a controllable place outside the network region. All sensor nodes are capable of data aggregation, compute its own residual energy and can find its geographic location.

All the sensor nodes send their geographic location to the base station which will group the sensor nodes into different clusters accordingly. The whole network will be divided in different levels and a level can have more than one clusters. The level nearest to base station will be the lowest level (i.e. level 1).

After deployment all the nodes will start to sense different events and send the sensed data to its cluster head. The cluster heads aggregate the data and send them to the sink node. A huge amount of energy is needed for this type of transmission which may cause early death of the cluster head. Thus for every round, change of cluster head is required. The algorithm helps to select the new cluster head and route the aggregated data to the base station.

The energy consumed in transmitting information from one node to another can be measured with the help of Inverse Square Law. It states that the intensity (energy per unit area perpendicular to the source) of linear waves radiating from a point source is inversely proportional to the square of the distance from the source. The intensity of energy for sending data from one node to another is thus –

$$\mathrm{E_I} = \frac{C}{(distance)^2} \tag{1}$$

Where C is a constant.

The consumption of energy will be increasing with the decrease in intensity. Thus energy consumed for a node i after sending n number of messages can be calculated as—

$$\mathrm{E_{ci}} = \sum_{j=1}^{n} \mathrm{k} * \mathrm{d_{ij}^2} * \mathrm{N_m} + \mathrm{E_a} \tag{2}$$

Where,

d = Distance between nodes i and j
Nm = Total number of sent messages from node i to j
k = Constant value
E_a = Energy required for data aggregation

The residual energy for node i will be—

$$E_{Ri} = E_i - E_{ci} \tag{3}$$

If the residual energy E_{Ri} of node i is greater than the threshold value then it becomes a candidate node of cluster head selection process. Threshold value can be defined as energy required to accept data from all nodes, aggregate that, and send that to neighbor nodes. Every candidate node will calculate the selection probability value Pi for itself.

$$P_i = \frac{E_{Ri}}{D_i} \tag{4}$$

Where,

$$D_i = \sum_{j=1}^{n} \mathrm{d_{ij}}/\mathrm{n} \tag{5}$$

E_{Ri} is the remaining energy at node i and D_i is the average distance of that node from all its neighbor nodes lying in the same cluster. ‘n’ is the total number of neighbor nodes of node i in that cluster.

Now all the nodes will broadcast their probability value P_i to their neighbors. Every node will check the values and find the maximum. The node containing the maximum P_i value will send the success message to all its neighbors.

A leader node is selected from among the cluster heads of the next lower level. The cluster head with highest weight value will selected as the leader node of that level. Every cluster head will try to send the aggregated data collected from different nodes to any of the cluster head of the next lower level. To select the cluster head of the next lower level it will calculate a weight value of those cluster heads by the following equation—

$$W_t = K * \frac{E_R}{D_B * D_N} \tag{6}$$

Where,

E_R=Remaining energy of the cluster head of next lower level to which the current cluster head of higher level can send data.
D_B=Distance from the base station of the cluster head of next lower level.
D_N=Distance of the current cluster head from the cluster head of the next level to which it can send data.
K=constant

The cluster head with highest weight value shall be called leader node of that level. Thus the cluster head will send the aggregated data to the leader node of next lower level. Then that leader node will send that to its next lower level leader node and so on. The process continues until the aggregated data reached at lowest level (i.e. level 1). The cluster head of lowest level will not try to calculate any weight value and send the aggregated data to the base station directly.

3.2 Data Dictionary

Table 1. Variables list

Variable name	Description
N	Total Number of nodes
x,y	Coordinate values of every node
d[i][j]	Distance between nodes i and j
P[node_number]	An array consists probability values of each node
E[node_number]	An array consists remaining energy of each node
E_{Ci}	Stores consumed energy for node i
D_i	Average distance of all neighbor nodes from node i

3.3 Description of PACBR Algorithm

```
1. Input geographic location (Two coordinate values
   x,y) of every node.
2. Make the cluster depending on the value of x and y,
   and send the Cluster Information Packet (CIP) [Node
   ID, Cluster ID] to every node $N_i$.
3. //Calculate energy consumed for every node i—
     Repeat step 4 and 6 for i = 1 to n
4. Repeat step 5 for j=1 to k
5.        a) Read Nm and Ea
            b) Eci=Eci+(c*d[i][j]*Nm)+Ea
     //End of Inner loop
6. //Calculate Residual Energy
            Set E[i]=E[i]-Eci
     //End of outer loop
7. For i=1 to n repeat step 8
8.        If E[i] > Eth Then
          a. Set S[i]=1
          b. For j=1 to n repeat step c
          c.        Set D=D+d[i][j]
                        //End of loop
          d. Set Di=D/n
          e. Set P[i]=E[i]/Di
             Else
                   Set S[i]=0, P[i]=0
     //End of Forlooop
9. //Broadcast probability value of every node to its
   neighbor within cluster only
      Call Prob_Broadcast(P[ ],n)
10.      //Find the max value among the received prob-
   ability values
     Call Find_max()
11.      For i=1 to n repeat step 12
```

```
12.         If P[i]>max then
        Send the success message to all other nodes
   within the cluster
       Else
        Wait for success message
          //End of loop
13.     Send data to CH
14.     Aggregate collected data and send to base
   station
```

Prob_Broadcast(P[ ],n).

```
         For every node of cluster i do—
             If cluster id matches with the cluster id
of neighbor node j then
                   Send the probability value to node j
```

Find_max( ) .

```
      1.Set max = 0

2.Forevery node within the cluster i
                   If probability value p>max then do
                        Set max = p
             // end loop
```

3.4 Case Study

In the above figure, the base station divides the network into different levels and there are different numbers of clusters in each level. The black nodes are the cluster heads of the respective clusters. Each cluster head node calculates its weight value W_t except the cluster head nodes of the highest level or LEVEL 4. The weight values of the cluster head nodes of level 3 are 2.4, 2.47 and 2.3 and their distance from the base station are 255unit, 240unit and 245unit respectively and thus the node with the highest weight value that is the cluster head node having the weight value 2.47 is chosen as the leader node of level 3 and is marked in the figure with a rectangle. Every cluster head node of level 4 forwards their data to the leader node of level 3. In level 2 the distance from the base station of the cluster head nodes are 100unit, 130unit and 98unit and their weight values are obtained as 3, 3.5 and 3.2 respectively. It is observed that the node having distance from the base station 98unit has a lower weight value than the node having distance from the base station 100unit, due to having lower residual energy. Thus the node having greater weight value is chosen as the leader node to increase the lifetime of the network. Forwarding of data to the leader nodes of the next lower level continues till level 1 is reached. All the cluster head nodes of level 1 directly forward the data to the base station.

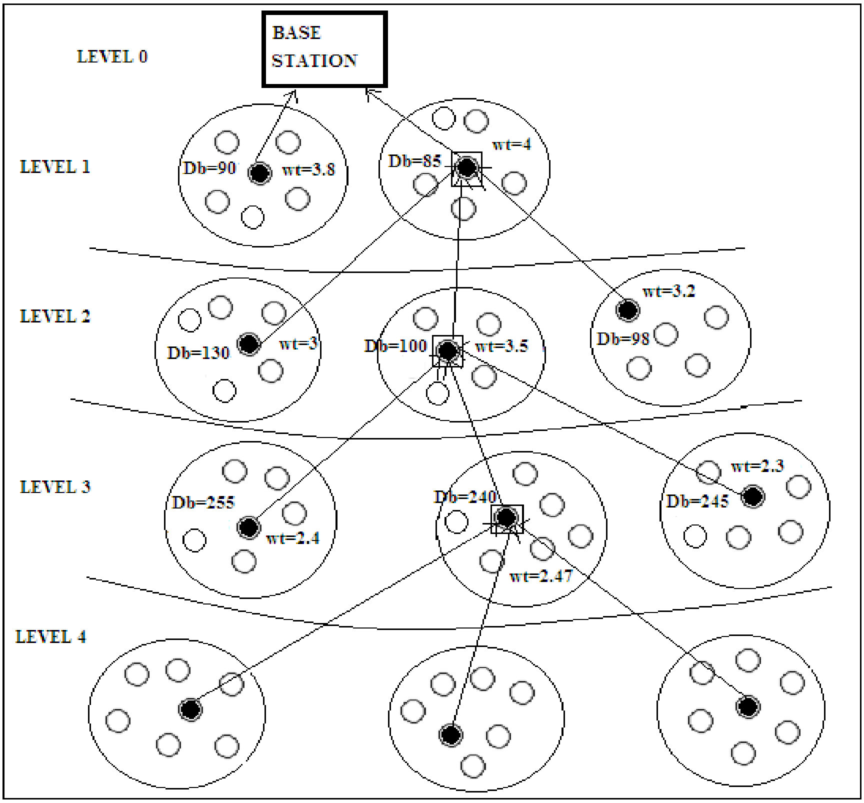


Fig. 1. Sending aggregated data through different levels

4 Simulation Result

To analyze the performance of the algorithm, a network with 20 nodes is created, which are distributed equally in four clusters residing in two different levels. The parameter list is given below—

Table 2. Parameter list

Parameters	Description
Network size	100 nodes
Initial energy	5000J per node
MAC Protocol	IEEE 802.15.4
Power consumption	Equivalent to packet size and distance
Number of rounds	At least 6

The initial power of every node is considered 500 units. The size of each packet of data is taken as 1KB. After every round the total reduction in energy of the network and the total number of dead nodes is calculated. The result obtained is compared with that of Energy Efficient Clustering Algorithm (EECA) [3] which proves that after 6th round more nodes are dead for EECA with compare to PACBR.

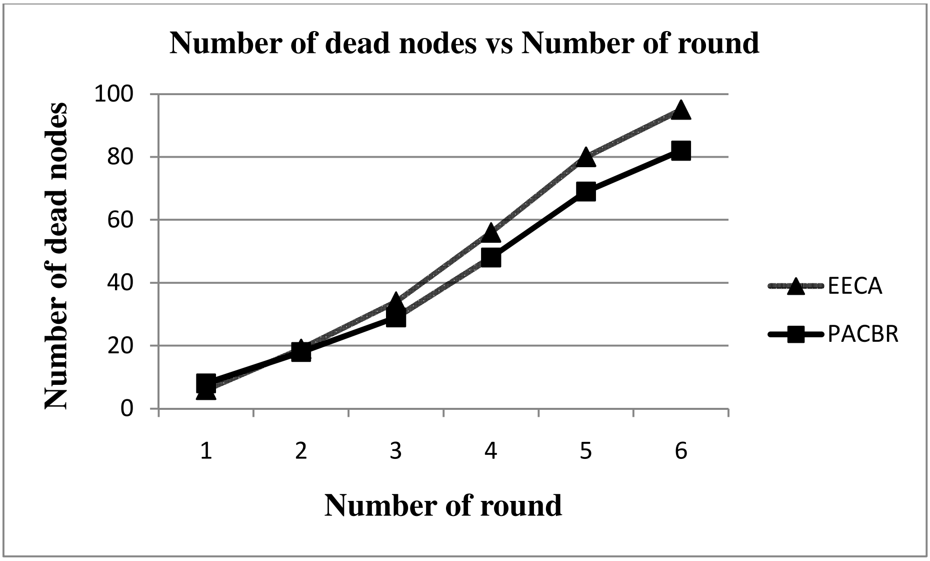


Fig. 2. Number of dead nodes in the network after each round

Again after each round the average residual energy of the network is measured for both the algorithms and has plotted as follows—

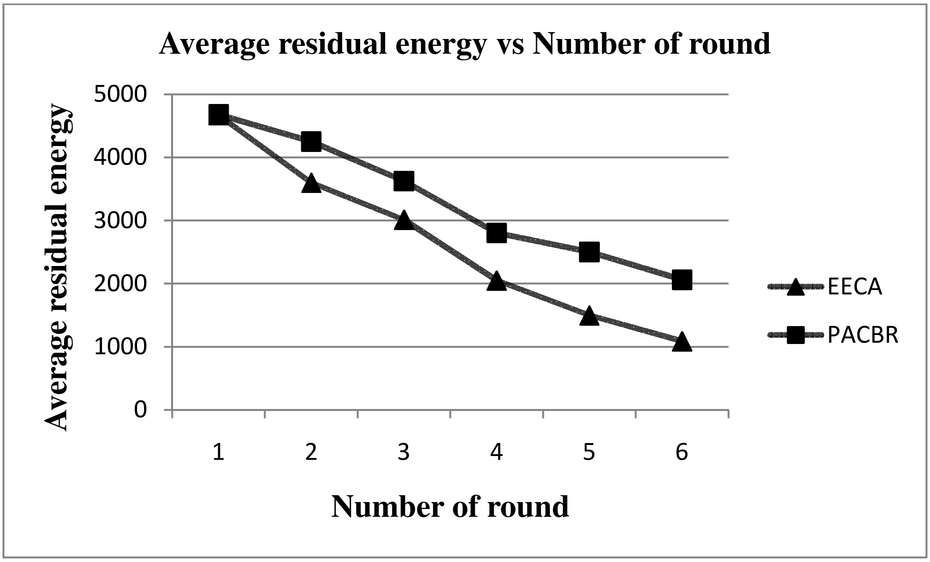


Fig. 3. The total loss of energy after each round

The above figure shows that after 6th round the algorithm EECA [3] lost more energy than PACBR. Thus the proposed algorithm PACBR can send the aggregated data to the base station in an energy efficient way and increases network lifetime.

5 Conclusion

The proposed algorithm Power Aware Cluster Based Routing (PACBR) protocol for Wireless Sensor Network takes care of the longevity of the network. It divides the network into different levels and each level contents of multiple clusters. Each cluster has a cluster head. The cluster heads collects sensed data from other nodes, aggregate that and sends that to the base station in an energy efficient way. The simulation results prove that the algorithm works more efficiently than other existing algorithms and also increases network lifetime.

References

1. Geetha, V., Kallapur, P.V., Tellajeera, S.: Clustering in Wireless Sensor Networks: PerformanceComparison of LEACH& LEACH-C Protocols Using NS2. In: 2nd International Conference on Computer, Communication, Control and Information Technology (C3IT 2012), February 25-26, vol. 4, pp. 163–170. Elsevier Journal (2012)
2. Xinhua, W., Sheng, W.: Performance Comparison of LEACH & LEACH-C Protocols by NS2. In: 9th International Symposium on Distributed Computing and Applications to Business Engineering and Science (DCABES) (2010)
3. Sha, C., Wang, R., Huang, H., Sun, L.: Energy EfficientClustering Algorithm for data aggregation in WSN. Elsevier Journal, the Journal of China Universities of Posts and Telecommunications (December 2010)
4. Chamam, A., Pierre, S.: On planning of WSNs: Energy Efficient Clustering under the joint Routing and coverage constraint. IEEE Transactions on Mobile Computing 8(8) (August 2009)
5. Kumar, D., Aseri, T.C., Patel, R.B.: Energy Efficient Heterogeneous Clustered scheme for Wireless Sensor Network. Computer Communication 32(4), 662–667 (2008)
6. Guo, S., Zheng, J., Qu, Y., Zhao, B.H., Pan, Q.K.: Clustering and multi-hop routing with power control in wireless sensor networks. The Journal of China Universities of Posts and Telecomunications 14(1) (March 2007)
7. Tang, S., Li, W.: QoS supporting and optimal energy allocation for a cluster based wireless sensor network. Computer Communication Journal 29(13-14), 2569–2577 (2006)
8. Younis, O., Fahmy, S.: HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks. IEEE Transactions on Appears in Mobile Computing 3(4) (October-December 2004) ISSN: 1536-1233
9. Cheng, H., Yang, G., Hu, S.: NHRPA: a novel hierarchical routing protocol algorithm for wireless sensor networks. The Journal of China Universities of Posts and Telecommunications 15(3), 75–81 (2008)

10. Lindsey, S., Raghavendra, C.S.: PEGASIS: power efficient gathering in sensor information systems. In: Proceedings of the IEEE Aerospace Conference, Big Sky, Montana (March 2002)
11. Heinzelman, W.B., et al.: An application-specific protocol architecture for wireless microsensor networks. IEEE Transactions on Wireless Communications 1(4), 660–670 (2002)
12. Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H.: Energy Efficient Communication Protocol for Wireless Microsensor Networks. In: Proceedings of the 33rd Hawaii International Conference on System Sciences (2000)

A Novel Incentive Based Scheme to Contain Selective Forwarding in Wireless Sensor Network

Saswati Mukherjee[1], Matangini Chattopadhyay[1], Samiran Chattopadhyay[2], Debarshi Kumar Sanyal[3], Roshni Neogy[2], and Samanwita Pal[2]

[1] School of Education Technology, Jadavpur University, India
[2] Department of Information Technology, Jadavpur University, India
[3] Xilinx India Technology Services Pvt. Ltd., Hyderabad
sash_cal@rediffmail.com, {matanginic,samirancju}@gmail.com, debarsh@xilinx.com

Abstract. Selective forwarding or dropping of packets is a serious threat to multi hop communication in a Wireless Sensor Network (WSN). There are various schemes to induce cooperation in a WSN to overcome this problem. In this paper, we have introduced a novel adversary model and have proposed an incentive based scheme to inspire cooperation among nodes in a WSN. The scheme has been formally analyzed. The efficacy of the scheme is also established through various simulation experiments.

Keywords: Wireless Sensor Network, Selective Forwarding, Random Graphs, Virtual Currency, Throughput, Network Security.

1 Introduction

Nodes in mobile ad-hoc networks are arbitrarily deployed without relying on any fixed network infrastructure. In a multi-hop wireless network, many pairs of nodes cannot communicate directly and must forward data to each other via one or more intermediate forwarding nodes.

Multi-hop communication is not an issue where nodes faithfully forward packets according to a global algorithm. Selfish nodes may like to send their own packets but may not be ready to relay packets for others since relaying packets for others consumes bandwidth and energy. This, in turn, decreases both individual and system throughput and might even lead to loss of connectivity in a network.

Hence, cooperation among the nodes needs to be enforced. The basic aim of any such mechanism is to force nodes to forward packets sent to it by other nodes. There are many proposed solutions [1-2] which use game theoretic and graph theoretic notions to examine whether cooperation can exist in multi-hop communication while many solutions are proposed based on providing incentives. Incentives can be positive or negative. That is, a node can be made to cooperate within a network either by providing some incentive or by taking punitive actions against a node when its rate of packet forwarding falls below a particular value. Marti et al. [4] have discussed schemes to identify misbehaving nodes (non-forwarders) and deflect traffic around

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 301–312, 2013.

them. Michiardi and Molva [5] have devised reputation mechanisms where nodes observe the behavior of others and prepare reputation reports. Zhong et al. [6] proposed the use of currencies to enforce cooperation. Buttyan and Hubaux [7, 8] devised a scheme based on a virtual currency called a *nuglet* that a node pays to send its own packets but receives, if it forwards other's packets.

In all these papers, nodes are classified in two categories: trusted those who forward packets and malicious, those who do not like to forward others' packets. Moreover, malicious nodes, according to these papers are content by dropping packets to conserve their resources. In this paper, we have introduced two further dimensions to this misbehavior model. First, we introduced a 'rational adversary' category of nodes. 'Rational adversary nodes' do not mind dropping packets if they are not penalized for that. Second, we have incorporated an idea by which 'malicious' nodes inspire their neighboring nodes to drop packets.

In this paper, we have also proposed design of a point based credit scheme in a fairly dense sensor network that encourages nodes to cooperate in packet forwarding and thereby contains selective forwarding and restores throughput of the network. In the proposed scheme, nodes that forward packets get incentives in the form of credit points. Nodes that drop packets to conserve resources are penalized by deducting credit points from them. Malicious nodes may send 'bribe' packets to neighbouring nodes to inspire them to drop packets. If a 'bribe' packet reaches a trusted node, the sender node is stripped of a good number of credit points. Credit point reserve of any node may not go below a threshold limit. The network is considered as an undirected graph with flat/unstructured topology. In flat topology each sensor node performs similar functions like packet forwarding data sensing with some exceptions in their behaviour as defined by the adversary model.

We have shown that if the graph is sufficiently dense then the likelihood of detecting malicious nodes is very high. We have also proposed an algorithm which helps to use the proposed scheme when the network is less dense. If the proposed scheme is allowed to run then simulation results suggest that the throughput of the network remains considerably high as long as the numbers of rational adversary nodes are less in proportion. Message overhead is low with small number of trusted and rational adversary node despite having large number of malicious nodes. The proposed scheme achieves scalability even if the number of nodes gets increased. Although the basic approach assumes a dense network, we have also presented a mechanism by which our scheme continues to work in less dense networks.

The remainder of this paper is organized as follows. Section 2 presents the related research work. In Section 3, the model definitions of the proposed work are described. Section 4 presents algorithmic view of the proposed scheme. In this section, we have also analyzed the scheme analytically. In Section 5, experimental results are presented to measure the performance of our proposed method. Finally, we draw our conclusion.

2 Related Work

The work to enforce cooperation in mobile and wireless ad hoc network is based on providing incentives while other works are reputation and trust based. Some others are formalized in game theoretic framework. In this section, we review some important reputation based and incentive based schemes.

CORE [5], CONFIDANT [5], OCEAN [9] are examples of reputation and trust based systems. In CORE and CONFIDANT systems, non-cooperative nodes are detected by using certain reputation measures. In CONFIDANT, the reputation of non-cooperative nodes is propagated throughout the network for punishment. In CORE, nodes with bad reputation are gradually removed from the network. CORE consists of two basic components: a watchdog mechanism and a reputation table. The watchdog mechanism is used to detect misbehavior nodes but the disadvantage is that it not only creates a performance bottleneck by increasing network congestion, transmission overhead etc. but also diminishes the network scalability. OCEAN, another trust based method, has five components to detect non-cooperative nodes and mitigate the risk of selective forwarding. The overhead to calculate trust or reputation values is a common severe problem of these methods.

Buttyan and Hubaux [7-8] proposed the concept of providing incentives to the nodes in a static wireless ad hoc network so that they faithfully forward packets. According to this scheme, nodes get paid for sending packets to other nodes. The 'money' used in this scheme is termed as *Nuglets*. The scheme is implemented using two models: The *Packet Purse Model* and *Packet Trade Model*. In *Packet Purse Model*, the originator of the packet pays for the packet forwarding service. The originator loads it with the number of *nuglets* sufficient to reach the destination. Each forwarding node acquires one or several *nuglets* from the packet and thus, increases its stock of *nuglets*. If the packet does not have enough *nuglets* to be forwarded, the packet is discarded. The problem with this model is that estimating the number of *nuglets* to be loaded with the packet is difficult. In *Packet Trade Model*, the packet is traded for *nuglets* by intermediate nodes. Each intermediary buys it from previous one for some *nuglets* and sells it to the next one for more *nuglets*. A basic disadvantage of this model is that packet flooding is possible. This scheme requires tamper proof hardware for reliably calculating the *nuglet* distribution.

Sheng Zhong et al. [6] proposed a scheme called *Sprite*, a simple cheat-proof credit based system. In *Sprite,* a Credit Clearance Service (CCS) is introduced to determine the charge and credit to each node involved in message transmission. When a node receives a message, the node keeps a receipt of the message and later reports it to the CCS. This scheme requires a central server to determine the charge and credit to each node involved in message transmission.

Salem et al. [10] proposed another incentive mechanism based on charging and rewarding scheme which forces selfish nodes to rationally opt for forwarding packets. Their proposal provides a set of protocols that rely on symmetric cryptography techniques.

In [11], Jackobson and others have proposed a micro-payment scheme for multi-hop cellular networks that encourages collaboration in packet forwarding. In this paper, an asymmetric communication model is assumed.

The scheme presented in this paper uses the basic concept of incentives as discussed in [6-8]. But our paper differs from them in many ways. First, our paper introduces a new adversary model in a fairly dense static wireless sensor network which is richer than the one proposed in all these papers. Second, our scheme is distributed with minimal overhead. Third, simulation results are provided to show that the scheme is scalable and effective.

3 Model Definitions

In this section, we give a clear definition of the proposed models and the reasons why we have chosen this model. Since we are studying cooperation in packet forwarding, we assume that the main reason for packet losses in the network is the non-cooperative behavior of the nodes.

3.1 Adversarial Model

We have identified primarily two types of non-cooperative nodes: faulty or malicious or misbehaved and selfish or rational adversary. Faulty/malicious activity refers to that class of misbehavior where nodes try to attack the system. They threaten the entire network by dropping packets in order to conserve their resources and by inspiring neighbors not to forward packets. As our scheme is based on credit point system where points are to be acquired by every node, the sanctity of the credit points in each node is essential. The misbehaved nodes are assumed to be capable of manipulating the credits available in the non-trusted neighbors around them. This is modeled by the malicious nodes being able to send "bribe" packets to their neighbors. A bribe packet offers certain points to its receivers. By receiving a bribe packet, a selfish node may drop some packets without jeopardizing its stock of points.

The adversary is rational, in the sense that it will only attempt to cheat if the expected benefit of doing so is greater than the expected benefit of acting honestly in network related operations. Naturally, if a rational adversary node is offered a "bribe" packet to increase its credit point, it will actually accept it as long as there is no harm in accepting it.

The third category of nodes is termed trustworthy or trusted. They refer to the class of well-behaved nodes that functions reliably and honestly throughout the network operations.

3.2 Network Model

We have considered a wireless network containing N nodes and the nodes are divided in three categories as explained in the Adversary model. The topology of the wireless sensor network is basically an undirected graph where an edge between two nodes denotes that they can communicate with each other. The topology is flat as these nodes are homogeneous except in their capability defined by the adversary model.

The network is modeled as an "Uncorrelated random graph (Erdős–Rényi)" [12] where N nodes are connected through n edges which are chosen randomly from the $N(N\text{-}1)/2$ possible configurations. The probability of selecting an edge is p.

3.3 Point Based Credit Scheme

A Point Based credit mechanism of charging/rewarding the service (which is forwarding a packet in this case) is presented to stimulate node cooperation in wireless ad-hoc networks. In this section, we give a formal description of proposed Point Based credit scheme.

- All the nodes in the network are initialized with some credit points.
- Nodes earn credit points as rewards if they forward packets.
- Nodes lose credit point as penalty if they do not forward packets.
- Credit points assigned to all the nodes must not fall below a specified threshold value.
- Trustworthy nodes always forward packets and earn points and hence they always help in increasing the network throughput.
- Bribe packets are offered by the malicious/misbehaving nodes to random neighboring nodes inspiring them to drop packets in order to pull down the network efficiency. Every bribe packet contains a certain number of points. These points are deducted from the stock of points of the malicious nodes.
- The rational adversary nodes on receiving the bribe packets check if its accumulated points are above the threshold value. If so, then they do not forward packets so long as points accumulated by accepting bribe packets are more than points lost due to not forwarding packets. Otherwise, they forward packets and earn credit points as rewards.
- A malicious/misbehaving node forwards packet if it's accumulated credit point falls below the threshold.
- Trustworthy nodes on receiving bribe packets from malicious nodes penalize them by deducting points.

The execution of the proposed charging/rewarding scheme requires tamper proof hardware [7-8] to monitor the addition or deduction of points assigned to the nodes or requires a central server to assess the charge involved or credit points of each node for message transmission. Thus, we do not assume any MAC layer misbehavior.

4 Algorithmic Description of the Proposed Scheme

The proposed scheme has two phases: Network Deployment and Charging/Rewarding Service.

```
Network Deployment Phase
Step 1:  Input the number of nodes for each class of nodes.
Step 2:  Assign equal credit points $E$ to all the nodes.
Step 3:  Set threshold limit $T$ of credit point to all the nodes.
```

```
Algorithm PACKET_FORWARDING_SCHEME
For (;;) {
    Step A.     The MALICIOUS nodes send out a number of "bribe" packets depending on
                the amount of its stock of points to randomly selected neighbouring nodes.
    On receiving a packet
    Step B.1:   If the node is malicious,
    Step B.1a:      Check its credit points.
    Step B.1b:      If credit point of malicious node is greater than threshold limit,
    Step B.1c:          Then drop the packet.
    Step B.1d:          Else forward the packet and earn $C$ points.
    Step B.2:   Else If the node is trusted
    Step B.2a:      If the type of the packet is "bribe packet"
    Step B.2b:          Then deduct $P$ points from the sender malicious node by send-
                        ing punishment packet
    Step B.2c:      Else forward the packet and earn $C$ points.
    Step B.3:   Else If the node is rational adversary
    Step B.3a:      If the type of the packet is "bribe packet" with $B$ point
    Step B.3b:          Then Accumulated_Bribe += $B$
    Step B.3c       If $C$ <= Accumulated_Bribe_Points
    Step B.3c:          Then drop the packet; Accumulated_Bribe -= $C$
    Step B.3d:          Else, forward the packet and earn $C$ points.
}
```

In the *for* loop, Step A and Step B run in parallel.

4.1 Analysis of Algorithm

For a random graph with n number of nodes, suppose that the probability of choosing an edge is *p*. That is, *p*, fraction of the total number of edges is selected randomly in such a graph. It is known that in such a graph, as n tends to infinity, the probability that a graph of *n* vertices with edge probability $p = 2ln\ (n)/n$ is connected, tends to 1.

It is also known that the average degree of a node in such random graph is *np* with variance npq where $q = 1 - p$.

In our case, $n = (n1+n2\ +n3)$ where *n1*, *n2*, *n3* denote the number of malicious nodes, rational adversary nodes and trustworthy nodes respectively. Thus, the average number of edges from any node to a trusted node is n_3p. However, this is the mean value. The connectivity of any node to a trusted node may decrease from the mean value by the square root of the variance. Thus, the following inequality must hold to maintain connectivity to a trusted node from any given node.

$$n_3 p > \sqrt{npq} \tag{1}$$

Solving the inequality, we get,

$$p/(1-p) > n/n_3^2 \tag{2}$$

Substituting n/n_3^2 with *k,* we get,

$$p > k/(1+k) \tag{3}$$

Therefore, if the fraction of edges in the network is greater than *k/(1+k)* then it is likely that a malicious node is always connected to some trusted node. In such a scenario, the bribe packet randomly sent out by a malicious node would reach the trusted node some time and the malicious node will surely be penalized.

4.2 Finding Positions of Additional Trusted Nodes

The proposed scheme heavily depends on a misbehaved node being connected with at least one trusted node in its neighborhood. So, the network becomes dense. In a given deployment, it may so happen that each malicious node does not have a trusted node as its neighbor. In such situations, we can adopt the following scheme to deploy additional trusted nodes if necessary so that every malicious node is ensured to be connected to a trusted node. Our approach is based on Minimum Connected Dominating Set of the network.

We have made some additional assumptions to handle the case where each malicious node is not connected directly to a trusted node. These assumptions are as follows. The trusted nodes are assumed to be deployed with a key for secured communication. All nodes are assumed to be equipped with location information. Location information of a node denotes information about its spatial coordinates in a given area.

The following algorithm determines the locations where additional trusted nodes should be deployed so that direct edge connectivity between a malicious node and a trusted node is ensured.

If the graph contains *n* nodes then Step 1, Step 2 can be performed in O(n) [13] time. If the approximate MCDS of G contains h number of nodes then Step 3a, Step 4a can be performed in O(h) time [14]. Step 3b.and Step 4b takes O(h^2) time. Step 5 can be performed in O(h) time.

```
Algorithm FindPositionsOfAdditionalTrustedNodes
Input: The Graph G describing the topology of the wireless sensor network
Output: Locations where additional trusted nodes should be deployed
Step 1:    Let T = set of trusted nodes. Let u be a designated trusted node.
           // These nodes have a key for secured communication.
Step 2:    Find an approximate Minimum Connected Dominating Set (MCDS) C of
           G starting with T using a standard greedy algorithm in the literature.
Step 3a:   u broadcasts a message asking for the location information from each
           node in C.
Step 3b:   u receives the location information from all nodes.
Step 4:    u broadcasts a challenge message to all nodes in C
           u receives responses only from the trusted nodes.
Step 5:    u finds out the locations of the nodes in C which are not trusted and writes
           them in P
Step 6:    return P
```

5 Experimental Analysis

A simulation experiment is performed with a by considering a number of random graphs having 100 wireless nodes and 1000 edges. In a given topology, there would be many different possibilities of mixes of trustworthy nodes, misbehaved nodes and rational adversary nodes. In order to model these varieties, we have chosen ten sets of different ratios of these three classes of wireless nodes as shown in Figure 1.

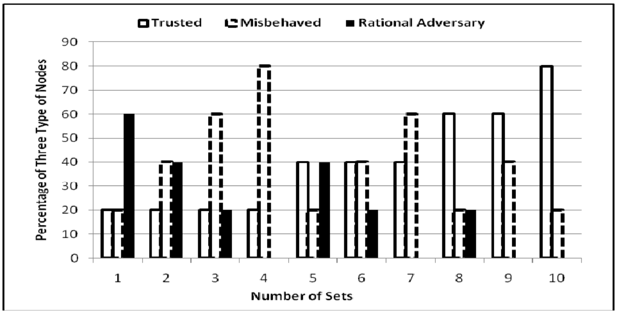


Fig. 1. Different sets of nodes with various proportions of trusted, misbehaved and rational adversary nodes in the Network

The rational behind the choice of different sets is as follows. Suppose we fix a small number of trusted nodes, say 20, in the network. Keeping this number fixed, we can choose several numbers of malicious nodes. Suppose that, the number of malicious nodes is chosen to be 20, 40, 60 and 80. Once the number of trusted nodes and malicious nodes are fixed, the number of rational adversary nodes can be computed. Next, we increase the number of trusted nodes to 40 and continue the same process. For every set, a number of random graphs are generated in our experiment.

The PACKET_FORWARDING_SCHEME algorithm is executed in terms of rounds where each round signifies time duration. There are several parameters in the algorithm. In our experiments, parameters of the algorithm are assigned with the following values.

- Each node is initialized with 15 credit points.
- Each node earns 1 credit point as reward if it forwards a packet.
- Each node is penalized by 1.5 credit point if it drops a packet.
- *Misbehaved* nodes are penalized with 2 points by the trusted node, if the trusted node receives a bribe packet from the misbehaved node.
- Offer in the “BRIBE” is set at 0.1 percent of the points owned by the malicious nodes sending the bribe packet
- Threshold credit point of a node is 7.5.

In the experiments *throughput* is defined as the number of packets dropped to number of packets generated. Detection ratio is defined as the ratio of the number of

MALICIOUS nodes detected to the actual number of MALICIOUS nodes in the network.

– Throughput measurement in the face of selective forwarding

In Figure 2, throughput is plotted against different proportion of *trusted*, *misbehaved* and *rational adversary* nodes. Recall that ten different compositions of trusted, rational adversary and misbehaved nodes are considered as shown in Figure 1.

We can see that the throughput drops for Set 1 and Set 5. We also note that this is because in both these sets, the number of *rational adversary* nodes is considerable. As the number of *rational adversary* nodes is significant, *misbehaved* nodes can offer many "bribe" packets to them. These bribe packets induces *rational adversary* nodes not to forward packets resulting in a drop in throughput.

In the presence of less number of rational adversary nodes, throughput is improved even if there are a small number of *trusted nodes* compared to the number of *malicious nodes*. This is observed for Set 4, Set 7, Set 9, and Set 10.

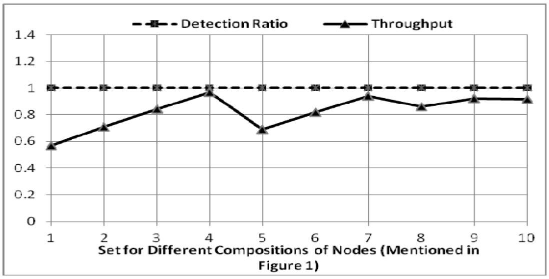


Fig. 2. Throughput of 10 sets of sensor nodes having different proportions of trusted, rational adversary, and malicious nodes

Figure 3 demonstrate that throughput remains almost unchanged once all malicious nodes are detected and they are routinely penalized. In Figure 3, we have chosen Set 5 and plotted throghput with respect to the number of rounds. It can be seen that after 10 rounds all malicious nodes are detected and throughput also remains unchanged at around 0.7 thereafter.

Similar trends have also been observed for Set 4 and Set 7.

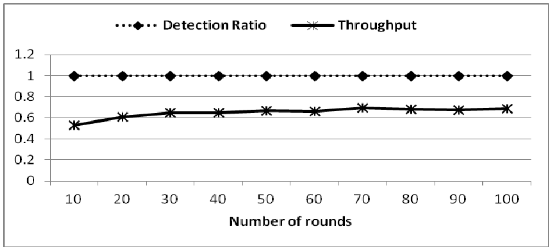


Fig. 3. Throughput after increasing number of rounds for Set 5

– Communication Overhead and 0.Scalability

Figure 4 depicts the communication overhead incurred in the algorithm. Overhead is measured by number of "bribe" packets and "penalty" packets exchanged. The number of these messages is averaged over 100 rounds.

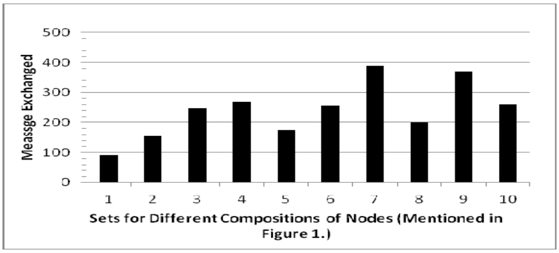


Fig. 4. Message Exchange overhead for all sets of combination of 3 types of node

Figure 5 shows the relationship between the number of rounds with the number of nodes to detect all malicious nodes. We have chosen three combinations of three types of nodes as defined by Set 1, Set 2 and Set 3. It can be seen that the number of rounds to detect all malicious nodes remains almost the same even if the number of nodes is increased.

Figure 6 plots average number of messages exchanged per node when number of nodes increase for various compositions of 3 types of nodes. The average numbers of message exchanges are not growing rapidly with respect to the number of nodes. Thus, it can be claimed that the scheme is scalable.

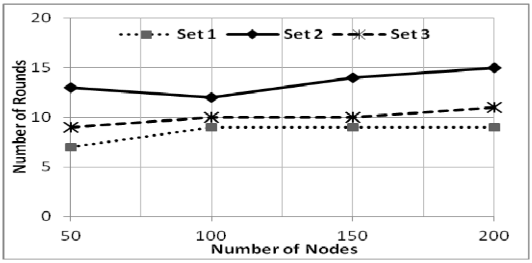


Fig. 5. Number of rounds required for 100% detection of misbehaved nodes with various number of nodes

– Effect of rational adversary nodes

In Figure 7, we have tried to capture the effect of the number of rational adversary nodes keeping the number of *trusted* nodes to 20. The number of misbehaved nodes is changed. Accordingly, the number of rational adversary nodes increase as the total numbser of nodes is kept at 100. It is observed that throughput steadily increases as the number of rational adversary nodes decrease (and the number of *misbehaved* nodes increase).

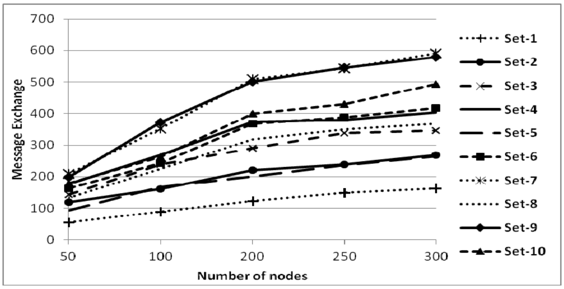


Fig. 6. Average number of messages exchanged for all sets having different compositions of nodes (as mentioned in Figure 1)

We can also verify the same from Figure 2. You may recall that the proportion of rational adversary nodes is more in the Sets 1, 2, 5, 3, 6, 8 in a non-increasing order. Throughput values for these sets in the graph reflect that the presence of rational adversary nodes bring throughput down.

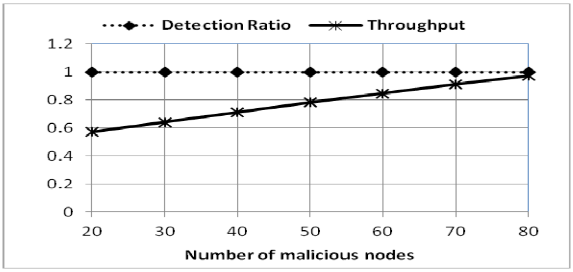


Fig. 7. Effect of rational adversaries with respect to throughput

6 Conclusion

In the paper, we have introduced a new adversary model for Wireless Sensor Networks. In this model, the idea of rational adversary nodes is floated in addition to trusted and malicious nodes. An incentive based scheme is presented to induce cooperation among nodes to solve the problem of selective forwarding. The scheme is analyzed and studied through simulation experiments.

The incorporation of rational adversary nodes introduces new twists to the problem. As a future work, we need to study further the effect of such nodes in throughput. We also plan to invent a trace algorithm which can detect malicious nodes even when it is not directly connected to some trusted node.

Acknowledgements. The work proposed is supported in part by UGC-UPE Phase II program of Jadavpur University.

References

1. Felegyhazi, M., Hubaux, J.P., Buttyan, L.: Nash equilibria of packet forwarding strategies in wireless ad hoc networks. IEEE Transactions on Mobile Computing 5(5), 463–476 (2006)
2. Mukherjee, S., Dey, S., Mukherjee, R., Chattopadhyay, M., Chattopadhyay, S., Sanyal, D.K.: Addressing Forwarder's Dilemma: A Game-Theoretic Approach to Induce Cooperation in a Multi-hop Wireless Network. In: Das, V.V., Stephen, J. (eds.) CNC 2012. LNICST, vol. 108, pp. 93–98. Springer, Heidelberg (2012)
3. Marti, S., Guili, T.J., Lai, K., Baker, M.: Mitigating Routing Misbehavior in Mobile Ad Hoc Networks. In: 6th Annual International Conference on Mobile Computing and Networking, pp. 255–265. ACM, New York (2000)
4. Michiardi, P., Molva, R.: CORE: A COllaborative REputation mechanism to enforce node cooperation in mobile ad hoc networks. In: IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security, pp. 107–121. ACM, New York (2002)
5. Zhong, S., Yang, Y.R., Chen, J.: Sprite: A simple, cheat-proof, credit-based system for mobile ad hoc networks. In: Twenty-Second Annual Joint Conference of the IEEE Computer and Communications, INFOCOM. IEEE Press, New York (2003)
6. Buttyan, L., Hubaux, J.P.: Enforcing service availability in mobile ad-hoc WANs. In: MobiHOC, pp. 87–96. IEEE Press, New York (2000)
7. Buttyaan, L., Hubaux, J.P.: Nuglets: a virtual currency to stimulate cooperation in self-organized mobile ad hoc networks. Technical Report, DSC/2001/001, Department of Communication Systems, Swiss Federal Institute of Technology (2001)
8. Bansal, S., Baker, M.: Observation-based Cooperation Enforcement in Ad Hoc Net-works (2003), http://arxiv.org/pdf/cs/0307012.pdf
9. Salem, N.B., Buttyan, L., Hubaux, J.P., Jakobsson, M.: A charging and rewarding scheme for packet forwarding in multi-hop cellular networks. In: 4th ACM International Symposium on Mobile ad hoc Networking & Computing, pp. 13–24. IEEE Press, New York (2003)
10. Jakobsson, M., Hubaux, J.-P., Buttyán, L.: A micro-payment scheme encouraging collaboration in multi-hop cellular networks. In: Wright, R.N. (ed.) FC 2003. LNCS, vol. 2742, pp. 15–33. Springer, Heidelberg (2003)
11. Erdős, P., Rényi, A.: On the Evolution of Random Graphs. In: Publication of the Mathematical Institute of the Hungarian Academy of Sciences, pp. 17–61 (1960)
12. Das, B., Bharghavan, V.: Routing in Ad-hoc Networks Using Minimum Connected Dominating Sets. In: IEEE International Conference on Communications, pp. 376–380. IEEE Press, New York (1997)
13. Sinha, P., Shivkumar, R., Bharghavan, V.: MCEDER: Multicast Core-Extraction Distributed Ad-hoc Routing. In: IEEE Wireless Communications and Networking Conference, pp. 1313–1317. IEEE Press, New York (1999)
14. Pandana, C., Han, Z., Liu, K.J.R.: Cooperation enforcement and learning for optimizing packet forwarding in autonomous wireless networks. Proceedings of the IEEE Transactions on Wireless Communications 7(8), 3150–3163 (2008)
15. Crosby, G.V., Pissinou, N.: Evolution of Cooperation in Multi-Class Wireless Sensor Networks. In: 32nd IEEE Conference on Local Computer Networks, pp. 489–495. IEEE Press, New York (2007)
16. Michiardi, P., Molva, R.: Prevention of Denial of Service attacks and Selfishness in Mobile Adhoc Networks. In: Mobile Ad Hoc Networks, Institute Eurecom Research Report (2002)

Synthetic Social Network Based on Competency-Based Description of Human Resources

Štěpán Kuchař[1], Jan Martinovič[2], Pavla Dráždilová[2], and Kateřina Slaninová[2]

[1] VŠB - Technical University of Ostrava
IT4Innovations
Ostrava, Czech Republic
stepan.kuchar@vsb.cz

[2] VŠB - Technical University of Ostrava
Department of Computer Science,
Ostrava, Czech Republic
{jan.martinovic,pavla.drazdilova,katerina.slaninova}@vsb.cz

Abstract. The approach presented in this paper is based on the field of human resource management with the aim to extend the analysis of human resources by a graph theory perspective with an output representation by synthetic social networks. Further analysis of human resources is focused on their division into communities with similar competencies and skills. We used betweenness concept of centrality for finding important persons in the network that share their skills and competencies with workers in other communities and can therefore serve as contact persons between communities with different skills. This method can also be used for suggesting worker team composition based on similarity of workers' skills for different roles.

Keywords: Synthetic social network, Complex network, Human resource management, Competency model.

1 Introduction

A typical social network is as a set of people, or groups of people, who socially interact among themselves [1]. In these networks the relations are usually defined by one of the types of interaction between the actors, e.g. personal knowledge of one another, friendship, membership, etc. However, in the area of synthetic social networks, we can explore the extended definition of social networks. This can be done by exploring social network as a set of people, or groups of people who have similar patterns of contacts of interactions, or generally with similar attributes [2].

This approach can then be extended to the analysis of complex networks. Complex networks, especially in the web sphere and internet areas, are often called synthetic, or derived, social networks [3]. This type of social network differs from natural social networks due to the relationship between the nodes. They are generated on the basis of the common attributes of the nodes [4]. These attributes do not necessarily represent the physical communication or the interaction among objects like in the natural social networks [5], but other attributes representing the personal similarity. The approach

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 313–324, 2013.

presented in this paper is based on these types of networks and the relations between workers based on similar skills and competencies.

The approach presented in the paper is based on the field of human resource management with the aim to extend the analysis and simulation of human resources in business processes by a graph theory perspective with an output representation by synthetic social networks (because real social relations and interactions between individual workers are not known). The relation to this special type of social networks can lead to further analysis of human resources focused on their thorough division into communities based on similar skills and competencies. This step is not possible without usage of the social network approach. Due to this reason, the issues of social network area are described and defined in this paper, including social network evaluation and community detection field. On the basis of performed analyses, the experiments which detect latent ties between particular resources are presented. The relations between human resources are defined not only by their similarity, but also by their membership in communities with similar behaviour.

Due to the fact, that the proposed approach consists of two different application areas, the structure of the paper is following. In Section *Social Network*, the area of social networks with evaluation and community detection is described. Section *Competency-based description of human resources* focuses on human resources and their competency description. This section ends with the description of connection between vector model for human resources and graph theory approach of synthetic social networks. Afterwards, Section *Experiments* with experiments is presented, which describes usage of social network analysis focused on human resource management. Social network evaluation is used for gaining new information about relations between human resources and for analysis of resources with interesting properties and behaviour.

2 Social Network

Social networking is a complex, large and expanding sector of the information economy. Researchers' interest in this field is growing rapidly. It has been studied extensively since the beginning of the 20th century. The first normative contributions in this area were proposed in 1970s by sociologist Mark Granovetter and mathematician Linton C. Freeman. The basic theory "The Strength of Weak Ties" was mentioned in 1973 [6]. Granovetter argued that within a social network, weak ties are more powerful than strong ties. Another significant principle was published in 1979 by Linton C. Freeman [7]. In his work was presented definition of centrality, which is one node's relationship to other nodes in the network. He defined basic metrics like degree, control and independence, from which reason researchers proceed in their present works.

Social network is a set of people or groups of people with similar patterns of contacts or interactions such as friendship, co-working, or information exchange [8]. The World Wide Web, citation networks, human activity on the internet, physical and biochemical networks are some examples of social networks. Social networks are usually represented by using graphs, where nodes represent individuals or groups and lines represent contacts among them. The configuration of relations among network members identifies a specific network structure, and this structure can vary from isolated structures where no members are connected to saturated structures in which everyone is interconnected.

A relationship between the actors in a social network can be very complex, often making them multidimensional. This fact leads to the formation of various types of social networks. Amongst others, we can mention multi-layered social networks (with homogeneous nodes, but with multiple relations), bipartite social networks (with two types of nodes), multi-modal social networks (with many types of nodes), temporal social networks (which reflect the network evolution), or multidimensional social networks, in which are combined a hierarchy of relations with a group hierarchy of nodes, and a time dimension [9].

Social network analysis was defined by Barry Wellman as "work at describing underlying patterns of social structure, explaining the impact of such patterns on behavior and attitudes" [10]. Therefore, researchers are not interested only on describing the different social structures, but they emphasize on investigating the consequences of this variation on the member's behaviors.

2.1 Evaluation of Social Networks

For a description of social networks defined in 1979, see Linton Freeman [7] various types of *centrality*, where individual network nodes are directly evaluated, or where the average value of selected centrality in a graph may be an item of interest.

A primary use of graph theory in social network analysis is to identify the important or prominent actors at both the individual and group levels of analysis. *Centrality* and *prestige* concepts and measures seek to quantify graph theoretic ideas about an actor's prominence within a complete network by summarizing the structural relations among all nodes. Centrality means that a prominent actor has high involvement in many relations, regardless of whether sending or receiving ties. Prestige is when a prominent actor initiates few relations but receives many directed ties. Knoke and Yang defined the above mentioned terms in [11].

Degree centrality requires the usage of matrix algebra notation. Unlike actor degree centrality, group degree centralization measures the extent to which the actors in a social network differ from one another extent to which the actors in a social network differ from one another in their individual degree centralities.

Closeness centrality was developed to reflect how near a node is to the other nodes in a social network [12]. Closeness and distance refer to how quickly an actor can interact with others, for example, by communicating directly or through very few intermediaries. An actor's closeness centrality is a function of its geodesic distance (length of the shortest path connecting the two nodes) to all other nodes.

Betweenness concept of centrality concerns how other actors control or mediate the relations between two nodes that are not directly connected. Actor betweenness centrality measures the extent to which other actors lie on the geodesic path between pairs of actors in the network.

To understand networks and their participants, we provide the location of actors in the network. Measuring the network location is finding the centrality of a node. These measures determine the various roles and groupings in a network – who are the connectors, specialists, leaders, bridges, isolates, where are the clusters and who is in them, who is in the core of the network, and who is on the periphery.

2.2 Community Detection

The discovery and analysis of community structure in networks is a topic of considerable recent interest in sociology, physics, biology and other fields. Networks are very useful as a foundation for the mathematical representation of a variety of complex systems such as biological and social systems, the Internet, the world wide web, and many others [13]. A common feature of many networks is âĂIJcommunity structureâĂİ, the tendency for vertices to divide into groups, with dense connections within groups and only sparser connections between them [14].

Newman and Girvan [15] proposed algorithms for finding and evaluating community structure in network. They used a âĂIJdivisiveâĂİ technique which iteratively removes edges from the network, thereby breaking it up in communities. The edges to be removed are identified by using one of a set of edge betweenness measures, of which the simplest is a generalization to edges of the standard shortest-path betweenness of Freeman. Than, their algorithms include a recalculation step in which betweenness scores are re-evaluated after the removal of every edge.

To detect communities, graph partitioning methods or hierarchical clustering has been applied. Originally, graph partitioning methods, based on edge removal [16], divide the vertices of a network into a given number of (non-overlapping) groups of a given size, while the number of edges between groups is minimal.

2.3 Spectral Clustering

Spectral clustering is one of the divisive clustering algorithms which can be applied in the graph theory. The spectral clustering algorithm uses eigenvalues and eigenvectors of a similarity matrix derived from the data set to find the clusters. In this section, there is described the type of spectral clustering based on the second smallest eigen vector of the Laplacian matrix.

Given a set of data points $\{x_1,\dots x_n\} \in \mathbb{R}^m$ and similarity (cosine measure) $a_{ij} \geq 0$ between all pairs of the data points x_i and x_j. Let $G = (V,E)$ be an undirected graph with vertex set $V = \{v_1,\dots,v_n\}$. Each vertex v_i in this graph represents the data point x_i. Two vertices are connected, if the similarity a_{ij} between the corresponding data points x_i and x_j is positive, and the edge is weighted by a_{ij}. The weighted adjacency matrix of the graph is the matrix $A = (a_{ij})$ $i, j = 1,\dots,n$. If $a_{ij} = 0$ than $(v_i,v_j) \notin E(G)$. It governs that A is symmetric for the undirected graph. The degree of a vertex $v_i \in V$ is defined as $d_i = \sum_{j=1}^{n} a_{ij}$. The degree matrix D is defined as the diagonal matrix with the degrees $d_1,\dots,d_n$ on the diagonal. The unnormalized graph of Laplacian matrix is defined as $L = D - A$. In[17], Fiedler defines the second smallest eigenvalue $a(G)$ of the of Laplacian matrix $L(G)$ as algebraic connectivity of the graph G. In his honor, the corresponding eigenvector is called *Fiedler vector*. The Spectral Partitioning Algorithm which uses Fiedler vector is summarized in [16]. We used algorithm for spectral clustering (Left-Right algorithm) which is described in article [18].

3 Competency-Based Description of Human Resources

The description of the employees' skills in the process is a human resources management area of expertise where the competency models [19–21] and skills frameworks (e.g. Skills Framework for the Information Age [22]) are used. Competency models define various competencies which are important for the company and its processes. Competencies are defined as sets of knowledge, abilities, skills and behaviour that contribute to successful job performance and the achievement of organizational results [21]. Skills frameworks have the same purpose, but they describe skills particular for one domain rather than general competencies. But in fact skills are just a special type of competencies.

Competency models and skills frameworks also describe how to measure and evaluate individual competencies. In most cases competencies are measured by a number of advancing stages where higher levels of competency include everything from their lower levels. The first competency model had five stages [19] and later models used the same system, but they did not keep the number of stages. There is no standard for how many stages should a competency model have and every model defines its own set of stages.

Therefore, competencies of a specific human resource can be described by the competency level acquired by the resource. This also means that this resource has mastered this given level and all lower levels of the competency. This way it is not important how many levels does the competency model have because the computing model can assume, that the highest acquired level of the best resource is also the highest level of the competency model.

Let's have a small example of one Developer working in a software development company. His competencies in a 10-level model could look as follows:

- Java development - 7. level,
- C# development - 2. level,
- UML knowledge - 4. level,
- communication - 2. level,
- customer knowledge of VSB-TUO - 4. level,
- customer knowledge of MyCompany - 0. level.

Domain specific skills (development, UML knowledge), general competencies (communication) and knowledge of the environment (customer knowledge) are contained in this example. It is clear that competencies in the model have to be based on the company requirements and professional domain.

3.1 Competency-Based Description of Process Activity Requirements

All activities in the process also have competency-based requirements that describe what competencies should the worker performing the activity know. Therefore, each activity will be defined by the set of competency levels for each required resource type entering the activity specifying that only workers with given or higher level will do the activity as planned. Resources with lower competencies are able to finish the activity, but it will take additional time to learn how to perform the activity and their work is prone to contain more errors. A simple example of requirements for the activity of developing customer specific code in the software development company follows:

- Development - 6. level,
- UML knowledge - 3. level,
- communication - 3. level,
- customer knowledge - 4. level.

If we compare this example with the worker example from previous chapter, one can notice the generalization of some requirements (development and customer knowledge). When assessing the employee's competencies, it is better to define the competency levels in specific parts of the domain so that the resources are assessed as precisely as possible. On the other hand, the activity requirements should only define a level for the whole competency category, and relevant part of the domain will be specified by actual process case. In other words, if the development company tackles with a case where they have to develop a Java code for the company VSB-TUO, then the requirements in this case will be refined as Java development and customer knowledge of VSB-TUO.

3.2 Competency Models and Synthetic Social Network

To create a synthetic social network based on the competencies of human resources, similarities between these resources had to be evaluated. This evaluation was performed using vector space model that is very often used in document searches [23]. To use this model for the competencies, a way to describe the resource competencies as vectors had to be found. This was solved by devising fragmented vector representation of the competency levels for given resource. This representation and its different properties and validation was described in our previous work (see [24]).

On the basis of created vector model, similarity matrix $M^{H \times H}$ can be constructed for the set of resources R. The matrix contains similarities of particular resources from range of values $< 0, 1 >$. It is suitable to filter vertices between resources, which are of lower importance, for construction of synthetic social network and for further finding of communities. For this purpose, the threshold λ is defined. Afterwards, it is possible to construct graph $G(R,E)$, where E represents a strength of vertices between particular resources, while weights $w \in E$ meet the constraint $w \geq \lambda$. After construction of graph G, we can find community resources with similar attributes. In the proposed approach, we use Left-Right Algorithm for community detection, described in Section *Spectral Clustering*. The output set of communities C is used in further experiments.

4 Experiments

For the experiment, we created a synthetic social network for the workers involved in a software process of a local middle-sized software development company. 8 roles were identified in the process and their possible competency level intervals were specified based on the process requirements and several selected worker profiles (competency profiles were not available for all workers in the process). Then, 143 competency profiles were created based on these constraints, each containing 19 competencies important in the software process, and each on a 10-level scale. 41 basic activities were analysed in the process and their requirements were specified for the same 19 competencies to ensure their compatibility.

The network in this experiment was created by using the similarities between competency profiles specifying individual human resources in the process. The threshold λ for creating the network was set to 0.7 to filter insignificant connections that cluttered the network.

4.1 Visualization of Detected Communities

The first added value that the created network brought was the possibility to detect communities of workers in the process based on their competencies. Communities shown in Fig. 1 were detected by the Left-Right Algorithm described in Section *Spectral Clustering*.

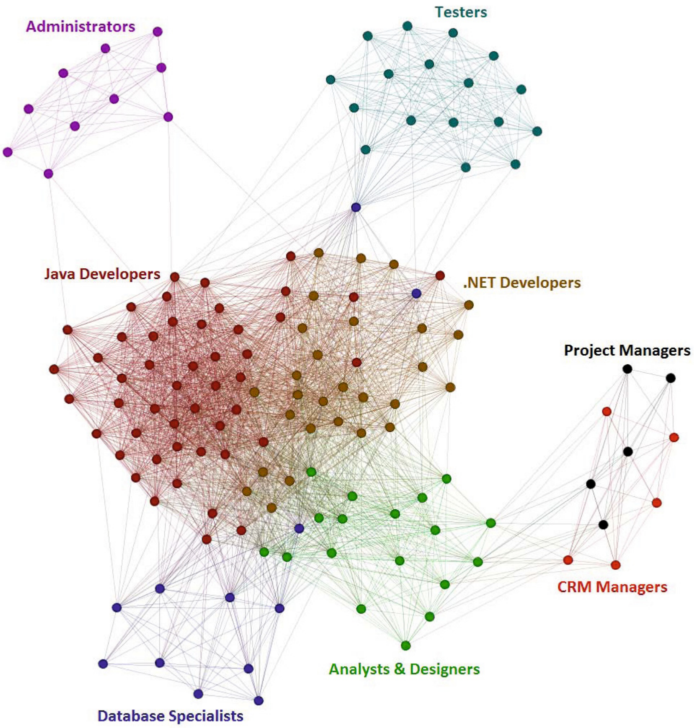


Fig. 1. Communities in the Competency-based Synthetic Social Network

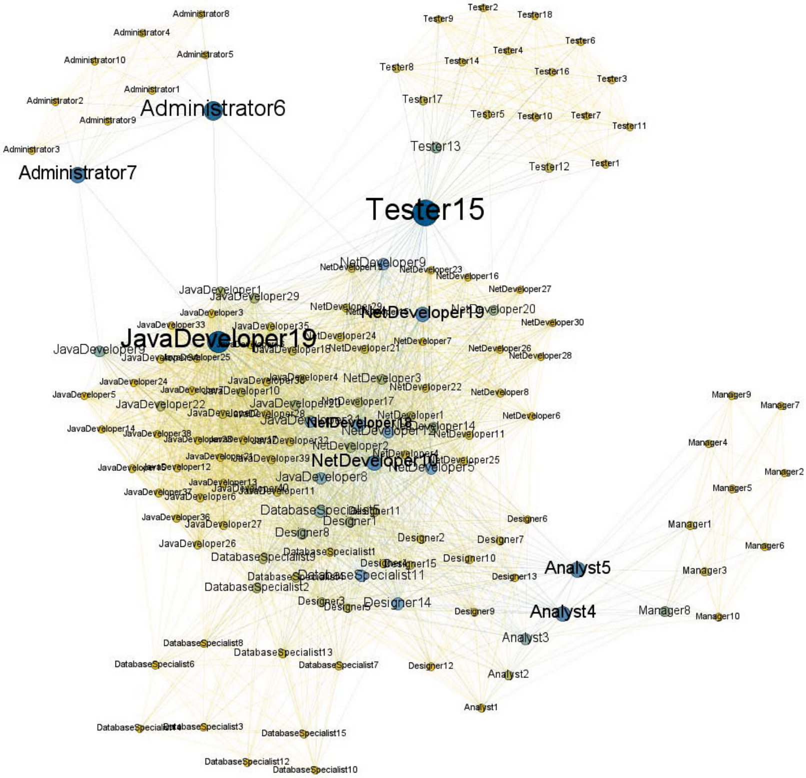


Fig. 2. Betweenness Centrality Evaluation

The algorithm detected eight distinct communities: Administrators, Analysts & Designers, CRM Managers, Database Specialists, Java Developers, .NET Developers, Project Managers and Testers. These communities are very similar to the roles in the process except for one combined community for both Analysts and Designers (showing that these two roles are very similar in their competencies) and two separate communities for two types of managers that were defined as one role in the process. Even though these roles were known prior to the experiment, it showed that this detection algorithm could be effectively used for discerning roles in the environment without predefined roles.

4.2 Evaluation of Betweenness Centrality

One of the interesting evaluation method of the social networks is betweenness centrality (see Section *Evaluation of Social Networks*) that specifies how much a node in the network links other nodes together. When considering the network based on competency similarities, betweenness identifies universal workers that have acquired knowledge from multiple disciplines. These workers can serve as communication bridges among different communities. Betweenness centrality for individual workers is proportionally displayed in Fig. 2.

Table 1. Number of Neighbours for Resources Suitable for the System Architecture Analysis-MSSQL,VSB-TUO Activity

Resource	Own community	Neighbouring communities				
	Analysts & Designers	.NET Developers	CRM Managers	Java Developers	Project Managers	Database Specialists
			Similarity threshold = 0.7			
Analyst3	17	5	2	2	2	
Analyst1	17	2	1	1		
Analyst5	17	5	3		4	1
Analyst4	16	4	4	1	3	1
			Similarity threshold = 0.75			
Analyst3	17	1	2		1	
Analyst1	9	2				
Analyst5	10	2	2		1	
Analyst4	11	2				
			Similarity threshold = 0.8			
Analyst3	5	1				
Analyst1	4					
Analyst5	6	1				
Analyst4	4					
			Similarity threshold = 0.85			
Analyst3	4					
Analyst1	1					
Analyst5	3					
Analyst4	1					

Tester15 has the biggest betweenness centrality for obvious reasons because he connects the community of Testers with the Java developers and .NET developers communities. Administrator6 and Administrator7 create a similar link between Administrators and Java and .NET developers. On the other hand, JavaDeveloper19 creates a bridge between a lot of Java developers and Administrators. Analyst4 and Analyst5 connect Managers with Analysts & Designers and other communities.

4.3 Analysis of Connections to Different Communities

All prior analyses considered the network as a whole, but very interesting results can be gained by looking at individual workers in the process. The network connects resources with similar competencies and with similar knowledge. Therefore, connected individuals can understand each other more easily because they share common knowledge and common behaviour. Analysing these connections for a specific worker can lead to finding people in other communities that could make a more effective team or that could be used for easier acquisition of additional knowledge from another part of the process. This information could also be used for choosing more appropriate worker for performing an activity because his hidden knowledge and easier collaboration could help him to understand the domain more quickly.

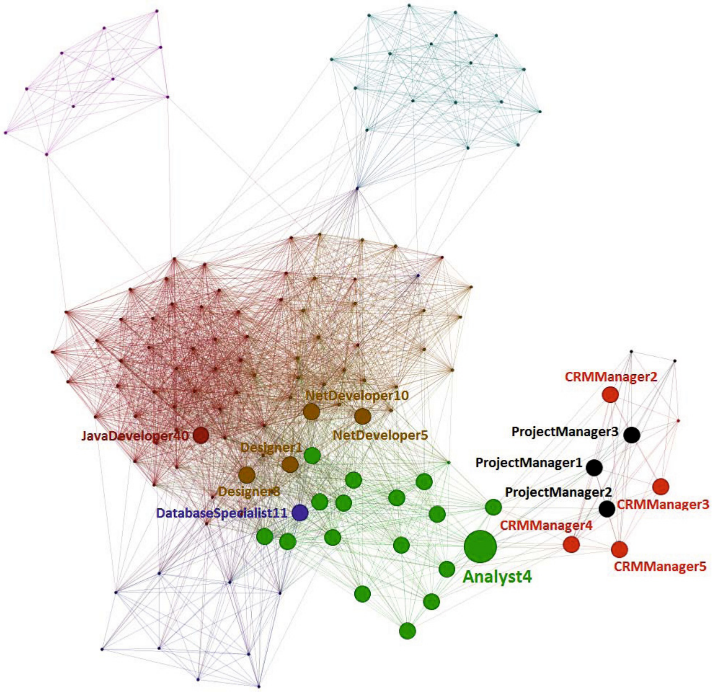


Fig. 3. Neighbours of the Analyst4 Resource

Process activities and their competency requirements play an important role in this analysis because the correct choice of appropriate worker is primarily based on these requirements. Table 1 contains the number of neighbouring resources for each resource that is suitable for the System Architecture Analysis activity for the process case specialized on MSSQL database for VSB-TUO customer.

This table is separated into several sections based on the similarity threshold that was used to look for neighbouring nodes for each resource. The resources are sorted according to their competency suitability to perform the specified activity, Analyst3 being the most suitable and Analyst4 being still able to perform the activity but having to spent more time with the activity. The neighbourhood analysis shows that even though Analyst5 is the third in suitability, his knowledge similar to one of the Database Specialists could help him overcome some specific difficulties concerning an analysis of heavily database-centric system and therefore it could be a better match for such task. On the other hand, Analyst3 could be a better match for a process case concerning Java-related specifics because he could simplify the further work on the design by providing language-related hints to the architecture analysis document.

In considering the team composition, not only number of neighbours is important, but actual neighbouring workers have to be identified. These neighbours share common knowledge and could find a better ground at understanding each other when collaborating even though they have not met before. Fig. 3 shows neighbouring workers for the Analyst4 worker.

5 Conclusion and Future Work

A new method for using the synthetic social networks in the field of allocating human resources and finding suitable representatives for other spheres of human activities was presented in this paper. The experiments proved the hypothesis that selected human resources have relation to other sources, which are oriented not only to queried resource and similar activities, but to other activities as well. Many neighbouring resources are classified into other communities and this information can be used to enhance the communication between different parts of the process. Moreover, the betweenness centrality evaluation detected resources that create bridges between communities obtained by our developed Left-Right algorithm.

In future work, presented results will be used for identifying teams that will be able to collaborate more effectively due to their common knowledge [25–27]. We intend to use extended queries by several fields and obtained knowledge about community overlapping. This means that selected worker may be important not only for his own community but that he may have relations to other communities as well.

Based on presented results, a combination of social network analysis and human resources field can provide added value for human resource allocation, collaboration and decision support processes.

Acknowledgments. This work was partially supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and by SGS, VSB – Technical University of Ostrava, Czech Republic, under the grant No. SP2013/167 Analysis of the behaviour patterns in complex networks.

References

1. Newman, M.E.J.: Networks: An Introduction. Osford University Press (2010)
2. Radicchi, F., Castellano, C., Cecconi, F., Loreto, V., Parisi, D.: Defining and identifying communities in networks (February 2004)
3. Musiałl, K., Kazienko, P.: Social networks on the internet. World Wide Web 1, 1–42 (2012)
4. Costa, L., Rodrigues, F., Travieso, G., Boas, P.: Characterization of complex networks: A survey of measurements. Advances in Physics 56(1), 167–242 (2007)
5. Bisgin, H., Agarwal, N., Xu, X.: Investigating homophily in online social networks. In: Proceedings of the 2010 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, WI-IAT 2010, pp. 533–536. IEEE Computer Society, Los Alamitos (2010)
6. Granovetter, M.S.: The Strength of Weak Ties. American Journal of Sociology 78(6), 1360–1380 (1973)
7. Freeman, L.C.: Centrality in social networks: Conceptual clarification. Social Networks 1, 215–239 (1979)
8. Garton, L., Haythornthwaite, C., Wellman, B.: Studying online social networks. Journal of Computer-Mediated Communication 3(1) (1997)
9. Kazienko, P., Musial, K., Kukla, E.z., Kajdanowicz, T., Bródka, P.: Multidimensional social network: Model and analysis. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part I. LNCS (LNAI), vol. 6922, pp. 378–387. Springer, Heidelberg (2011)

10. Knoke, D., Yang, S.: Social network analysis. Quantitative applications in the social sciences, vol. 154. Sage (2008)
11. Knoke, D., Yang, S.: Social Network Analysis, 2nd edn. Sage Publications, Inc. (2008)
12. Sabidussi, G.: The centrality index of a graph. Psychometrika 31(4), 581–603 (1966)
13. Newman, M., Barabasi, A.L., Watts, D.J.: The Structure and Dynamics of Networks (Princeton Studies in Complexity). Princeton University Press, Princeton (2006)
14. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proceedings of the National Academy of Sciences of the United States of America 99(12), 7821–7826 (2002)
15. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Physical Review E - Statistical, Nonlinear and Soft Matter Physics 69(2 pt. 2), 16 (2004)
16. Pothen, A., Simon, H.D., Liou, K.P.: Partitioning sparse matrices with wigenvectors of graphs. SIAM J. Matrix Anal. Appl. 11(3), 430–452 (1990)
17. Fiedler, M.: Algebraic connectivity of graphs. Czechoslovak Mathematical Journal 23, 298–305 (1973)
18. Drázdilová, P., Martinovic, J., Slaninová, K.: Spectral clustering: Left-right-oscillate algorithm for detecting communities. In: ADBIS Workshops, pp. 285–294 (2012)
19. Dreyfus, S.E., Dreyfus, H.L.: A five-stage model of the mental activities involved in directed skill acquisition. Technical report, DTIC Document (1980)
20. Ennis, M.R.: Competency models: a review of the literature and the role of the employment and training administration (ETA). US Department of Labor (2008)
21. Sinnott, G., Madison, G., Pataki, G.: Competencies: Report of the competencies workgroup, workforce and succession planning work groups (September 2002)
22. SFIA Foundation: Framework reference SFIA version 4G (2010)
23. Berry, M.W.: Survey of text mining: clustering, classification, and retrieval, vol. 1. Springer-Verlag New York Inc. (2004)
24. Kuchar, S., Martinovic, J.: Human Resource Allocation in Process Simulations Based on Competency Vectors. AISC, vol. 188. Springer, Heidelberg (2013)
25. Fitzpatrick, E.L., Askin, R.G.: Forming effective worker teams with multi-functional skill requirements. Computers & Industrial Engineering 48(3), 593–608 (2005)
26. Karduck, A., Sienou, A.: Forming the optimal team of experts for collaborative work. In: Bramer, M., Devedzic, V. (eds.) Artificial Intelligence Applications and Innovations, IFIP 18th World Computer Congress, TC12 First International Conference on Artificial Intelligence Applications and Innovations (AIAI 2004), August 22-27, pp. 267–278. Kluwer, Toulouse (2004)
27. Wi, H., Oh, S., Mun, J., Jung, M.: A team formation model based on knowledge and collaboration. Expert Systems with Applications 36(5), 9121–9134 (2009)

Displaying Genealogy with Adoptions and Multiple Remarriages Using the WHIteBasE

Seiji Sugiyama[1], Atsushi Ikuta[2], Daisuke Yokozawa[2], Miyuki Shibata[2], and Tohru Matsuura[3]

[1] Info. Science & Engineering, Ritsumeikan University, Kusatsu, Shiga, Japan
seijisan@is.ritsumei.ac.jp
[2] Human Informatics, Otani University, Kyoto, Japan
a.ikuta@sch.otani.ac.jp, dyokozawa@gmail.com, neko@res.otani.ac.jp
[3] Hokkaido University Hospital, Sapporo, Hokkaido, Japan
macchan@med.hokudai.ac.jp

Abstract. In this research, needs of displaying genealogy with various family relations are described and those solutions using our WHIteBasE method are proposed. Previous WHIteBasE method has perfectly been able to integrate each relation that includes a married couple and their children, and has been able to display complex relations with segment intersections easily. It has also been added that Genealogy with Direct Segments (DS), Genealogy with Hooked Segments (HS) and Annotation Data Always Displayed (ADAD). In this paper, two new functions are added to the WHIteBasE method. One is 'Genealogy with Adoptions'. Using 'Adopted Segments (AS)', not only biological family relations but also social family relations can be displayed simultaneously. The other is 'Genealogy with Multiple Remarriages'. Using 'Double Bends (DB)', crossing relations more than 3x3 both sexes can be displayed perfectly. Our improved software that can display AS and DB automatically and seamlessly by only mouse operation is presented.

Keywords: Adoptions, Multiple Remarriages, Hidden Node, Segment Intersections, Search Algorithm, Free Layout.

1 Introduction

There is a requirement to display complex relations in family trees. However, it cannot be displayed perfectly in existing genealogy display software [1]–[20]. When complex relations are inputted in the software, one individual is often displayed in multiple places automatically. It is thought that almost all of them consider only simple family trees and have no idea to display complex layout. As a result, users must reconstruct the scattered individual placements into only one relation in their mind, though it is graphical display software. On the other hand, 3D graphics is rather more difficult to understand complex relations than 2D because many names and segments overlap [21]. GEDCOM [22], a de facto standard for recording genealogy data exchange format, is not enough to display complex relations because it considers no layout information.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 325–336, 2013.

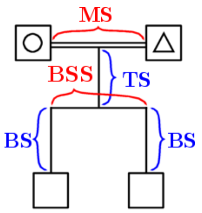


Fig. 1. Segment Names for Japanese Regular Layout

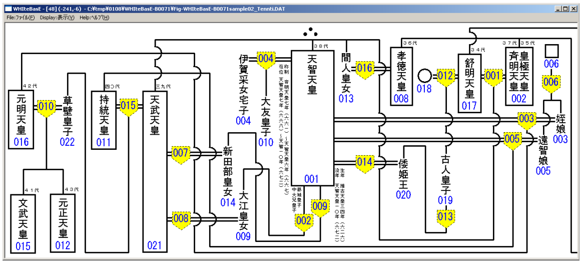


Fig. 2. Our Previous Software [24]

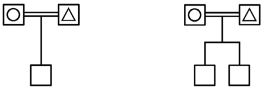

(a) a child (b) children

Fig. 3. Regular Layout

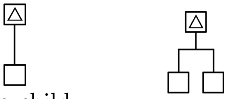

(a) a child (b) children

Fig. 4. DS from a Single Parent

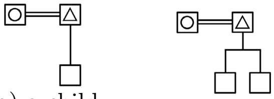

(a) a child (b) children

Fig. 5. DS from Parents

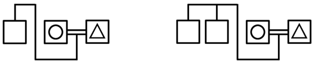

(a) a child (b) children

Fig. 6. HS from MS

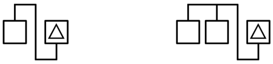

(a) a child (b) children

Fig. 7. HS from a Single Parent

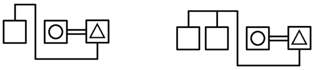

(a) a child (b) children

Fig. 8. HS from Parents

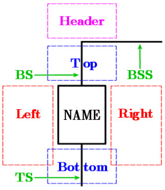


Fig. 9. Five Areas

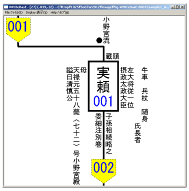
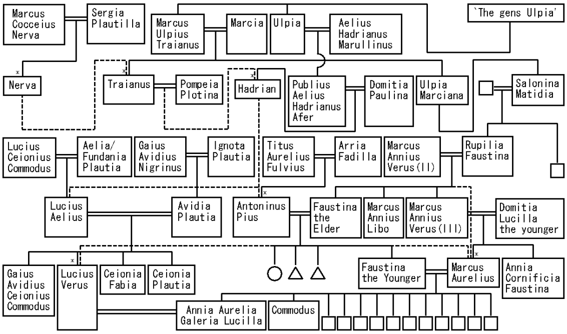


Fig. 10. ADAD

New software has been constructed in our research so that it can display complex relations [23]. Our software uses an event oriented data management method, “WHIteBasE” (Widespread Hands to InTErconnect BASic Elements). It is a hidden node for integrating relations that include a married couple and their children. If WHIteBasE is used, one individual can be displayed only once because complex relations with segment intersections can be displayed. Intuitive inputs and inspections such as map image display systems can be realized.

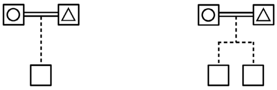
In our previous software, the regular Japanese layout style has been used as shown in Fig. 1. It includes a double horizontal segment MS (Marriage Segment), a vertical segment TS (Trunk Segment), a horizontal segment BSS (Brothers and Sisters Segment), and a vertical segment BS (Branch Segment). The ‘△’ denotes a male and the ‘◯’ denotes a female that are a couple connected by using MS. Children are connected from the intermediate of MS by using TS, BSS and BS.

Fig. 2 shows our previous software [24]. It has been constructed by using not only ‘Regular Layout’ as shown in Fig. 3 but also ‘Genealogy with Direct Segments (DS)’ as shown in Figs. 4–5, ‘Genealogy with Hooked Segments (HS)’ as shown in Figs. 6–8, and ‘Annotation Data Always Displayed (ADAD)’ as shown



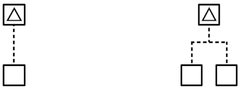
Solid Segments: Biological Relations
Dashed Segments: Social Family Relations

Fig. 11. Sample of Genealogy with Adoptions [25]



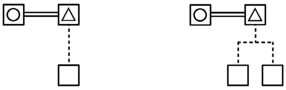
(a) a child (b) children

Fig. 12. AS from MS



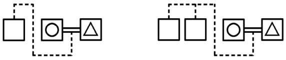
(a) a child (b) children

Fig. 13. AS from a Single Parent



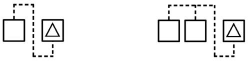
(a) a child (b) children

Fig. 14. AS from Parents



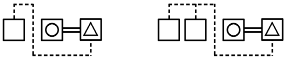
(a) a child (b) children

Fig. 15. AS from MS in HS



(a) a child (b) children

Fig. 16. AS from a Single Parent in HS



(a) a child (b) children

Fig. 17. AS from Parents in HS

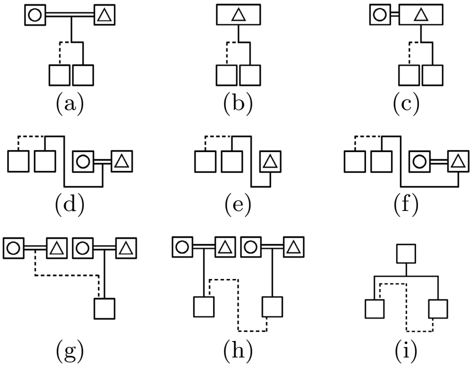


Fig. 18. Biological Relations and Adoptions

in Fig. 9–10. Regular layout, DS, HS and ADAD can be displayed automatically and seamlessly by only mouse operations.

In this paper, needs of two more different family relations are described and those solutions are proposed. One is 'Genealogy with Adoptions' such as shown in Fig. 11 [25]. Not only 'Biological Relations' showing solid segments but also 'Social Family Relations' such as adoptions showing dashed segments are often written simultaneously. It is necessary to display these dashed segments named 'Adopted Segments (AS)' for connecting adoptions. Connections of AS from parent(s) to a child/children have three types as the following:

(A) **AS from MS:** One married couple is connected with a child/children from the intermediate of MS as shown in Figs. 12(a),(b).
(B) **AS from a Single Parent:** Only a single parent is connected with a child/children directly without using MS as shown in Figs. 13(a),(b).
(C) **AS from Parents:** Only a single parent that has the other parent using MS is connected with a child/children directly as shown in Figs. 14(a),(b).

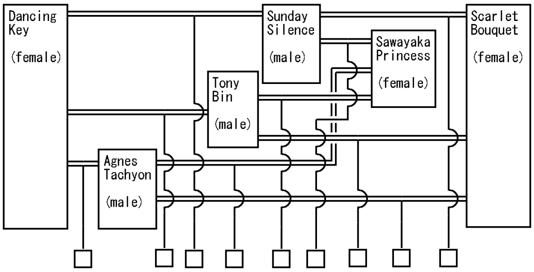
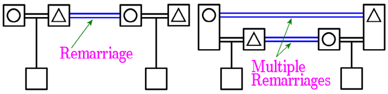
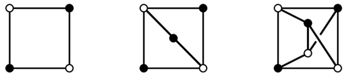


Fig. 19. Sample of Multiple Remarriages [26]



(a) Regular (b) Multiple

Fig. 20. Type of Remarriages



(a) 2x2 (b) 2x3 (c) 3x3

Fig. 21. Connection Models

On the other hand, connections of AS from parent(s) to a child/children in HS have three types as the following:

(a) **AS from MS in HS:** One married couple is connected with a child/children from the intermediate of MS as shown in Figs. 15(a),(b).

(b) **AS from a Single Parent in HS:** Only a single parent is connected with a child/children directly without using MS as shown in Figs. 16(a),(b).

(c) **AS from Parents in HS:** Only a single parent that has the other parent using MS is connected with a child/children directly as shown in Figs. 17(a),(b).

The (A)–(C) and (a)–(c) are similar to DS and HS respectively (Fig. 3–8). If all descendants in a family are adoptions, the difference is only a change of segment types from solid segments to dashed segments. On the other hand, if a family has children including both biological relations and adoptions, the segment type of TS becomes the solid segment, and both solid segments and dashed segments are used on the way to the children as shown in Figs. 18(a)–(f). If there are two families, one is biological relations and the other is adoptions, and/or if there is a connection from one of brothers and sisters to others, both solid segments and dashed segments in HS are used as shown in Figs. 18(g)–(i).

The other is 'Genealogy with Multiple Remarriages' such as mating of the racehorses shown in Fig. 19. In this case, a bloodline is important. As a result, there is a case that relations including only several horses grow very powerful, because of focusing on a strong horse. Connections of remarriages have two types as the following:

(I) **Regular Remarriages:** A married person marries again after her husband or his wife has died or after being divorced as shown in Fig. 20(a). In this case, there is no closed areas by MS and individuals' text boxes.

(II) **Multiple Remarriages:** All persons of plural couples marry again by replacing each other as shown in Fig. 20(b). In this case, there are closed areas.

Closed areas cause crossing of MS (Fig. 19). In our previous research, no crossing of MS has been considered, and the horizontal MS style has been only defined. To cope with the difficulty, 'Double Bends (DB)' for crossing of MS is added.

Fig. 21(a) shows the connection model of the minimum unit of closed areas by 2x2 partners (two males and two females). The '•' and the '◦' denote a sexual difference. Figs. 22(a)–(d) show genealogy style of the connections 2x2. In this case,

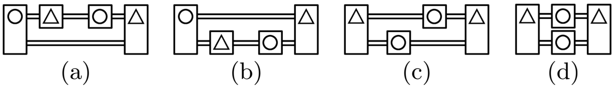


Fig. 22. Connections 2x2

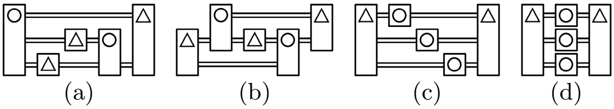


Fig. 23. Connections 2x3

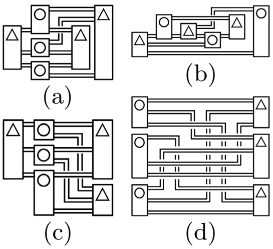


Fig. 24. Connections 3x3

no crossing of MS is occurred. Fig. 21(b) shows the connection model of closed areas by 2x3 partners. Figs. 23(a)–(d) show genealogy style of the connections 2x3. In this case, no crossing of MS is also occurred. On the other hand, Fig. 21(c) shows the connection model of closed areas by 3x3 partners. Figs. 24(a)–(d) show genealogy style of the connections 3x3. In this case, a crossing of MS is occurred and it is necessary to display DB.

These kinds of family relations including AS and DB cannot be displayed in the existing software. Because almost all of them focus on no adoptions and no multiple remarriages. Therefore, it is necessary to construct new software.

In this research, an upgrade, adding two kinds of family relations that use AS and DB to our previous genealogy display software by using the WHIteBasE method, is proposed. Our improved software that can also display AS and DB automatically and seamlessly by only mouse operations is presented.

2 WHIteBasE

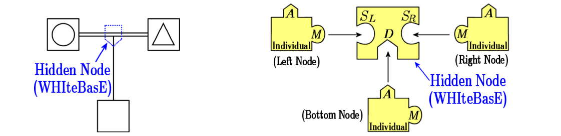
In this section, the WHIteBasE method that is our previous proposal [23][24] is briefly introduced. A relation between a married couple and their child is managed as an event by a Hidden Node, WHIteBasE as shown in Fig. 25(a). The connection model using WHIteBasE is shown in Fig. 25(b). WHIteBasE has three keyholes, S_L, S_R (Substance) and D (Descendant). Individuals have two keys, A (Ascendant) and M (Marriage). A can connect with D, and M can connect with S_L or S_R, where denote one family.

For plural children as shown in Fig. 26(a), D is extended in multiple keyholes D_j as shown in Fig. 26(b). For multiple marriages as shown in Fig. 27(a), M is extended in multiple keys M_k (Fig. 27(b)) and plural WHIteBasEs are used. That can define all of biological relations perfectly.

A set of W_i that defines WHIteBasEs and a set of I_j that defines Individual Nodes are represented by

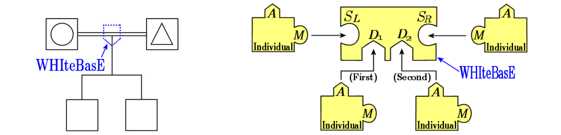
$$\begin{aligned} W_i &= \{S_L, S_R, D_j, \mathbf{Q}\} \\ I_j &= \{A, M_k\} \end{aligned} \quad \begin{cases} i = 0, 1, \cdots, i_{max} \\ j = 0, 1, \cdots, j_{max} \\ k = 0, 1, \cdots, k_{max} \end{cases} \tag{1}$$

where i, j, k denote IDs on the data table respectively, $i_{max}, j_{max}, k_{max}$ are the maximum values, S_L, S_R denote IDs of a couple, D_j denotes ID of descendants,



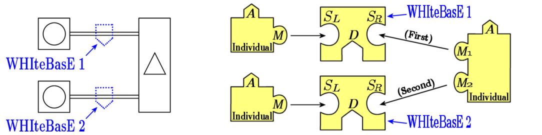
(a) Genealogy Style (b) Model of WHIteBasE

Fig. 25. Connection for a married couple and a child



(a) Genealogy Style (b) Model of WHIteBasE

Fig. 26. Connection for a married couple and children



(a) Genealogy Style (b) Model of WHIteBasE

Fig. 27. Connection for multiple marriages

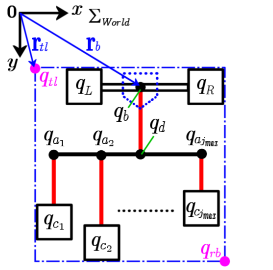
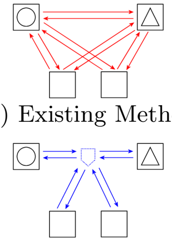


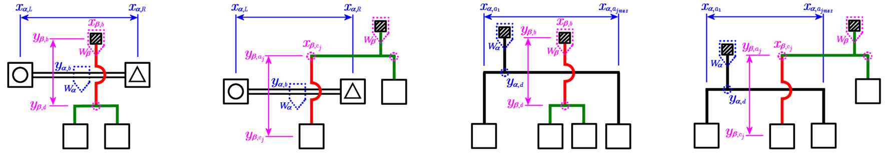
Fig. 28. Coordinate System



(a) Existing Method

(b) WHIteBasE Method

Fig. 29. Number of references



(a) Pattern 1 (b) Pattern 2 (c) Pattern 3 (d) Pattern 4

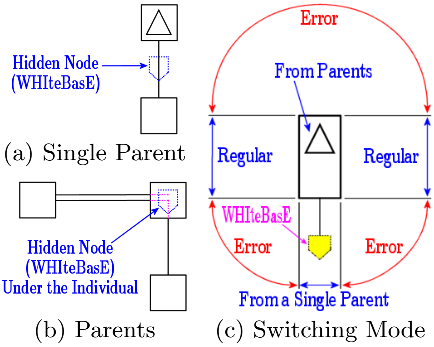
Fig. 30. Search Method for Segment Intersections

A denotes ID of an ascendant WHIteBasE, and M_k denotes ID of WHIteBasEs for handling marriages. Individuals are managed by using data table including names and annotation data. WHIteBasEs are managed by using data table separated from Individuals. Remarriages are managed by using plural WHIteBasEs. $\mathbf{Q}$ denotes a set of coordinate values of each position managed by a WHIteBasE measured from the origin of the displaying area (Fig. 28) and is represented by

$$\mathbf{Q} = \{q_b, q_L, q_R, q_d, q_{a_j}, q_{c_j}, q_{tl}, q_{rb}\}\,. \tag{2}$$

where q_b denotes a WHIteBasE's position, q_L, q_R denote the parents' positions, q_d denotes a junction's position between MS and TS, q_{c_j} denotes children's positions, q_{a_j} denotes junctions' positions between BSS and BS, q_{tl}, q_{rb} denote positions of top-left and bottom-right of all area managed by a WHIteBasE.

One of advantages using WHIteBasE is the decreased reference volume. Using the existing software, all of individuals connect with other individuals (Fig. 29(a)). On the other hand, using the WHIteBasE method, two reference links per a child decrease (Fig. 29(b)). Moreover, the user can understand the complex relations intuitively and can input and inspect them easily.

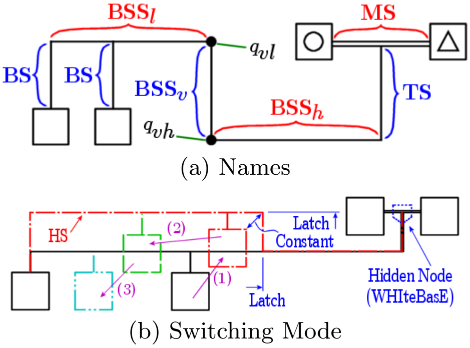


(a) Single Parent

(b) Parents

(c) Switching Mode

Fig. 31. Genealogy with Direct Segments



(a) Names

(b) Switching Mode

Fig. 32. Genealogy with Hooked Segments

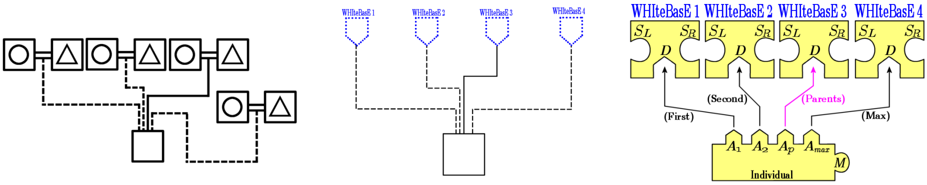
Using segment intersections is necessary for displaying complex relations. If only one WHIteBasE is used, there is no segment intersections. On the other hand, if plural WHIteBasEs are used and horizontal segments (MS, BSS) and vertical segments (TS, BS) are displayed, the positions of segment intersections can be calculated by using only four patterns of line crossing as the following:

(a) MS of W_α and TS of W_β are crossing (Fig. 30(a))
If $x_{\alpha,L} < x_{\beta,b} < x_{\alpha,R}$ & $y_{\beta,b} < y_{\alpha,b} < y_{\beta,d}$, the intersection is $(x_{\beta,b}, y_{\alpha,b})$.
(b) MS of W_α and BS of W_β are crossing (Fig. 30(b))
If $x_{\alpha,L} < x_{\beta,c_j} < x_{\alpha,R}$ & $y_{\beta,a_j} < y_{\alpha,b} < y_{\beta,c_j}$, the intersection is $(x_{\beta,c_j}, y_{\alpha,b})$.
(c) BSS of W_α and TS of W_β are crossing (Fig. 30(c))
If $x_{\alpha,a_1} < x_{\beta,b} < x_{\alpha,a_{j_{max}}}$ & $y_{\beta,b} < y_{\alpha,d} < y_{\beta,d}$, the intersection is $(x_{\beta,b}, y_{\alpha,d})$.
(d) BSS of W_α and BS of W_β are crossing (Fig. 30(d))
If $x_{\alpha,a_1} < x_{\beta,c_j} < x_{\alpha,a_{j_{max}}}$ & $y_{\beta,a_j} < y_{\alpha,d} < y_{\beta,c_j}$, the intersection is $(x_{\beta,c_j}, y_{\alpha,d})$.

The four patterns are only calculated while only two WHIteBasEs' management rectangle q_{tl}, q_{rb} are overlapped. Therefore, this search method is faster than checking all segments.

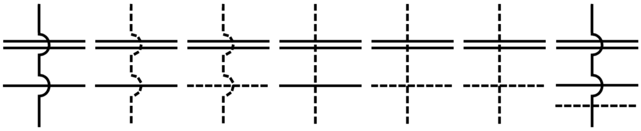
Genealogy with Direct Segments (DS) is necessary for connecting parent(s) and their child directly. DS from a Single Parent is used while WHIteBasE lies under a parent (Fig. 31(a)). DS from Parents is used while WHIteBasE overlapped one of parents (Fig. 31(b)). Two kinds of DS are automatically switched according to the positions only (Fig. 31(c)) without changing the connection model [24]. Genealogy with Hooked segment (HS) is necessary for connecting parents and their children in parallel layouts. BSS as shown in Fig. 1 is extended to three parts; BSS$_l$, BSS$_v$, and BSS$_h$ (Fig. 32(a)). For calculating HS, positions q_{vl}, q_{vh} are added to Eq. (2). Occurring HS is automatically switched according to the positions only (Fig. 32(b)) [24].

The search method of segment intersections for DS and HS has been proposed in our previous paper [24]. Therefore, the figures of search patterns are omitted in this paper. Only nine search patterns are used for DS and HS because there are three horizontal segments and three vertical segments.



(a) Genealogy Style (b) Hidden Node Connection (c) Model of WHIteBasE

Fig. 33. Connection Model for Genealogy with Adoptions



(a) (b) (c) (d) (e) (f) (g)

Fig. 34. Segment Intersections using AS

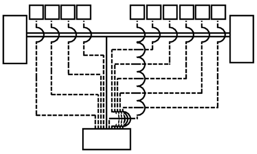
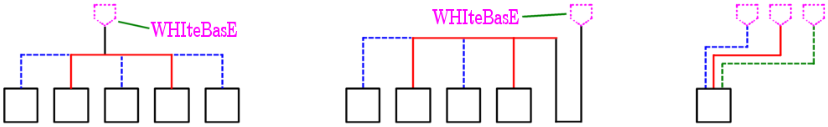


Fig. 35. AS and Arc



(a) Regular Layout (b) Using HS (c) Multiple

Fig. 36. Displaying Method for AS

3 Genealogy with Adoptions

Adoptions have not only biological parents but also other plural parents as shown in Figs. 33(a),(b). Therefore, the ascendant key A in the Individual is extended in multiple keys A_l $(l = 0, 1, \cdots, p, \cdots, l_{max})$ as shown in Fig. 33(c), where l denotes the IDs on the data table, p denotes the ID of biological parents' WHIteBasE, l_{max} denotes the maximum value, respectively. If $l = p$, use the solid segments. If $l \neq p$, use the dashed segments AS.

Consider the arc style of segment intersections in the case of using AS. It is necessary to use arcs on the intersection between solid horizontal segments and solid vertical segments as shown in Fig. 34(a). It can be displayed to use arcs on the intersection between solid horizontal segments and dashed vertical segments as shown in Fig. 34(b). However, the dashed segments are broken up if arcs are used in the intersection between dashed horizontal segments and dashed vertical segments as shown in Fig. 34(c). In addition, it is difficult to display arcs in the narrow area, especially near the area connecting with AS as shown in Fig. 35. Therefore, it is useful that no arcs is used for crossing by using AS as shown in Figs. 34(d)–(g).

The search method of segment intersections for AS is very simple because AS has no arcs. Classify only solid segment sections and dashed segment sections as shown in Figs. 36(a),(b). Then, search the segment intersections for only solid sections. The connecting points of AS on the top of Individual text box is line up in order to locate WHIteBasE's horizontal positions as shown in Fig. 36(c).

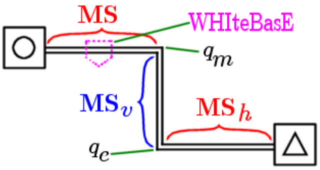


Fig. 37. Names for Double Bend

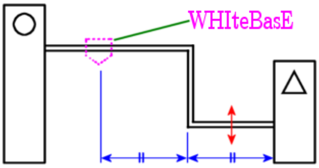


Fig. 38. ON/OFF for Double Bend

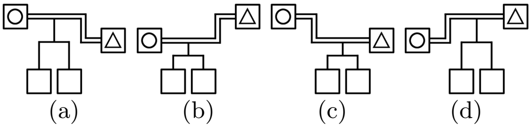


Fig. 39. Children from Double Bend

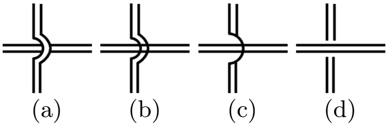


Fig. 40. Intersection style of DB

4 Genealogy with Multiple Remarriages

'Double Bend (DB)' is necessary for displaying multiple remarriages more than 3x3 partners in 2D. Therefore, MS is extended as shown in Fig. 37. In this case, define that MS denotes the section that WHIteBasE exists, MS$_v$ denotes the vertical section, and MS$_h$ denotes the horizontal section that no WHIteBasE exists. For calculating DB, positions q_m, q_e are added to Eq. (2). Finally, $\mathbf{Q}$ is represented by

$$\mathbf{Q} = \{q_b, q_L, q_R, q_d, q_{a_j}, q_{c_j}, q_{vl}, q_{vh}, q_m, q_e, q_{tl}, q_{rb}\} \,. \tag{3}$$

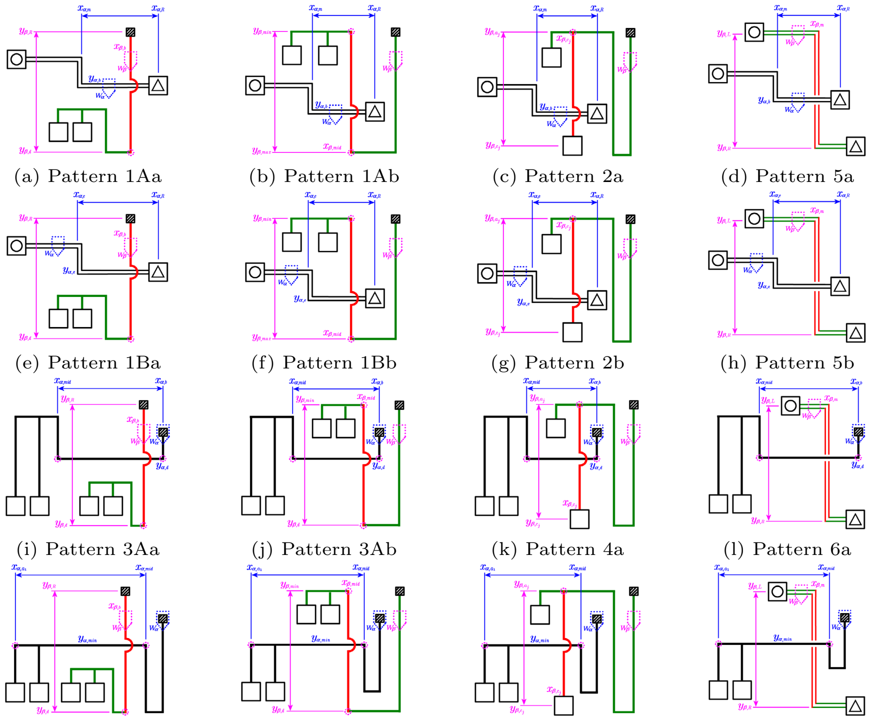
For seamless mouse operations, set the movable range between WHIteBasE and Individual as shown in Fig. 38. When mouse drag vertically in the movable range, MS becomes DB. MS$_h$ can be arbitrarily set both upper side and lower side. In the first dragging operation, the bending positions occurs at the center between WHIteBasE an Individual. After that, MS$_v$ can be arbitrarily moved by using mouse drag operation on the section from WHIteBasE to Individual. If MS$_h$ approaches MS, DB return to MS automatically.

WHIteBasE only exists on the section MS when DB is used. Therefore, the connection styles of children from DB have only four patterns as shown in Figs. 39. Provided that DB and BSS do not overlap.

The intersection style of DB, that is, crossing between double horizontal segments and double vertical segments, has not been defined in custom. Therefore, some crossing styles can be thought such as shown in Figs. 40(a)–(d). For simple viewing, Fig. 40(d) is used in this research.

5 Search Method of Segment intersections

Using not only regular layout but also AS and DB, there are four horizontal segments and four vertical segments. The positions of segment intersections can be calculated by using only 16 patterns of line crossing as the following:



(a) Pattern 1Aa (b) Pattern 1Ab (c) Pattern 2a (d) Pattern 5a

(e) Pattern 1Ba (f) Pattern 1Bb (g) Pattern 2b (h) Pattern 5b

(i) Pattern 3Aa (j) Pattern 3Ab (k) Pattern 4a (l) Pattern 6a

(m) Pattern 3Ba (n) Pattern 3Bb (o) Pattern 4b (p) Pattern 6b

Fig. 41. Search Pattern of segment intersections including Double Bend

(a) MS of W_α and TS of W_β are crossing (Fig. 41(a))
If $x_{\alpha,m} < x_{\beta,b} < x_{\alpha,R}$ & $y_{\beta,R} < y_{\alpha,b} < y_{\beta,d}$, the intersection is $(x_{\beta,b}, y_{\alpha,b})$.

(b) MS of W_α and BSS$_v$ of W_β are crossing (Fig. 41(b))
If $x_{\alpha,m} < x_{\beta,mid} < x_{\alpha,R}$ & $y_{\beta,min} < y_{\alpha,b} < y_{\beta,max}$, the intersection is $(x_{\beta,mid}, y_{\alpha,b})$[1].

(c) MS of W_α and BS of W_β are crossing (Fig. 41(c))
If $x_{\alpha,m} < x_{\beta,c_j} < x_{\alpha,R}$ & $y_{\beta,a_j} < y_{\alpha,b} < y_{\beta,c_j}$, the intersection is $(x_{\beta,c_j}, y_{\alpha,b})$.

(d) MS of W_α and MS$_v$ of W_β are crossing (Fig. 41(d))
If $x_{\alpha,m} < x_{\beta,m} < x_{\alpha,R}$ & $y_{\beta,L} < y_{\alpha,b} < y_{\beta,R}$, the intersection is $(x_{\beta,m}, y_{\alpha,b})$.

(e) MS$_e$ of W_α and TS of W_β are crossing (Fig. 41(e))
If $x_{\alpha,e} < x_{\beta,b} < x_{\alpha,R}$ & $y_{\beta,R} < y_{\alpha,e} < y_{\beta,d}$, the intersection is $(x_{\beta,b}, y_{\alpha,e})$.

(f) MS$_e$ of W_α and BSS$_v$ of W_β are crossing (Fig. 41(f))
If $x_{\alpha,e} < x_{\beta,mid} < x_{\alpha,R}$ & $y_{\beta,min} < y_{\alpha,e} < y_{\beta,max}$, the intersection is $(x_{\beta,mid}, y_{\alpha,e})$.

(g) MS$_e$ of W_α and BS ofW_β are crossing (Fig. 41(g))
If $x_{\alpha,e} < x_{\beta,c_j} < x_{\alpha,R}$ & $y_{\beta,a_j} < y_{\alpha,e} < y_{\beta,c_j}$, the intersection is $(x_{\beta,c_j}, y_{\alpha,e})$.

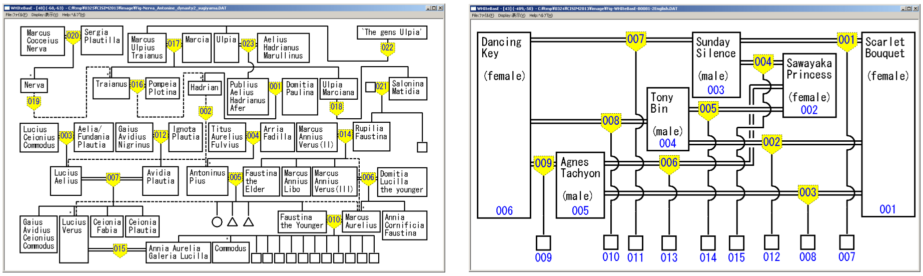
(h) MS$_e$ of W_α and MS$_v$ of W_β are crossing (Fig. 41(h))
If $x_{\alpha,e} < x_{\beta,m} < x_{\alpha,R}$ & $y_{\beta,L} < y_{\alpha,e} < y_{\beta,R}$, the intersection is $(x_{\beta,m}, y_{\alpha,e})$.

(i) BSS$_h$ of W_α and TS of W_β are crossing (Fig. 41(i))
If $x_{\alpha,mid} < x_{\beta,b} < x_{\alpha,b}$ & $y_{\beta,R} < y_{\alpha,d} < y_{\beta,d}$, the intersection is $(x_{\beta,b}, y_{\alpha,d})$.

(j) BSS$_h$ of W_α and BSS$_v$ of W_β are crossing (Fig. 41(j))
If $x_{\alpha,mid} < x_{\beta,mid} < x_{\alpha,b}$ & $y_{\beta,min} < y_{\alpha,d} < y_{\beta,d}$, the intersection is $(x_{\beta,mid}, y_{\alpha,d})$.

(k) BSS$_h$ of W_α and BS of W_β are crossing (Fig. 41(k))
If $x_{\alpha,mid} < x_{\beta,c_j} < x_{\alpha,b}$ & $y_{\beta,a_j} < y_{\alpha,d} < y_{\beta,c_j}$, the intersection is $(x_{\beta,c_j}, y_{\alpha,d})$.

[1] *mid* denotes q_{vl} or q_{vh}, *min* denotes q_{vl}, and *max* denotes q_{vh}.



(a) Sample of Adoptions (b) Sample of Multiple Remarriages

Fig. 42. Our New Genealogy Display Software

(l) BSS$_h$ of W_α and MS$_v$ of W_β are crossing (Fig. 41(l))
If $x_{\alpha,mid} < x_{\beta,m} < x_{\alpha,b}$ & $y_{\beta,L} < y_{\alpha,d} < y_{\beta,R}$, the intersection is $(x_{\beta,m}, y_{\alpha,d})$.

(m) BSS$_l$ of W_α and TS of W_β are crossing (Fig. 41(m))
If $x_{\alpha,a_1} < x_{\beta,b} < x_{\alpha,mid}$ & $y_{\beta,R} < y_{\alpha,min} < y_{\beta,d}$, the intersection is $(x_{\beta,b}, y_{\alpha,min})$.

(n) BSS$_l$ of W_α and BSS$_v$ of W_β are crossing (Fig. 41(n))
If $x_{\alpha,a_1}<x_{\beta,mid}<x_{\alpha,mid}$ & $y_{\beta,min}<y_{\alpha,min}<y_{\beta,d}$, the intersection is $(x_{\beta,mid}, y_{\alpha,min})$.

(o) BSS$_l$ of W_α and BS of W_β are crossing (Fig. 41(o))
If $x_{\alpha,a_1}<x_{\beta,c_j}<x_{\alpha,mid}$ & $y_{\beta,a_j}<y_{\alpha,min}<y_{\beta,c_j}$, the intersection is $(x_{\beta,c_j}, y_{\alpha,min})$.

(p) BSS$_l$ of W_α and MS$_v$ of W_β are crossing (Fig. 41(p))
If $x_{\alpha,a_1}<x_{\beta,m}<x_{\alpha,mid}$ & $y_{\beta,L}<y_{\alpha,min}<y_{\beta,R}$, the intersection is $(x_{\beta,m}, y_{\alpha,min})$.

If all children of W_β are adoptions, (a),(b),(e),(f),(i),(j),(m) and (n) are unnecessary to search. If all children of W_α are adoptions, (i)-(p) are unnecessary. If the target child of W_β is adoptions, (c),(g),(k) and (o) are unnecessary.

6 Demonstration of Our New Software

Figs. 42(a),(b) show the sample demonstrations of our new genealogy display software that can be display Adoptions in (a) and Multiple Remarriages in (b) respectively with segment intersections. Only changing from solid segments to dashed segments can realize adoptions display by using not only regular layout but also DS and HS. Using DB can realize multiple remarriages display. A lot of individuals with complex relations including adoptions could be displayed in narrow area using one window. Even if the client area is filled, scrolling mouse wheel, the zoom rate can be changed and displaying area beyond the window size can be used. Using our method, genealogy required in a favorite rectangle size can be displayed easily. Even if AS and DB are used, segment intersections could be displayed automatically and seamlessly by only mouse operation.

7 Conclusion

In this research, Genealogy with Adoptions and Multiple Remarriages could be constructed by using the WHIteBasE method. Future research will be conducted to construct automated layouts, generation search, thinned-out individuals, grid layouts, improving GUI, etc. *This research has received the assistance of the "Shin Buddhist Comprehensive Research Institute, Otani University, Japan".*

References

1. Sugito, S.: “Alliance”, news letter of Oceania conference, vol. (86), pp. 10–37 (2006) (in Japanese)
2. The Generations Network. Genealogy, Family Trees and Family History Records on line, http://ancestry.com
3. MyHeritage, http://myheritage.jp
4. Jurek Software, http://www.pedigree-draw.com/
5. He, M., Li, W.: PediDraw: A web-based tool for drawing a pedigree in genetic counseling. In: BMC Medical Genetics, pp. 1–4 (2007)
6. Brun-samarcq, L., et al.: CoPE: a collaborative pedigree drawing environment. Bioinformatics ‘Applications Note’ 15(4), 345–346 (1999)
7. Dudbridge, F., et al.: Pelican: pedigree editor for linkage computer analysis. Bioinformatics ‘Applications Note’ 20(14), 2327–2328 (2004)
8. Trager, E.H., et al.: Madeline 2.0 PDE: a new program for local and web-based pedigree drawing. Bioinformatics ‘Applications Note’ 23(14), 1854–1856 (2007)
9. Makinen, V.P., et al.: High-throughput pedigree drawing. European Journal of Human Genetics 13, 987–989 (2005)
10. Mancosu, G., Ledda, G., Melis, P.M.: PedNavigator: a pedigree drawing servlet for large and inbred populations. Bioinformatics ‘Applications Note’ 19(5), 669–670 (2003)
11. Tores, F., Barillot, E.: The art of pedigree drawing: algorithmic aspects. Bioinformatics 17(2), 174–179 (2001)
12. Loh, A.M., et al.: Celestial3D: a novel method for 3D visualization of familial data. Bioinformatics ‘Applications Note’ 24(9), 1210–1211 (2008)
13. Aida, M.: Construction of a Japanese classic genealogy database. IPSJ SIG Computers and the Humanities, 2001-CH-051-6, 39–46 (2001) (in Japanese)
14. Bennett, R.L., et al.: Recommendations for Standardized Human Pedigree Nomenclature. Journal of Genetic Counseling 4(4), 267–279 (1995)
15. PED Pedigree Software, http://www.medgen.de/ped/
16. PAF, http://www.familysearch.org/
17. ScionPC, http://homepages.paradise.net.nz/scionpc/
18. XY Family Tree, http://www.xy-family-tree.com/
19. WeRelate, http://www.werelate.org/wiki/Main_Page/
20. GenoPro, http://www.genopro.com/
21. Naito, M.: Topic Map for Displaying Genealogy. SIG-SWO-A603-04, pp. 4–1–4–7 (2007) (in Japanese)
22. GEDCOM LETTER, http://en.wikipedia.org/wiki/GEDCOM
23. Sugiyama, S., Ikuta, A., Shibata, M., Matsuura, T.: A Study of an Event Oriented Data Management Method for Displaying Genealogy: Widespread Hands to InTErconnect BASic Elements (WHIteBasE). International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM), 280–289 (2011) ISSN: 2150-7988/2
24. Sugiyama, S., Ikuta, A., Yokozawa, D., Shibata, M., Matsuura, T.: Displaying Genealogy with Various Layouts by using the “WHIteBasE” Method. International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM), 102–115 (2014), ISSN: 2150-7988/6
25. Nerva–Antonine dynasty, http://en.wikipedia.org/wiki/Nerva
26. Racehorses, http://db.netkeiba.com/?pid=horse_top (in Japanese)

User Relevance for Item-Based Collaborative Filtering

R. Latha and R. Nadarajan

Department of Applied Mathematics and Computational Sciences,
PSG College of Technology, Coimbatore
Tamil Nadu, India
{lathapsg,nadarajan_psg}@yahoo.co.in

Abstract. A Collaborative filtering (CF), one of the successful recommendation approaches, makes use of history of user preferences in order to make predictions. Common drawback found in most of the approaches available in the literature is that all users are treated equally. i.e., all users have same importance. But in the real scenario, there are users who rate items, which have similar rating pattern. On the other hand, some users provide diversified ratings. We assign relevance scores to users based on their rating pattern in order to improve the quality of predictions. To do so, we incorporate probability based user relevance scores into the similarity calculations. The improvement of predictions of benchmark item based CF approach with the inclusion of user relevance score is demonstrated in the paper.

Keywords: Collaborative filtering, Recommendation System, Information Retrieval, User Relevance.

1 Introduction

With the tremendous growth of Web, volume of data available in net based systems become large and thus results in large repositories. So it becomes too difficult for individuals to handle the data effectively and efficiently. This scenario is commonly referred to as ‘information overload’ problem. Recommender system addresses the problem [6]. Many online business systems such as Amazon.com and Netflix are using recommender systems to provide personalized suggestions.

Two common approaches are available for making recommendations. They are Collaborative Filtering (CF) [3] and Content Based filtering (CB) [8]. Pazzani, et. al [8] defines that content-based algorithms base their recommendations on the contents of items and profiles of users. The profiles allow programs to associate users with matching products. Content Based Filtering uses the assumption that items with similar objective features will be rated similarly and users with similar profile/taste will prefer items in a similar manner. CB approaches are specific to a domain and the scope of the approach is limited to the domain for which they are proposed.

Collaborative Filtering (CF) is the most popular and successful approach for recommendation systems. CF relies only on the past user behavior (ratings, preferences, purchase history, time spent etc) [6]. Breese et. al [3] classifies Collaborative Filtering

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 337–347, 2013.

techniques into two categories: Memory-based and Model based techniques. Memory based algorithms predict the rating of users using the previously rated items by the users and other users who have similar tastes. They operate over the entire user database to make predictions. The most common memory-based models are based on the notion of nearest neighbors, using a variety of distance measures [8]. Model based systems are based on a compact model inferred from the data. We have considered model based CF approach for our work.

In Collaborative Filtering techniques, all users are treated uniformly. But in the real scenario, there are some users who are consistent in providing preferences, where as some users provide ad hoc preferences which do not form any pattern. Ultimately the former kind of users must be assigned higher score, and so their preferences must be given higher importance. We assign scores to users based on their rating pattern in order to improve the quality of predictions. We assign probability based relevance score to users.

The contribution of the proposed work can be divided into two phases, namely offline phase and online phase. In offline phase user relevance scores are calculated and a model is built on relevance scores. In on line phase rating predictions are made based on the model created.

The rest of this paper is organized as follows.

Section 2 presents the overview of item based CF and user relevance problem, Section 3 discusses about the proposed approach to address the problem. Section 4 presents user relevance model building and Section 5 discusses about experimental evaluations of our approach. MovieLens [1] database is used for proving the results. Finally, Section 6 provides some concluding remarks and an outline of the future research.

2 Overview of Item Based CF and User Relevance Problem

In this section we discuss about Item based CF techniques and user relevance problem.

2.1 Item Based CF

To address the scalability concerns of user-based recommendation techniques, item-based recommendation techniques (also known as model-based) have been developed [9]. These techniques compute similarity between items, and then use these similarity values to compute top-N recommendations or make predictions. The reason behind these techniques is that a customer will more likely purchase items that are similar or related to the items that he/she has already purchased. These approaches are faster than user based approaches, since the similarity computations can be done offline which results in faster recommendation engines.

Different kinds of similarity measures are available in the literature for computing relationship between items. In [5], the author has discussed about cosine similarity, adjusted cosine similarity and Pearson correlation coefficient.

In Cosine based approach [4] the similarity between items i and j is calculated as cosine of the angle between them. Only the items which are rated by both the users will be considered for computing the angles. The similarity $Sim_{i,j}$ is defined as

$$Sim_{i,j} = \frac{\sum_{u\in u(i)\cap u(j)} r_{u,i} \times r_{u,j}}{\sqrt{\sum_{u\in u(i)\cap u(j)} \mathrm{r}_{\mathrm{u,i}}{}^{2}}\ \sqrt{\sum_{u\in u(i)\cap u(j)} \mathrm{r}_{\mathrm{u,i}}{}^{2}}} \tag{1}$$

where $u(i)$ represents set of users who have rated for the item i and $r_{u,i}$ represents the ratings given by the user u for the item i. According to B. Sarwar et.al, [5] Cosine similarity does not account for the difference in user ratings. Adjusted Cosine Similarity is suggested to overcome the problem, in which the average rating of the user is subtracted from his actual ratings. The function used to calculate the similarity between item i and item j is given below:

$$Sim_{i,j} = \frac{\sum_{u\in u(i)\cap u(\mathrm{j})} \left(r_{u,i} - \bar{r_u}\right) \times \left(r_{u,j} - \bar{r_u}\right)}{\sqrt{\sum_{u\in u(i)\cap u(j)} (r_{u,i} - \bar{r_u})^2}\ \sqrt{\sum_{u\in u(i)\cap u(j)} (r_{u,j} - \bar{r_u})^2}} \tag{2}$$

where $u(i)$ represents the users who have rated for the item i and $r_{u,i}$ represents the ratings given by user u for item i. $\bar{r_u}$ is the average rating of user u. There are many ways to compute predictions [4]. Weighted sum is a commonly used approach for predicting unknown ratings which is calculated as given in (3)

$$p_{u,i} = \frac{\sum_{j\in \mathrm{S(i)}} Sim_{i,j} * \mathrm{r}_{\mathrm{u,i}}}{\sum_{\mathrm{j}\in \mathrm{S(i)}} \left|\mathrm{Sim}_{\mathrm{i,j}}\right|} \tag{3}$$

Here S(i) is the set of items that are similar to i, the item to be predicted. $p_{u,i}$ is the prediction for item i for the user u. Although CF techniques take different kinds of input, we focus on the input in the form of *m x n* user item matrix.

Definition 1. User-item Matrix R

If there are *m* users who have given ratings for *n* items, then the ratings data can be represented as an *m x n* matrix with rows representing users and columns representing items. The matrix is called user-item matrix *R*. Each element $R_{u,i}$ is an ordinal value which ranges from R_{min} to R_{max}. Unrated values are considered to be zero. A sample rating matrix *R* is shown in Table 1.

2.2 User Relevance in Item Based CF

In Collaborative Filtering techniques we discussed above, the importance of users is not taken into account. There are some users who prefer items with similar rating pattern. i.e, if a user rates an item and most of its similar items, then the user is consistent in rating. By assigning higher weight to them brings accuracy in recommendations. On the other hand there are some users who rate items which do not have any similar rating pattern. They rate dissimilar items. Such users are divergent in thinking and assigning higher weight to them improves diversity in recommendations. But current recommender systems treat all users uniformly.

Min Gao et. al [11] proposed a technique to assign rank to users based on popularity and use the rank values in similarity calculations. The technique improves accuracy of predictions.

Our approach proposed in this paper is similar to the work described in [11], but instead of user rank, user relevance score is calculated. Moreover the proposed technique improves accuracy as well as diversity of recommendations.

Despite of much work available in the literature to improve quality of predictions, up to our knowledge no other work is available in the literature to assign weights to users and treat them differently.

Table 1. Original rating Matrix R

	I1	I2	I3	I4	I5	I6
U1	1	2	0	5	0	0
U2	0	3	0	0	5	0
U3	0	0	5	0	0	2
U4	0	1	0	0	0	0

3 Probability Based Relevance Score for Users

This section describes the procedure for building a model for assigning relevance score for the users. To calculate the relevance score for users we apply a probability based technique. The model building takes two phases namely item-item similarity calculation phase and relevance score computation phase.

3.1 Item-item Similarity Calculation Phase

In order to define pair-wise similarity between item vectors many similarity measures namely Pearson correlation coefficient, cosine similarity, adjusted cosine similarity etc are available in the literature. The similarity measure we have considered is based on number of users who agree on item vectors *i* and *j*. The Correlation Rating Matrix (*CRM*) stores the similarities between each pair of item vectors. $CRM_{i,j}$ between item vectors *i* and *j* is defined as given in (4)

$$CRM_{i,j} = \begin{cases} 1, & \text{if} \quad |ui \cap uj| > 0, \quad i,j = 1,2,3,\dots,n \\ 0\,, & Otherwise \end{cases} \tag{4}$$

where ui is the set of users who have rated for item i. $|u_i \cap u_j|$ represents cardinality of the set $(u_i \cap u_j)$.

Definition 2. Correlation Rating Matrix (*CRM*)

For the given $m \times n$ user item rating matrix R, item-item Correlation Rating Matrix can be represented as an $n \times n$ matrix. The matrix rows and columns represent items and each $CRM_{i,j}$ represents 1 if at least one user rates both the items. $CRM_{i,j}= 0$, implies that item i is not related to item j.

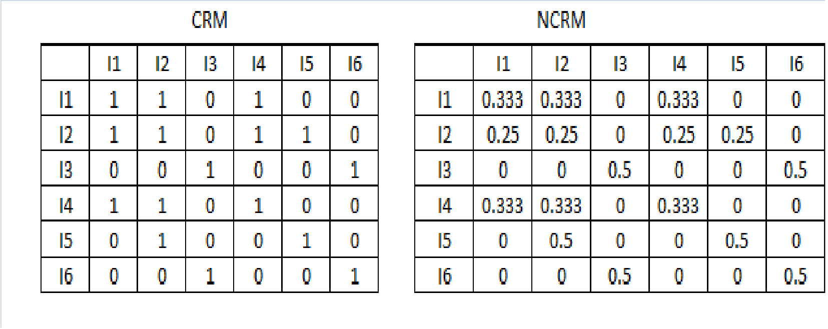
CRM is a symmetric matrix. But if an item i is rated by few users and item j is rated by many users, then their correlation should differ. So we normalize *CRM* in to Normalized Correlation Similarity Matrix *NCRM*.

Definition 3. Normalized Correlation Rating Matrix (*NCRM*)

NCRM is a matrix that records the relationship between items as ratio of how the user is correlating in the current item and the remaining items. NCRM can be calculated based on the formula given in (5)

$$NCRM_{\mathrm{i,j}} = \frac{CRM_{i,j}}{\sum_i CRM_{i,j}} \tag{5}$$

The rows and columns of *NCRM* represent items and it is an asymmetrical matrix. For each item, the *NCRM* value is between 0 and 1. Since the relevance score computation procedure we employ requires binary similarity matrix for relevance score computations. So we convert *NCRM* into S, a binary similarity matrix of various items of the rating matrix. *CRM* and *NCRM* of original rating matrix *R* are shown in Fig. 1.



CRM

	I1	I2	I3	I4	I5	I6
I1	1	1	0	1	0	0
I2	1	1	0	1	1	0
I3	0	0	1	0	0	1
I4	1	1	0	1	0	0
I5	0	1	0	0	1	0
I6	0	0	1	0	0	1

NCRM

	I1	I2	I3	I4	I5	I6
I1	0.333	0.333	0	0.333	0	0
I2	0.25	0.25	0	0.25	0.25	0
I3	0	0	0.5	0	0	0.5
I4	0.333	0.333	0	0.333	0	0
I5	0	0.5	0	0	0.5	0
I6	0	0	0.5	0	0	0.5

Fig. 1. CRM and NCRM of original matrix R given in Table1

Definition 4. Item-item Similarity Matrix (S)

For the given $m \times n$ user item rating matrix R, item-item similarity matrix is a binary similarity matrix(S) which can be calculated by the formula

$$\mathrm{S_{i,j}} = \begin{cases} 1, & \mathrm{NCRM_{i,j}} > 0.5, \quad i,j = 1,2,3,\dots.,n \\ 0, & \text{Otherwise} \end{cases} \tag{6}$$

S is a binary similarity matrix. Similarity matrix S of R is shown in Table 2.

Table 2. Binary Similarity Matrix, S

	I1	I2	I3	I4	I5	I6
I1	0	0	0	0	0	0
I2	0	0	0	0	0	0
I3	0	0	1	0	0	1
I4	0	0	0	0	0	0
I5	0	1	0	0	1	0
I6	0	0	1	0	0	1

3.2 Relevance Score Computation Phase

From the binary similarity matrix *S*, for each item, the items with similarity value 1 are called relevance items (*R*) and with similarity value 0 are called non-relevant items (*NR*). Next we investigate how user *u* has rated an item *i* and all its similar and dissimilar items. If he has rated most of the similar items of *i*, then he can be assigned a good score for his consistency in rating. At the same time high consistence reveals monotonic rating behavior of users which needs to be balanced.

On the other hand if the user has rated an item and most of the dissimilar items of it, then he can be assigned a good score for his divergent thinking and rating. So we consider both scores in order to have a tradeoff between them. We combine both of them to calculate relevance score of the user. The relevance score of the user is estimated using (7) and (8). This is based on the basic probability model proposed by Robertson and Spark [15] in Information Retrieval domain. This feature weighing scheme is used by search engines to rank matching documents according to their relevance to a given search query.

$$CW_{u,i} = \log\frac{(r_i + 0.5) * (N - R - n_i + r_i + 0.5)}{(n_i - r_i + 0.5) * (R - r_i + 0.5)} \tag{7}$$

$$DW_{u,i} = \log\frac{(nr_i + 0.5) * (N - NR - n_i + nr_i + 0.5)}{(n_i - nr_i + 0.5) * (NR - nr_i + 0.5)} \tag{8}$$

where $CW_{u,i}$ and $DW_{u,i}$ are the consistency and diversity scores of user u for item i. $CW_{u,i}$ and $DW_{u,i}$ are log odd ratios of user u in relevant and non-relevant items of item i respectively. N is the number of items in the collection. R is the number of items similar to item *i*. *NR* is the total number of dissimilar items of *i*. n_i is the number of items in the collection for which the user *u* has rated and r_i is the number of similar items of *i* for which user u has rated, $\mathrm{n}r_i$ is the number of dissimilar items of *i* for which user *u* has rated. Weight assigned to each user is calculated as the total scores assigned to all items and is calculated as given in (9) and (10).

$$CW_u = \sum\nolimits_{i \in I} CW_{u,i} \tag{9}$$

$$DW_u = \sum\nolimits_{i \in I} DW_{u,i} \tag{10}$$

where I is the set of items and CW_u and DW_u are called consistency and diversity scores of user u. Consistency score signifies user's consistency in rating items where as diversity signifies user's diversified thinking in analyzing items. Since both the scores are important as one improves accuracy of predictions where as other improves diversity in predictions, we consider average of the two scores as a common score for users, which we call as relevance score. Relevance score of users is calculated as given in (11).

$$RW_u = \frac{CW_u + DW_u}{2} \tag{11}$$

4 Building User Relevance Model for CF

In the previous section we discussed about computing user relevance score. We incorporate user relevance score into the most widely used similarity measure namely Cosine Similarity. A new similarity measure which we call User Relevance based Cosine Similarity, $RelSim_{i,j}$ is formulated as given in (12).

$$RelSim_{i,j} = \frac{\sum_{u \in u(i) \cap u(j)} (r_{u,i}) \times (r_{u,j}) * \mathrm{RW_u}^2}{\sqrt{\sum_{u \in u(i) \cap u(j)} \mathrm{r_{u,i}}^2 * \mathrm{RW_u}^2} \sqrt{\sum_{u \in u(i) \cap u(j)} \mathrm{r_{u,i}}^2 * \mathrm{RW_u}^2}} \tag{12}$$

RWu is the relevance score of the user who rated both the items i and j. After calculating the similarity values, a model is built as described in [5].

5 Experimental Evaluation

This section discusses about the data sets used, accuracy of predictions and computational complexities. Experiments are conducted using Item-based collaborative filtering recommendation algorithms [5] and Userrank for item-based Collaborative filtering recommendation [11], in order to prove the efficiency of the proposed technique.

5.1 Data Set Used

The experiments are run on two datasets namely Movielens[1] and a subset of rating selected from it. The subset is populated by taking top 100 users based on total number of rating and top 100 items rated by those users. We call it as MovieTop100. The datasets have ratings given for movies in the range 1 to 5.

The details of the data sets are given in Table 3. We partition the data sets into training set with 80% of the ratings and test data set with 20% of the ratings. Five cross validation is done for all experiments.

Table 3. Data sets used

	Movielens	MovieTop100
# users	943	100
# items	1682	100
# ratings	100,000	2474

5.2 Experimental Metrics and Evaluation Methodologies

Here we outline the experiments we have taken to study the significance of the proposed technique in improving the quality of predictions in terms of accuracy and diversity.

Accuracy Measures

In order to evaluate the performance of the proposed technique in terms of accuracy of predictions, we follow the approach used in [14]. The metrics used are HR (Hit Ratio), MAE (Mean Absolute Error) and RMSE (Root Mean Square Error). HR is the ratio of the number of hits to the size of test data set used. The predicted rating is a called as a hit if its rounded value is equal to the actual rating given in the test data set. MAE is average absolute deviation between predicted and actual ratings. MAE penalizes each wrong prediction by its distance to the actual rating, whereas RMSE emphasizes larger deviations. These measures are as formulated below.

$$MAE = \frac{\sum_{i=1}^{n} |\, p_i - a_i|}{n} \qquad RMSE = \sqrt{\frac{\sum_{i=1}^{n}(p_i - a_i)^2}{n}} \qquad HR = \frac{no\ of\ hits}{n}$$

Where p_i is the predicted rating and a_i is the actual rating for the item i. n is the number of test cases.

Diversity Measure (Cosine Diversity)

In order to show that the proposed approach improves diversity of predictions we employ average inter list dissimilarity of predicted values. The cosine coefficient is a commonly used similarity measure. Based on cosine similarity a new diversity measure called cosine dissimilarity is proposed in [16]. Cosine dissimilarity is calculated for each pair of predicted ratings and its average is taken as diversity value.

We compared our technique with ItemCF [5] and ItemCF(U-Rank) [11]. Table 4 shows the results of computations for various measures. The quality of predictions increases as HR and Diversity increase whereas quality decreases as MAE and RMSE increase. The values of various measures from Table 4 show that the proposed approach performs better than the benchmark techniques ItemCF and ItemCF(U_Rank) on both the datasets.

Table 4. Comparison of various measures by ItemCF, ItemCF(U_Rank) and ItemCF(U_Rel)

	Movielens			MovieTop100		
	ItemCF	ItemCF (U_Rank)	ItemCF (U_Rel)	ItemCF	ItemCF (U-Rank)	ItemCF (U_Rel)
MAE	0.9318	0.9306	0.9155	0.9208	0.9055	0.9020
RMSE	1.2041	1.1976	1.1778	1.1710	1.1575	1.1519
HR	0.3386	0.3388	0.3441	0.3233	0.3278	0.3288
DIV	0.0243	0.0244	0.0250	0.0152	0.0175	0.0177

Fig. 2 and Fig. 3 show the comparison of all the three algorithms using different measures on Movielens dataset.

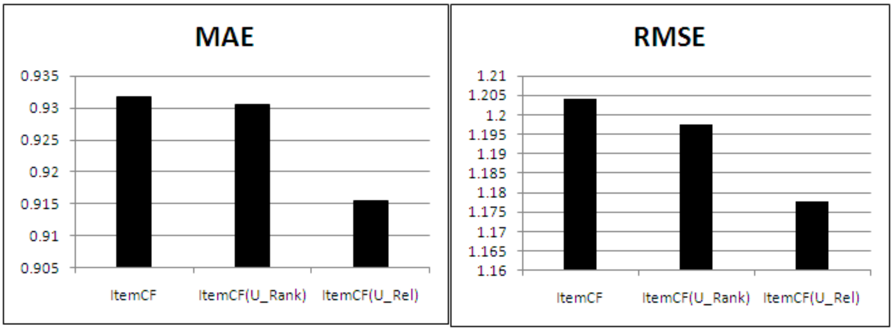


Fig. 2. Comparison of MAE and RMSE measures for Movielens dataset

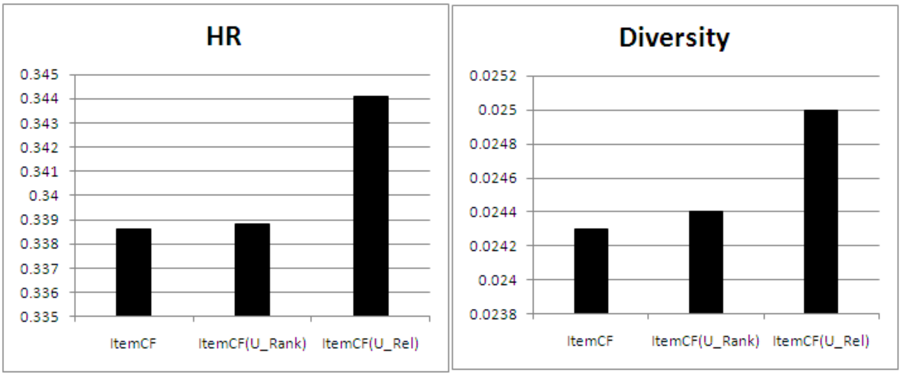


Fig. 3. Comparison of HR and Diversity measures for Movielens dataset

Fig. 4 and Fig. 5 show the comparison of all the three algorithms using different measures on MovieTop100 dataset.

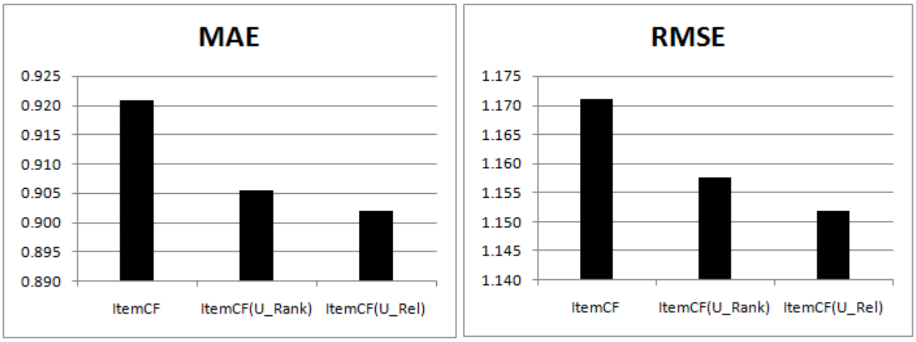


Fig. 4. Comparison of MAE and RMSE measures for MovieTop100 dataset

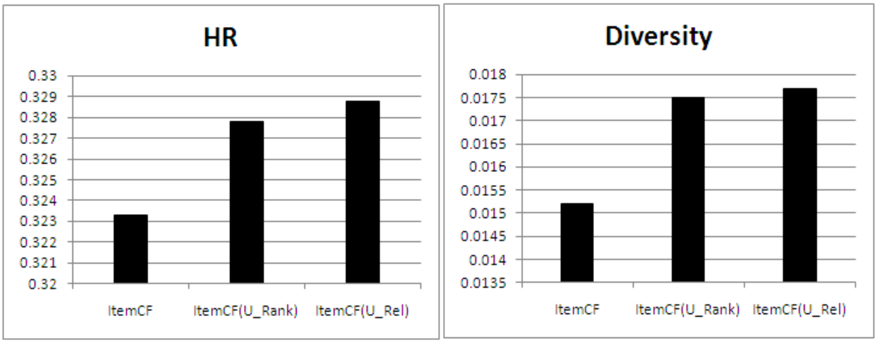


Fig. 5. Comparison of HR and Diversity measures of MovieTop100 dataset

5.3 Comparing Computational Complexities

High computational complexity is often needed to enhance the predictions. In the model based view, the computational complexity can be split into complexity in offline phase and complexity in online phase. Offline phase includes similarity computations, relevance score calculations and model building. The complexity of offline computations is $O(mn^2+kn^2)$ for *m* users, *n* items and top *k* neighbours. Online phase includes prediction computations of n test items with complexity *O(kn).* For most recommender systems, the online complexity is more important than offline complexity.

6 Conclusion

The goal of this work is to improve accuracy and diversity of predictions by including relevance score of users. The proposed approach sets user relevance score as a combination of user's consistency and diversity scores and a new similarity measure is proposed based on user relevance score. A bench mark algorithm with the inclusion of user relevance score is examined for proving the efficiency of results.

The experimental results show that the inclusion of relevance weight helps in improving predictions. In future we plan to include the content based attribute values in order to calculate the relevance score of the users.

References

1. MovieLens data, `http://www.grouplens.org/`
2. Ricci, F., Rokach, L., Shapira, B., Kantor, P.B.: Recommender Systems Handbook. Springer (2011)
3. Breese, J.S., Heckerman, D., Kadie, C.: Empirical Analysis of Predictive Algorithms for Collaborative Filtering, UAI (1998)
4. Adomavicius, G., Tuzhilin, A.: Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Transactions on Knowledge and Data Engineering (June 2005)
5. Sarwar, B., Karypis, G., Konstan, J., Reidl, J.: Item-based collaborative filtering recommendation algorithms. World Wide Web, 285-295 (2001)
6. Melville, P., Sindhwani, V.: Recommender Systems, Encyclopedia of Machine Learning (2010)
7. Ekstrand, M.D., Reiedl, J.T., Konstan, J.A.: Collaborative filtering Recommendation systems, Foundations@Trends. Human-Computer Interaction (2010)
8. Pazzani, M.J., Billsus, D.: Content-based recommendation systems. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 325–341. Springer, Heidelberg (2007)
9. Karypis, G.: Evaluation of Item-Based Top-N Recommendation Algorithms. In: CIKM, Atlanta, Georgia, USA (2001)
10. Su, X., Khoshoftaar, T.M.: A Survey of Collaborative Filtering Techniques. Hindawi Publishing Corporation, Advances in Artificial Intelligence (2009)
11. Gao, M., Wu, Z., Jiang, F.: Userrank for item-based Collaborative filtering recommendation. Information Processing Letters (2011)
12. Lemire, D., Maclaclan, A.: Slope One Predictors for Online Rating-Based Collaborative Filtering. In: Society for Industrial and Applied Mathematics International Conference on Data Mining, SDM (2005)
13. Schafer, J.B., Konstan, J., Riedl, J.: Recommender systems in E-Commerce. In: ACM E-Commerce Conference (1999)
14. Yildirim, H., Krishnamoorthy, M.S.: A Random Walk Method for Alleviating the Sparsity Problem in Collaborative Filtering. In: RecSys 2008 (2008)
15. Robertson, S.E., Sparck Jones, K.: Relevance Weighting of Search Terms. Journal of American Society of Information Science (1976)
16. Hurley, N., Zhang, M.: Novelty and diversity in top-*n* recommendation – Analysis and evaluation. ACM Trans. Internet Technology (March 2011)

Extraction of Agent Groups with Similar Behaviour Based on Agent Profiles

Kateřina Slaninová[1], Jan Martinovič[2], Roman Šperka[1], and Pavla Dráždilová[2]

[1] School of Business Administration in Karviná,
Silesian University in Opava,
Univerzitní nám. 1934/3a, 733 40, Karviná, Czech Republic
{slaninova,sperka}@opf.slu.cz

[2] Faculty of Electrical Engineering,
VŠB Technical University in Ostrava,
17. listopadu 15/2172, 708 33 Ostrava, Czech Republic
{jan.martinovic,pavla.drazdilova}@vsb.cz

Abstract. This paper deals with the log files suitable to extract valuable information about agents and their behaviour from agent-based simulation in a model of virtual company. Such information, presented in a transparent way, can be used as a support for simulation verification to achieve the suitable design of the proposed system. Hence, based on the similar behaviour (represented by extracted sequences) of agents, we are able to construct models which explain certain aspects of agent behaviour. Moreover, we can extract agent profiles based on behaviour and find latent ties between different agent groups with similar behaviours. The paper extends the results of our previous works about sequence extraction and comparison. The approach for agent network construction based on agent profiles is described. Two different methods were used for construction of agent network. One method uses cosine similarity and graph partitioning and the second self organization maps and Euclidean similarity for agent relations. Each of these methods has its advantages and disadvantages which are summarized in the paper and presented in the form of the visualization of relations between agents.

Keywords: Agent Profile, Log Analysis, Agent Behaviour.

Introduction

Modern applications like information systems, enterprise systems or e-commerce systems, as well as monitoring applications, web applications, and other systems produce a large amount of data collections. These data collections are usually kept in databases, data warehouses, or simply in data or log files.

A standard log file typically consists of records with information about recorded events that have occurred in the system. These records may contain various attributes such as information about the date and time when the event happened, an originator, the type of event and additional information. The originator can be a person, a device, or software. This depends on the type of log file. In the case of a person, we can extract the relevant records and obtain information about his behaviour.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 348–357, 2013.

This paper deals with the log files suitable to extract valuable information about agents and their behaviour from agent-based simulation in a model of virtual company [1]. The motivation to use agents as simulation subjects outcomes from the agents characteristics e.g. autonomy, coordination, communication etc. precisely declared in [2,3]. This ensures to substitute real business processes participants (sellers and customers) and simulate their trading communication. Such information, presented in a transparent way, can be used as support for simulation verication to achieve the suitable design of the proposed system.

This paper is focused on log files where one log file attribute is an originator of the recorded activity, and the originator is an agent. Hence, based on the similar attributes of agents, we are able to construct models which explain certain aspects of an agent behaviour [4]. Moreover, we can extract agent profiles based on behaviour and find latent ties between different agent groups with similar behaviours.

1 Log Analysis

Log analysis is a data mining process focused on the analysis of computer-generated records (also called audit trail records, event logs or transaction logs) [5]. Log file analysis has gained growing attention in all areas of human activity. This domain is very interesting not only for researchers, but for commercial software developers as well. There are various disciplines considered in the analysing of data sources with the intention of achieving worthy information, often represented as knowledge. Obtained information (or knowledge) is then used for management, system maintenance, and system optimisation, marketing campaigns, recommender systems or other purposes such as discovering a company's structure or the structure of social networks.

A log file is usually a simple text file generated by a device, software, application or system. It consists of messages, which are represented by records of events performed in the system. An event log consists of cases, whilst cases consist of events. It typically contains activity information in events, but mostly timestamp, originator (performer) and other necessary data as well.

In a case, the events are represented in the form of a *trace*, i.e. a sequence of unique events [4,6]. If an event log contains timestamps (a time when the event was performed in the system) then the ordering in a trace respects this attribute $\#_{time}(e)$.

2 Pattern Mining

Unsupervised pattern mining is used for pattern recognition tasks, where training feature vectors with known class labels are not available. In this type of problem, there is used a set of feature vectors **x**, and the main goal is to achieve clusters (groups) of vectors on the basis of their similarity (or proximity measure). This process is generally known as clustering.

In the experiments, selected pattern mining methods were used to find patterns of similar behaviour of agents and to find groups of agents with similar behaviour. This section is only short introduction of the selected methods, used in the case study.

2.1 Self Organizing Maps

Self organizing maps (SOM), also called Kohonen maps, is a type of artificial neural network invented by professor Teuvo Kohonen in 1982 [7]. The input space of the training samples is represented in a low dimensional (often two-dimensional) space, called map. The model is capable of projecting a high-dimensional space to a lower-dimensional space and is efficient in structure visualization due to its feature of topological preservation using a neighbourhood function. Obtained low-dimensional map is often used for pattern detection, clustering, or for characterization and analysis of the input space [8]. The Euclidean distance is used in the SOM algorithm.

2.2 Spectral Clustering

Spectral clustering is one of the divisive clustering algorithms which can be applied in the graph theory. The spectral clustering algorithm uses eigenvalues and eigenvectors of a similarity matrix derived from the data set to find the clusters. In this section, there is described the type of spectral clustering based on the second smallest eigenvector of the Laplacian matrix.

How to use the spectral algorithm is studied in [9] by Cheng et al. In [10] Ding et al. proposed a new graph partition method based on the min-max clustering principle: the similarity between two subgraphs (cut set) is minimized, while the similarity within each subgraph (summation of similarity between all pairs of nodes within a subgraph) is maximized.

Spectral clustering method optimized by our proposed Left-Right Oscillate algorithm [11] was used in two parts of the proposed approach: for finding the behavioural patterns and for better visualization of agent network based on their similar behaviour in the system.

3 Business Processed and Agent-Based Simulation

The main motivation for the research in the field of business process simulation was to investigate agent' behavioural patterns and to find groups of agents with similar behaviour during the agent-based simulation in a model of virtual company. Used multi-agent system (MAS) operates with randomly (resp. pseudo randomly) generated parameters, and is able to deal with unpredictable phenomena surrounding the company. To achieve the suitable design of the proposed system, a mechanism for verification is needed.

Agent-based simulations dealing with a company simulation can bring several crucial advantages [12,13]. It is possible to involve unpredictable disturbance of the environment into the simulation with the agents. The analysis and visualization of agent behaviour lead to facilitate the verification of the simulation model and to effectively observe the reaction of the model in relation to the initialization of its input attributes, or in relation to the influence of the model environment.

3.1 Simulation Model

The agent-based simulation model from which were analysed data outputs is described in [14], with the focus on Business Process Management (BPM). Usual business process simulation approaches are based on the statistical calculation, as can be seen for example in [15]. However, only several problems can be identified while using the statistical methods. The model uses the advantages of the agent technology, to describe some of the core business processes of a typical business company. The authors in [16] developed a business process simulation framework in order to simulate the real behaviour of the company on the market.

The core of the experiments was the analysis of the agent-based simulation outputs. To ensure the outputs, above mentioned framework was used to trigger simulation experiments. The simulation framework covered business processes supporting the selling of goods by company sales representatives to the customers.

Each activity performed by agents in the simulation model was recorded into a log file. The log was a simple text file, where each row represented one event. The main row structure, which was valid for all agents, consisted of following event attributes: $\#_{time}(e)$, denoted as `TimeStamp` (date and time when the event was performed), $\#_{resource}(e)$, denoted as `AgentID` (each agent had its unique ID), $\#_{activity}(e)$, denoted as `TypeOfAction`, and other additional attributes like `AgentClassName` or `ActionAttribute`. Example of the log file is presented in Example 1.

Example 1 (Example of Log File - Simulation Model).

```
2012/03/26 01:14:14 : SA0022 bpm.selling.SellerAgent SENT_PROPOSAL CA0223,12,5.0006104,73
2012/03/26 01:14:14 : SA0008 bpm.selling.SellerAgent SENT_PROPOSAL CA0088,8,5.0097656,31
...
2012/03/26 01:14:15 : CA0183 bpm.selling.CustomerAgent RECEIVED_CPF_MESSAGE SA0018,8,5.004883,53
2012/03/26 01:14:15 : CA0170 bpm.selling.CustomerAgent SENT_REPLY SA0017,false,1.0462701894732744,5.0006104,29
2012/03/26 01:14:15 : CA0170 bpm.selling.CustomerAgent FINISHED_TURN
```

In used multi-agent model, there were monitored five types of agents: SellerAgent, CustomerAgent, ManagerAgent, InformativeAgent and DisturbanceAgent.

4 Proposed Approach with Case Study in Business Process Simulation

The proposed approach proceeds from the original social network approach with a modification focused on agent behaviour. The modification is based on a definition of the relation between agents. The original approach to the analysis of social networks deals with the assumption that the social network is a set of people (or groups of people) with social interactions among themselves [17]. Social interaction is commonly defined as interaction between the actors like communication, personal knowledge of each other, friendship, membership, etc...

The modification extends the original approach of social network analysis by the perspective of the complex networks. This type of view differs from the original approach in the description of relations between the nodes (agents). The relation between the agents is defined by their common attributes characterizing their behaviour in the system. More specifically, the agent behaviour in the system is defined by agent profiles.

Creation of the agent profiles requires a set of agents U and a set of event sequences S performed in the system by the agents. Extraction of these sets is described in our article [18].

Definition 1. (Base agent profile, sequences)

Let $U = \{u_1, u_2, \ldots, u_n\}$, be a set of agents, where $i = 1, 2, \ldots, n$ is a number of agents u. Then, sequences of events $\sigma_{ij} = \left\langle e_{ij1}, e_{ij2}, \ldots e_{ijm_j} \right\rangle$, are sequences of events executed by the agent u_i in the system, where $j = 1, 2, \ldots, p_i$ is a number of that sequences, and m_j is a length of j-th sequence. Thus, a set $S_i = \left\{\sigma_{i1}, \sigma_{i2}, \ldots \sigma_{ip_i}\right\}$ is a set of all sequences executed by the agent u_i in the system, and p_i is a number of that sequences.

Sequences σ_{ij} extracted with relation to certain agent u_i are mapped to set of sequences $\sigma_l \in S$ without this relation to agents: $\sigma_{ij} = \left\langle e_{ij1}, e_{ij2}, \ldots, e_{ijm_j} \right\rangle \rightarrow \sigma_l = \langle e_1, e_2, \ldots, e_{ml} \rangle$, where $e_{ij1} = e_1, e_{ij2} = e_2, \ldots, e_{ijm_j} = e_{ml}$.

Define matrix $B \in N^{|U| \times |S|}$ where

$$B_{ij} = \begin{cases} frequency\ of\ sequence\ \sigma_j \in S\ for\ agent\ u_i & \text{if } \sigma_j \ \in S_i \\ \hfill 0 & \text{else} \end{cases}$$

A base agent profile of the agent $u_i \in U$ is a vector $b_i \in N^{|S|}$ represented by row i from matrix B.

Example 2 shows an example of base agent profile generated for agent behaviour during a business process simulation in the multi-agent system.

Example 2 (Example of Base Agent Profiles).

```
#agent:sa0028
0|1|ability_generated;|0;
1|253|sent_proposal;sent_proposal;sent_proposal;sent_proposal; ...
3|1|sent_proposal;sold;|6;11;
```

Similarity between two profiles can be quantified by several methods. The selection of the method depends on the type of data used [19]. For Boolean data, Hamming distance, Jaccard dissimilarity or Rogers-Tanimoto dissimilarity may be used. In the case of numerical data, Euclidean distance, Manhattan distance, cosine similarity or correlation distance may be used. If we use string data, then Edit distance, Levenshtein distance or Hamming distance are all potential options. In this paper, the similarity between agent profiles was determined by using Euclidean similarity [20] and cosine similarity.

4.1 Network of Agent Groups with Similar Behaviour

The network of agent groups with similar behaviour was created using graph theory approach. In this phase of the presented approach, it was constructed a weighted and undirected graph $G(V, E)$, where vertices (graph nodes) V were agents u_i and relations (edges) E expressed the similarity between the agent profiles.

The output graph can be obtained by comparison of Base agent profiles. The edge weights were counted by cosine similarity in this paper. As the output, we obtained a

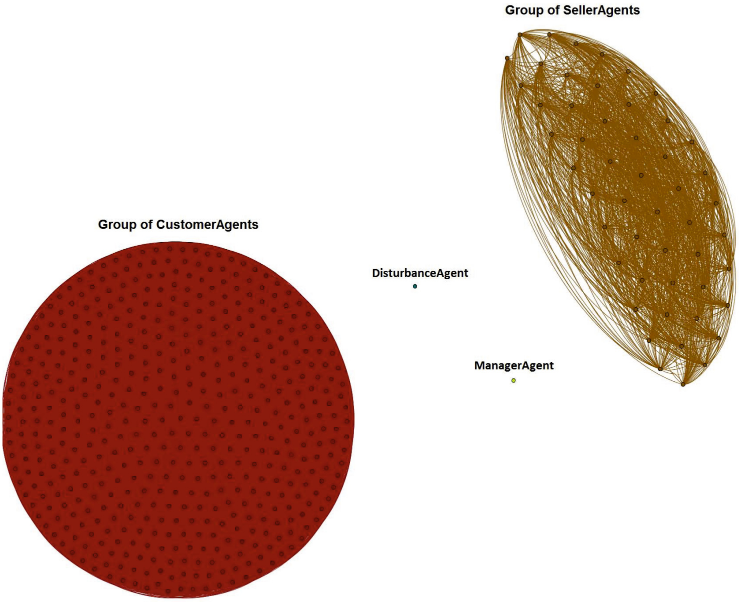


Fig. 1. Example of Agent Network (Spectral Clustering and LRO)

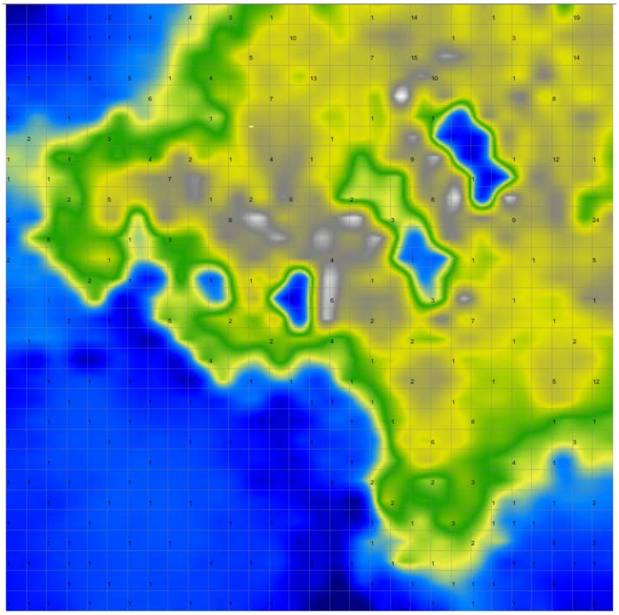


Fig. 2. Example of Clusters with Agents Visualized by SOM

graph of agents (a *.gdf* file). Another output was a file with set of agents U and their vectors consisted of sequences (behavioural patterns) and their frequency.

In this last phase of the proposed approach, two methods for construction of network of groups with agents were compared. The first method proceeded from graph of agents, mentioned before. The groups of agents with similar behaviour (similar agent profiles) were found using spectral clustering by Fiedler vector using Left-Right Oscillate algorithm. The graph was constructed only from edges, which weight was higher than a selected threshold. The graph contained the isolated agents (nodes) as well, because they are as important as the other connected nodes. They created a separate component in the graph, see example graph of groups of agents with similar behaviour in multi-agent system in Figure 1.

The second method used for the construction of agent network with groups of agents with similar behaviour was SOM. This method used a file with agents and their vectors consisted of sequences (behavioural patterns) and their frequency as the input for SOM.

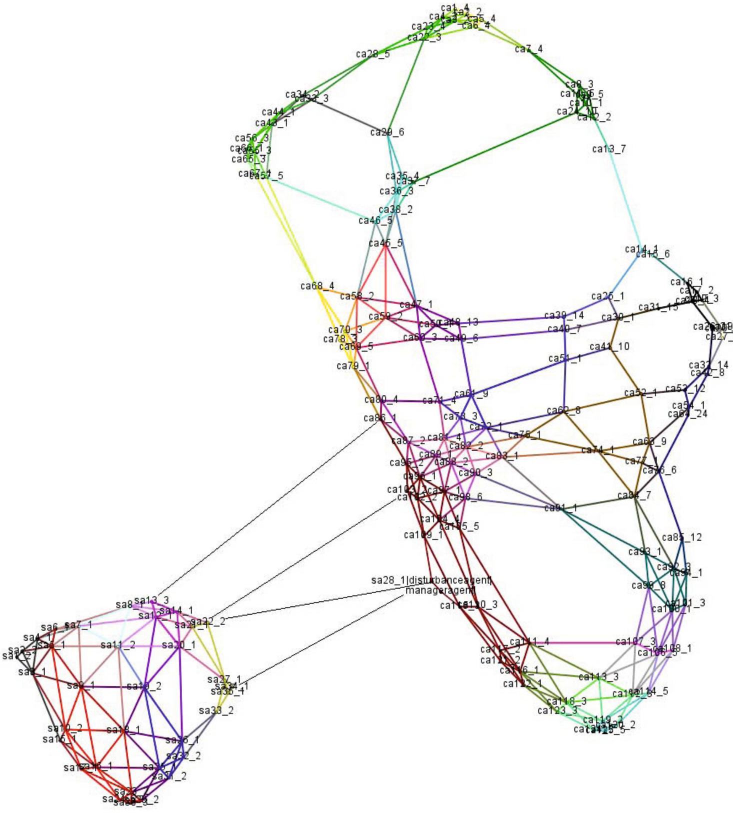


Fig. 3. Example of Agent Network (SOM)

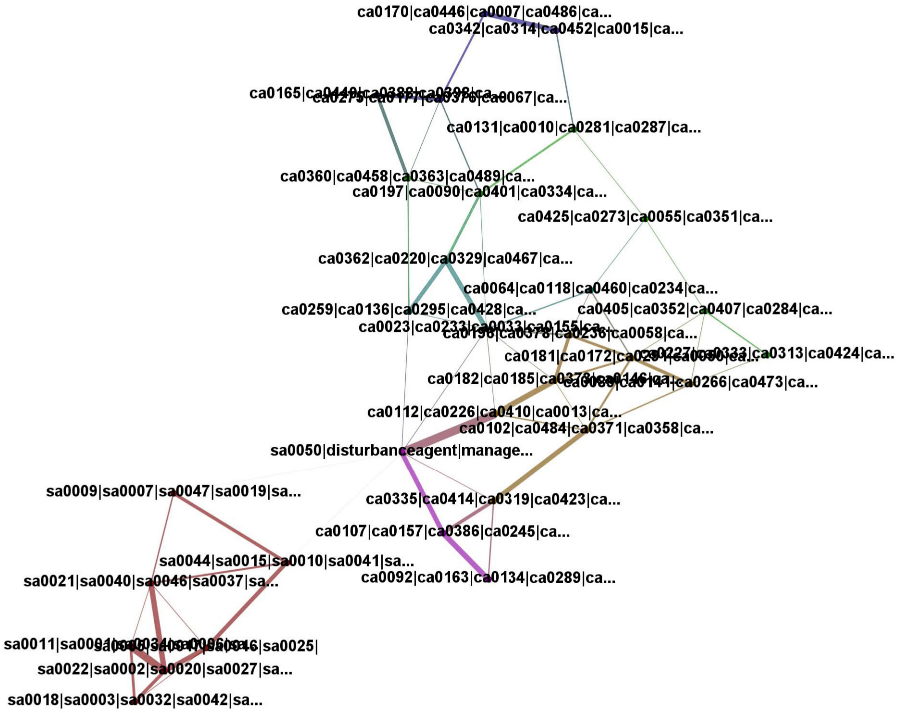

ca0170|ca0446|ca0007|ca0486|ca...
ca0342|ca0314|ca0452|ca0015|ca...
ca0165|ca0449|ca0388|ca0398|ca...
ca0275|ca0177|ca0376|ca0067|ca...
ca0131|ca0010|ca0281|ca0287|ca...
ca0360|ca0458|ca0363|ca0489|ca...
ca0197|ca0090|ca0401|ca0334|ca...
ca0425|ca0273|ca0055|ca0351|ca...
ca0362|ca0220|ca0329|ca0467|ca...
ca0064|ca0118|ca0460|ca0234|ca...
ca0259|ca0136|ca0295|ca0428|ca...
ca0405|ca0352|ca0407|ca0284|ca...
ca0023|ca0233|ca0033|ca0155|ca...
ca0198|ca0378|ca0236|ca0058|ca...
ca0181|ca0172|ca0294|ca0080|ca...
ca0227|ca0333|ca0313|ca0424|ca...
ca0182|ca0185|ca0378|ca0146|ca...
ca0086|ca0141|ca0266|ca0473|ca...
ca0112|ca0226|ca0410|ca0013|ca...
ca0102|ca0484|ca0371|ca0358|ca...
sa0050|disturbanceagent|manage...
sa0009|sa0007|sa0047|sa0019|sa...
ca0335|ca0414|ca0319|ca0423|ca...
ca0107|ca0157|ca0386|ca0245|ca...
sa0044|sa0015|sa0010|sa0041|sa...
sa0021|sa0040|sa0046|sa0037|sa...
ca0092|ca0163|ca0134|ca0289|ca...
sa0011|sa0091|sa0034|sa0096|sa...
sa0048|sa0047|sa0046|sa0025|
sa0022|sa0002|sa0020|sa0027|sa...
sa0018|sa0003|sa0032|sa0042|sa...

Fig. 4. Example of Reduced Agent Network (SOM)

The goal of this method was to obtain such output SOM map, where each neuron of the output map consisted of agents with similar behaviour (with similar agent profiles). However, the output SOM map has its predefined structure (in our case lattice), which is not suitable for visualisation of agent network. In Figure 2, we can see example of smoothed data histogram used for cluster visualization in SOM map by SOMToolbox [21]. The example shows the visualization of agents with similar behaviour, where white colour represents neurons with the higher amount of agents and blue colour represents empty (almost empty) neurons.

Therefore, we have used further steps to transform SOM output lattice to graph of groups with agents. This transformation was performed by removing the empty neurons (neurons to which none of agents were assigned). The rest of nodes from output SOM lattice were considered as graph nodes with groups of agents. Due to information about the distance between the neurons, we obtained the weights for relations between the nodes (Euclidean similarity). It is common that some neurons may be very close, which means that the agents in such neurons may be very similar, and may create one group in the agent network. Therefore, a further clustering and colouring of the obtained clusters in a graph of agents was performed using spectral clustering method by Fiedler vector and Left-Right Oscillate algorithm. The different colours represent clusters of agents with similar behaviour.

Moreover, the graph of agents can be reduced into a graph of agent clusters, where clusters contain agents with similar profiles. This approach is suitable for better visualization in cases with large amounts of agents, see example in Figure 4.

5 Conclusion and Future Work

In the paper, the approach which uses agent profiles for finding of agents groups with similar behaviour have been introduced. This approach may use different methods for groups construction. As founded during presented experiments, each method has its advantages and disadvantages. In the last section, there was presented cosine similarity and graph partitioning method as one possible way for this purpose. Another presented way was based on SOM and graph extraction from SOM output which uses Euclidean similarity.

The results showed, that if we use cosine similarity, the agent without any identical sequences will be in different clusters (groups of agents). The Figure 2 presented results of the tested simulation where two big and compact clusters were found. The problem of this method was, that it was not able to detect tiny changes in the frequency of sequences in agent profiles. Therefore, the agent clustering based on their similar behaviour was too rough.

To remove this disadvantage, SOM method was used, and the SOM output map was transformed to the graph. Such approach resolved the problem with detection of tiny changes between agents behaviour. However, this representation had a problem that the relation between totally different profiles could be constructed. This was caused by the reduction to lower dimensions represented by SOM map.

The both disadvantages described above could be resolved by combination of the obtained results. We intent to work on splitting the agents without common sequences in the graph obtained from SOM map. For this purpose, we will use clusters obtained by cosine similarity and graph partitioning.

Acknowledgments. This work was partially supported by the grant of Silesian University in Opava, Czech Republic, No. SGS/06/2013 'Advanced Modeling and Simulation of Economic Systems' and by SGS, VSB – Technical University of Ostrava, Czech Republic, under the grant No. SP2013/167 'Analysis of the behaviour patterns in complex networks'.

References

1. Barnett, M.W.: Modeling & simulation in business process management. Gensym Corporation (2003)
2. Wooldridge, M.: An Introduction to MultiAgent Systems, 2nd edn. Wiley Publishing (2009)
3. Macal, C.M., North, M.J.: Tutorial on agent-based modeling and simulation. In: Proceedings of the 37th Conference on Winter Simulation, WSC 2005, pp. 2–15. Winter Simulation Conference (2005)
4. van der Aalst, W.M.P., Reijers, H.A., Song, M.: Discovering social networks from event logs. Comput. Supported Coop. Work 14(6), 549–593 (2005)

5. van der Aalst, W.M.P., van Dongen, B.F., Herbst, J., Maruster, L., Schimm, G., Weijters, A.J.M.M.: Workflow mining: a survey of issues and approaches. Data Knowl. Eng. 47(2), 237–267 (2003)
6. van der Aalst, W.M.P.: Process Mining: Discovery, Conformance and Enhancement of Business Processes, 1st edn. Springer, Heidelberg (2011)
7. Kohonen, T.: Self-Organization and Associative Memory, 3rd edn. Springer Series in Information Sciences, vol. 8. Springer, Heidelberg (1984)
8. Vojáček, L., Martinovič, J., Slaninová, K., Dráždilová, P., Dvorský, J.: Combined method for effective clustering based on parallel som and spectral clustering. In: Snášel, V., Pokorný, J., Richta, K. (eds.) DATESO 2011, VŠB - TU Ostrava, pp. 120–131 (2011)
9. Kannan, R., Vempala, S., Vetta, A.: On clusterings: Good, bad and spectral. J. ACM 51(3), 497–515 (2004)
10. Ding, C.H.Q., He, X., Zha, H., Gu, M., Simon, H.D.: A min-max cut algorithm for graph partitioning and data clustering. In: ICDM 2001: Proceedings of the 2001 IEEE International Conference on Data Mining, pp. 107–114. IEEE Computer Society, Washington, DC (2001)
11. Dráždilová, P., Martinovič, J., Slaninová, K.: Spectral clustering: Left-right-oscillate algorithm for detecting communities. In: ADBIS Workshops, pp. 285–294 (2012)
12. De Snoo, D.: Modeling planning processes with talmod. Master's thesis, University of Groningen (2005)
13. Jennings, N., Faratin, P., Norman, T., O'Brien, P., Odgers, B.: Autonomous agents for business process management. Int. Journal of Applied Artificial Intelligence 14, 145–189 (2000)
14. Macal, C., North, J.: Tutorial on agent-based modeling and simulation. In: Proceedings: 2005 Winter Simulation Conference (2005)
15. Scheer, A.-W., Nüttgens, M.: ARIS architecture and reference models for business process management. In: van der Aalst, W.M.P., Desel, J., Oberweis, A. (eds.) Business Process Management. LNCS, vol. 1806, pp. 376–389. Springer, Heidelberg (2000)
16. Spišák, M., Šperka, R.: Financial market simulation based on intelligent agents - case study. Journal of Applied Economic Sciences VI(17), 249–256 (2011)
17. Newman, M.E.J.: Networks: An Introduction. Oxford University Press (2010)
18. Slaninová, K., Martinovič, J., Dráždilová, P., Vymětal, D., Šperka, R.: Analysis of agents' behavior in multiagent system. In: 24th European Modeling and Simulation Symposium, EMSS 2012, pp. 169–175 (2012)
19. Deza, M.M., Deza, E.: Dictionary of Distances. Elsevier Science, Amsterdam (2006)
20. Elmore, K.L., Richman, M.B.: Euclidean distance as a similarity metric for principal component analysis. Monthly Weather Review 129, 540 (2001)
21. Pampalk, E., Rauber, A., Merkl, D.: Using smoothed data histograms for cluster visualization in self-organizing maps. In: Dorronsoro, J.R. (ed.) ICANN 2002. LNCS, vol. 2415, pp. 871–876. Springer, Heidelberg (2002)

Aesthetic Patterns from the Perturbed Orbits of Discrete Dynamical Systems

Krzysztof Gdawiec

Institute of Computer Science, University of Silesia, Poland
kgdawiec@ux2.math.us.edu.pl

Abstract. The aim of this paper is to present some modifications of the orbits generation algorithm of discrete dynamical systems. The first modification is based on introduction of a perturbation mapping in the standard Picard iteration used in the orbit generation algorithm. The perturbation mapping is used to alter the orbit during the iteration process. The second modification combines the standard Picard iteration with the iteration which uses the perturbation mapping. The obtained patterns have unrepeatable structure and aesthetic value. They can be used for instance as textile patterns, ceramics patters or can be used in jewellery design.

Keywords: dynamical system, orbit, perturbation mapping, aesthetic pattern.

1 Introduction

One of the most elusive goals in computer aided design is artistic design and pattern generation. Pattern generation involves diverse aspects: analysis, creativity, development [15]. A designer have to deal with all of these aspects in order to obtain an interesting pattern which later could be used in jewellery design, carpet design, as a texture etc. Usually the most work during the design stage is carried out by a designer manually. Especially, in the case in which a graphic design should contain some unique unrepeatable artistic features. Therefore, it is highly useful to develop an automatic method for aesthetic pattern generation.

Aesthetics in the world of art and photography is connected with the principles of the nature and the perception of beauty [1]. Judging the beauty and other aesthetic qualities of patterns, paintings, photographs is a highly subjective task, so there is no standard method of measuring aesthetic values and it is a challenge to create such method in the emerging discipline of computational aesthetic [16]. However, in most of the works about pattern generation, in this paper also, the aesthetic is judged by the subjective feeling of the authors.

The literature is full of very diverse methods of pattern generation. For instance in [10] [14] the methods based on Iterated Function Systems and Genetic Algorithms for jewellery design were proposed. An interesting method based on root-finding polynomials, called polynomiography, was presented in [4]. In [13] a product-delay algorithm was proposed which is based on multiplication of sine

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 358–366, 2013.

and square waves and in [5] authors presented a method which creates stone-like decorations using marbling.

Also very diverse methods based on discrete dynamical systems were proposed. In [8] authors made use of Gumowski-Mira transform for the generation of patterns used later as textile patterns. In their method they took the Gumowski-Mira transform and changed only the parameters of it. Other approach was presented in [2]. The authors have used different kinds of dynamical systems and using the switching process between the dynamical systems in conjunction with the Krasnoselskij iteration they generated aesthetic patterns. In this paper we propose another approach in obtaining aesthetic patterns from discrete dynamical systems. We modify the standard Picard iteration using the perturbation mapping and also combine both methods.

The paper is organized as follows. In Sect. 2 we introduce the basic information about discrete dynamical systems and present examples of dynamical systems which produce nicely looking orbits. Next, in Sect. 3 we introduce the notion of a perturbation mapping and how it can be used in the pattern generation. Some examples of aesthetic patterns obtained with the proposed methods are presented in Sect. 4. Finally, in Sect. 5 we give some concluding remarks.

2 Discrete Dynamical Systems

Let M be a subset in the q-dimensional Euclidean space $\mathbb{R}^q$. Following [9] discrete dynamical system is a continuous mapping $\Phi : M \times \mathbb{N} \to M$ such that

$$\Phi(x, 0) = x, \tag{1}$$

$$\Phi(\Phi(x, t), s) = \Phi(x, t + s) \tag{2}$$

for all $t, s \in \mathbb{N}$ and $x \in M$. The variable t is thought of as the time and generally discrete dynamical systems result from iterative processes or difference equations [9].

Assume that $f : M \to M$ is a continuous mapping. Then f generates a discrete dynamical system of the form [9]:

$$\Phi(x, n) = f^n(x) = \underbrace{f \circ \ldots \circ f}_{n \text{ times}}(x). \tag{3}$$

If $n = 0$ then $f^0(x) = x$.

In the rest of the paper we will be interested in discrete dynamical systems given in a following form:

$$x_n = f^n(x) = f(f^{n-1}(x)) = f(x_{n-1}), \tag{4}$$

where $n > 0$. Formula (4) is also called the Picard iteration.

For our further considerations we also need the notion of an orbit. Orbit (or trajectory) of a point x_0 is a sequence $\{x_n\}_{n=0}^{\infty}$, where x_n is given by (4).

Many examples of dynamical systems are known [7], but we are mainly interested in those which produce geometric patterns that can be recognized as aesthetic ones. Now, we present the examples of such dynamical systems in $\mathbb{R}^2$:

- Hopalong transformation [6]

$$x_n = y_{n-1} - sgn(x)\sqrt{|bx_n - c|},\\ y_n = a - x_{n-1}, \tag{5}$$

where $a, b, c \in \mathbb{R}$ and $sgn : \mathbb{R} \to \mathbb{R}$ is defined as follows:

$$sgn(x) = \begin{cases} -1 & \text{if } x < 0, \\ 0 & \text{if } x = 0, \\ 1 & \text{if } x > 0, \end{cases} \tag{6}$$

- Zaslavsky transformation [7]

$$x_n = (x_{n-1} + K\sin y_{n-1})\cos\alpha + y_{n-1}\sin\alpha,\\ y_n = -(x_{n-1} + K\sin y_{n-1})\sin\alpha + y_{n-1}\cos\alpha, \tag{7}$$

where $K \in \mathbb{R}$, $\alpha = \frac{2\pi}{q}$, $q \in \mathbb{N}$, $q \geq 3$,
- Chip transformation created by Peters for the HOP program [11]

$$x_n = y_{n-1} - sgn(x_{n-1})\cos(\ln|bx_{n-1} - c|)^2 \cdot \arctan(\ln|cx_{n-1} - b|)^2,\\ y_n = a - x_{n-1}, \tag{8}$$

where $a, b, c \in \mathbb{R}$,
- Quadrup Two transformation also created by Peters for the HOP program [11]

$$x_n = y_{n-1} - sgn(x_{n-1})\sin(\ln|bx_{n-1} - c|)\arctan(cx_{n-1} - b)^2,\\ y_n = a - x_{n-1}, \tag{9}$$

where $a, b, c \in \mathbb{R}$,
- Three Ply transformation is another transformation created by Peters [11]

$$x_n = y_{n-1} - sgn(x_{n-1})|\sin x_{n-1}\cos b + c - x_{n-1}\sin(a + b + c))|,\\ y_n = a - x_{n-1}, \tag{10}$$

where $a, b, c \in \mathbb{R}$.

The other examples of dynamical systems, producing interesting orbits, are: Gumowski-Mira transformation (CERN, 1980) [3], Martin [6], Cockatoo [6] etc. In Fig.1 the examples of orbits for the first 100000 points for transformations (5), (7), (8), (9), (10) are presented. The parameters for the transformations were following: Hopalong – $a = 55$, $b = -10$, $c = -42.1$, $x_0 = 0$, $y_0 = 0$, Zaslavsky – $K = 4$, $q = 7$, $x_0 = 1$, $y_0 = 1$, Chip – $a = 34$, $b = 1$, $c = 5$, $x_0 = 0$, $y_0 = 0$, Quadrup Two – $a = 34$, $b = -1.2$, $c = -20$, $x_0 = 0$, $y_0 = 0$, Three Ply – $a = 55$, $b = -10$, $c = -42.1$, $x_0 = 0$, $y_0 = 0$.

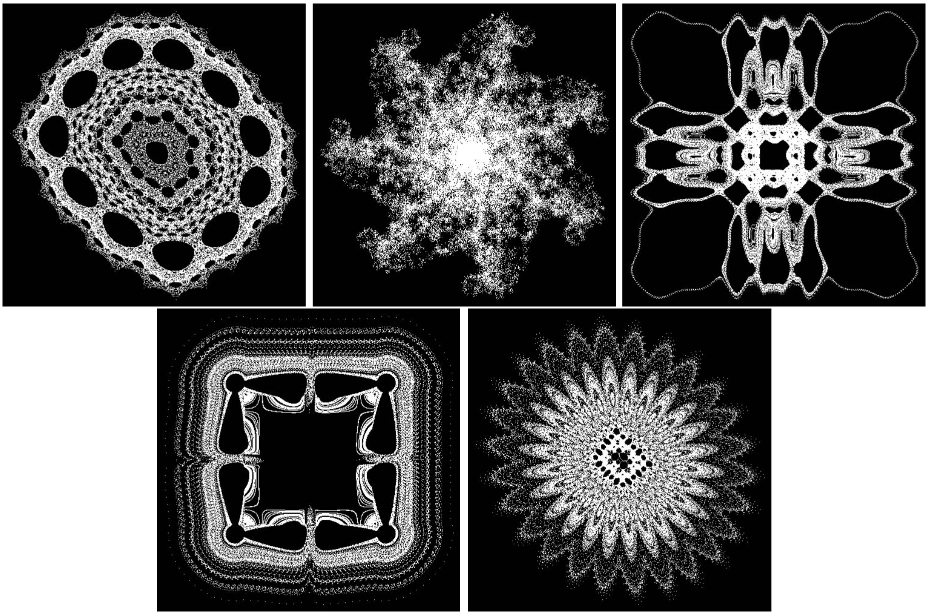


Fig. 1. The examples of orbits ($n = 100000$), the top row (from the left): Hopalong, Zaslavsky, Chip, the bottom row (from the left): Quadrup Two, Three Ply

3 Orbit Perturbation

As we mention at the beginning of the paper there exist methods which use discrete dynamical systems presented in Sect. 2. They use changing of the transformation parameters or switching process in conjunction with the Krasnoselskij iteration. In our approach we are interested in the orbits of discrete dynamical systems.

Lets see an example of orbits for different starting points of a fixed dynamical system. Figure 2 presents orbits starting from (from left): $(1.2, -2.05)$, $(-3.55, -1.2)$ and $(3.601, -1.05)$ for the Quadrup Two transformation with $a = 34$, $b = 1$, $c = 5$ and $n = 100000$. To colour the orbits we used the iteration method [2] in which the colour of a point is given according to the number of iteration at which this point arises and a fixed colourmap. In this example we see that for different starting points we obtain various shapes of the orbits. This gives an idea that we can alter the orbits during the iteration process to change the shape of the final orbit.

We modify the Picard iteration given by (4) in a following way:

$$x_n = (f \circ p)^n(x) = (f \circ p)((f \circ p)^{n-1}(x)) = (f \circ p)(x_{n-1}), \tag{11}$$

where $p : M \to M$ is a mapping. The mapping p is called a perturbation mapping and its aim is to alter (perturb) the orbit during the iteration process. We do

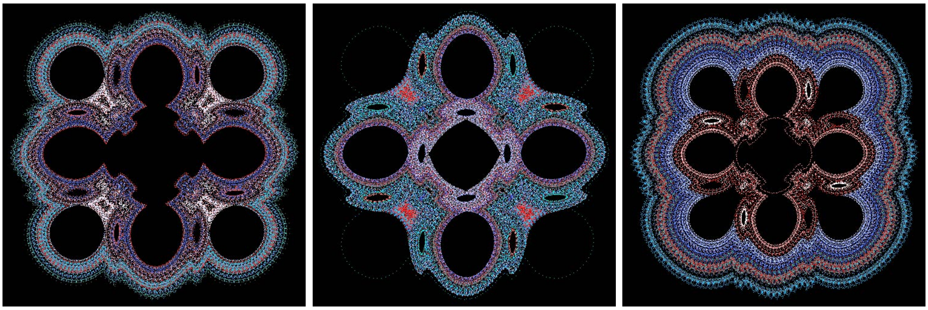


Fig. 2. Orbits of Quadrup Two transformation for different starting points

not make any assumptions about the perturbation mapping because we can alter the orbit in very different ways. When $p(x) = x$ for all $x \in M$ we do not have any perturbation of the orbit and therefore (11) reduces to (4).

A very simple example of perturbation mapping is:

$$p_v(x) = x + v, \tag{12}$$

where $v \in M$. This mapping simply translates the given point by v. In this way we change the orbit in every iteration in the same direction. We also can change the orbit by v only in some iterations, e.g. if the number of iteration is divisible by some fixed value.

Of course there is nothing in the way to use different values of v in consecutive iterations. In this case we can define some equation for v or we can define a grid with vectors and the appropriate vector is then computed using the bilinear interpolation. We also can give some randomness in the perturbation mapping taking in each iteration a random value of v.

When we have a perturbation mapping p we can define a new iteration process which combines the standard Picard iteration with the iteration given by (11). For instance this can be done in a following way:

$$x_n = \alpha x'_n + (1 - \alpha)x''_n, \tag{13}$$

where $\alpha \in \mathbb{R}$ and

$$x'_n = f(x_{n-1}), \tag{14}$$

$$x''_n = f(p(x_{n-1})). \tag{15}$$

4 Examples

In this section we show some examples obtained using the methods presented in Sect. 3. In all the examples we use the same colouring method as in Fig. 2 but with other colourmaps.

We start with an example of Zaslavsky transformation with $K = 3$, $q = 5$, starting point $(1, 1)$ and number of iterations equal 100000. In the iteration process we used p_v given by (12) with $v = (1, 0.5)$ which is used after every: 500, 2500, 10000 iterations. The obtained results are presented in Fig. 3. The top part of the figure presents the original orbit without any perturbation, and the bottom part presents the patterns obtained with the perturbed orbits after every (from left): 500, 2500, 10000 iterations.

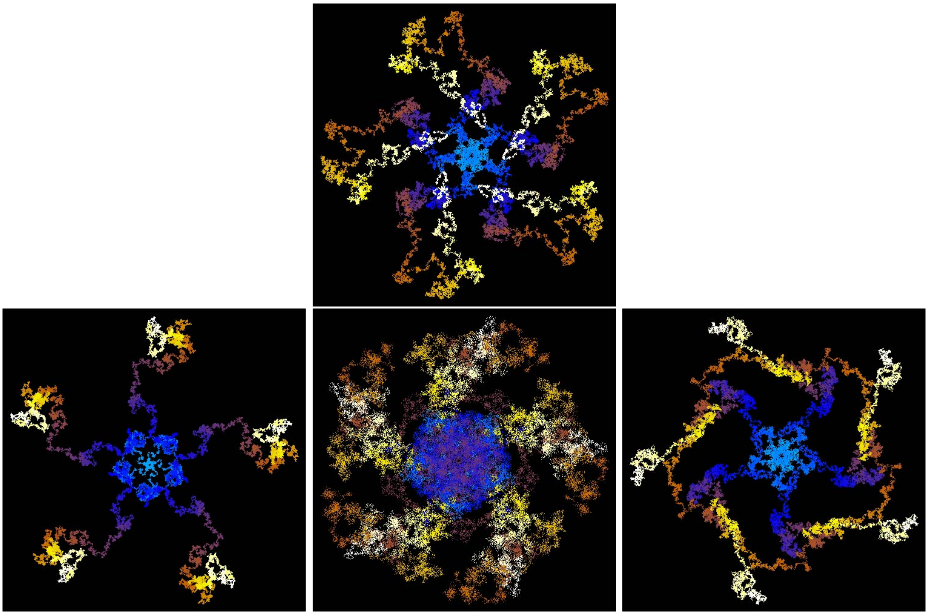


Fig. 3. Original orbit of Zaslavsky transformation (top) and patterns obtained with perturbation after a fixed number of iterations (bottom)

The next example presents the use of perturbation mapping p_v given by (12) with different values of v. This time we used the Chip transformation with $a = 55$, $b = -10$, $c = -42$, starting point $(1, 1)$ and number of iterations equal 100000. Figure 4 presents the obtained patterns. The top part presents the original orbit, and the bottom part presents patterns for different values of v (from left): $(0.1, 0.2)$, $(9.1, -8.3)$, $(-0.58, 2.16)$.

Using the random vectors in the perturbation mapping we are also able to obtain very interesting and diverse patterns which shows next example. For this example we used the Hopalong transformation with $a = -55$, $b = -1$, $c = -42$, starting point $(0, 0)$ and number of iterations equal 100000. The obtained patterns are presented in Fig. 5, where the original orbit is in the top part, and in the bottom part we have patterns obtained with the random vectors. The co-ordinates of the random vector were randomly chosen from $[-0.05, 0.05]$.

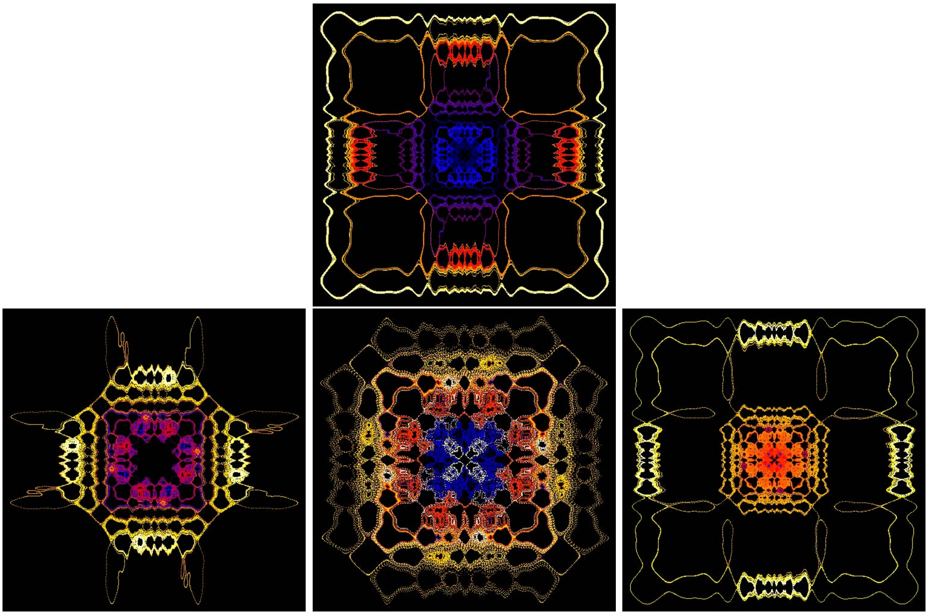


Fig. 4. Original orbit of Chip transformation (top) and patterns obtained with perturbation using different vectors (bottom)

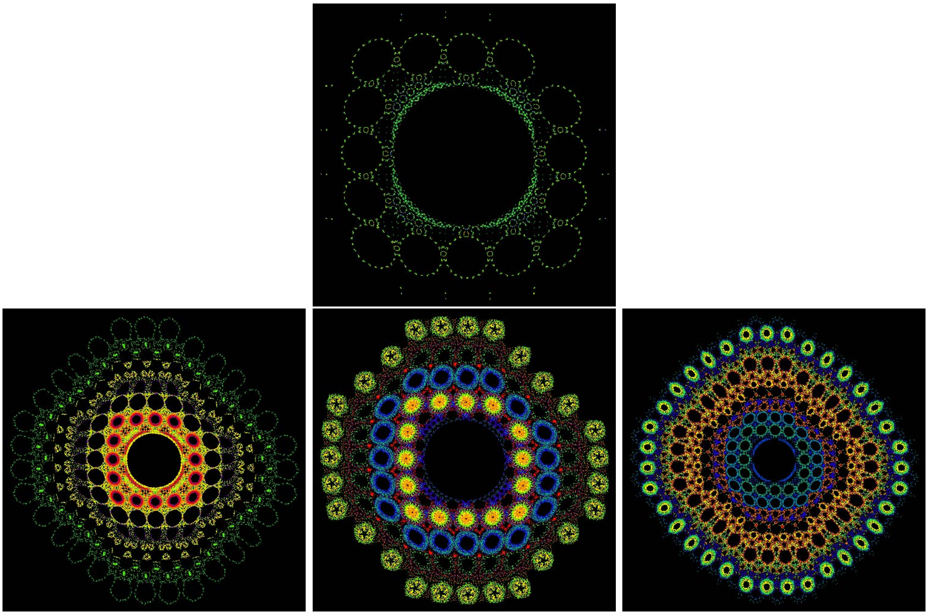


Fig. 5. Original orbit of Hopalong transformation (top) and patterns obtained with perturbation using random vectors (bottom)

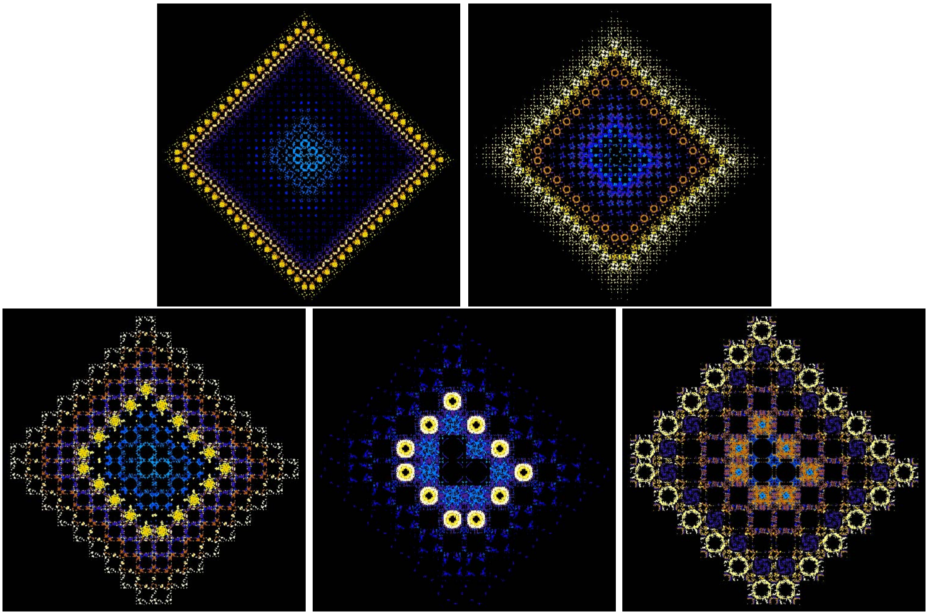


Fig. 6. Original orbit of Three Ply transformation (top-left), pattern obtained with perturbation mapping with a fixed vector (top-right) and patterns obtained using the combination (13) with different values of α (bottom)

In the last example we present some patterns obtained using the combination of the Picard iteration and the perturbed iteration. We used the Three Ply transformation with $a = -54.4$, $b = -1$, $c = -42$, starting point $(0, 0)$ and number of iterations equal 100000. The obtained patterns are presented in Fig. 6. The original orbit is in top-left part of the figure, and in the top-right we have a pattern obtained with perturbation with a fixed vector $v = (-0.5, 0.5)$. In the bottom part we have the patterns obtained with the combination method with different values of α (from left): 0.3, 0.6, 0.9.

5 Conclusions

In the paper we presented new methods for patterns generation with the use of discrete dynamical systems. The first method uses the perturbation mapping, which alters the orbit of a dynamical system during the iteration process. We also presented some perturbation mappings. The second method uses a combination of the two methods: the standard Picard iteration and the method with the perturbation mapping.

The presented examples show that using the proposed methods we are able to obtain very interesting and diverse patterns. The patterns differ from those obtained with the standard Picard iteration. Comparing to the method from [8] the proposed methods give more possibilities of obtaining unrepeatable patterns.

The obtained patterns have an aesthetic value so they can be used as usable patterns, e.g. textile patterns, ceramics patterns, or can be used in jewellery, decoration design or in generating textures using for instance the glyph bombing technique [12].

References

1. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Studying Aesthetics in Photographic Images Using a Computational Approach. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. LNCS, vol. 3953, pp. 288–301. Springer, Heidelberg (2006)
2. Gdawiec, K., Kotarski, W., Lisowska, A.: Automatic Generation of Aesthetic Patterns with the Use of Dynamical Systems. In: Bebis, G., et al. (eds.) ISVC 2011, Part II. LNCS, vol. 6939, pp. 691–700. Springer, Heidelberg (2011)
3. Gumowski, I., Mira, C.: Recurrences and Discrete Dynamic Systems. Springer, New York (1980)
4. Kalantari, B.: Polynomial Root-Finding and Polynomiography. World Scientific, Singapore (2009)
5. Lu, S., Jaffer, A., Jin, X., Zhao, H., Mao, X.: Mathematical Marbling. IEEE Computer Graphics and Applications 32(6), 26–35 (2012)
6. Martin, B.: Graphic Potential of Recursive Functions. In: Landsdwon, J., Earnshaw, R.A. (eds.) Computers in Art, Design and Animation, pp. 109–129. Springer, Heidelberg (1989)
7. Morozov, A.D., Dragunov, T.N., Boykova, S.A., Malysheva, O.V.: Invariant Sets for Windows. World Scientific, Singapore (1999)
8. Naud, M., Richard, P., Chapeau-Blondeau, F., Ferrier, J.L.: Automatic Generation of Aesthetic Images for Computer-assisted Virtual Fashion Design. In: Proceedings 10th Generative Art Conference, Milan, Italy (2007)
9. Osipenko, G.: Dynamical Systems, Graphs, and Algorithms. Springer, New York (2007)
10. Pang, W., Hui, K.C.: Interactive Evolutionary 3D Fractal Modeling. Visual Computer 26(12), 1467–1483 (2010)
11. Peters, M.: HOP – Fractals in Motion, `http://www.mpeters.de/mpeweb/hop/`
12. Rost, R.J., Licea-Kane, B.: OpenGL Shading Language, 3rd edn. Addison-Wesley, Boston (2010)
13. Sen, A.K.: A Product-Delay Algorithm for Graphic Design. Computers & Graphics 22(6), 759–764 (1998)
14. Wannarumon, S., Bohez, E.L.J.: A New Aesthetic Evolutionary Approach for Jewelry Design. Computer-Aided Design & Applications 3(1-4), 385–394 (2006)
15. Wannarumon, S., Unnanon, K., Bohez, E.L.J.: Intelligent Computer System for Jewelry Design Support. Computer-Aided Design & Applications 1(1-4), 551–558 (2004)
16. Wu, Y., Bauckhage, C., Thurau, C.: The Good, the Bad, and the Ugly: Predicting Aesthetic Image Labels. In: Proceedings 20th International Conference on Pattern Recognition, Istanbul, Turkey, pp. 1586–1589 (2010)

Weighted Approach to Projective Clustering

Przemysław Spurek[1], Jacek Tabor[1], and Krzysztof Misztal[2]

[1] Jagiellonian University
Faculty of Mathematics and Computer Science
Łojasiewicza 6, 30-348 Kraków, Poland
{przemyslaw.spurek,jacek.tabor}@ii.uj.edu.pl
[2] AGH University of Science and Technology
Faculty of Physics and Applied Computer Science
al. A. Mickiewicza 30, 30-059 Kraków, Poland
Krzysztof.Misztal@fis.agh.edu.pl

Abstract. k-means is the basic method applied in many data clustering problems. As is known, its natural modification can be applied to projection clustering by changing the cost function from the squared-distance from the point to the squared distance from the affine subspace. However, to apply thus approach we need the beforehand knowledge of the dimension.

In this paper we show how to modify this approach to allow greater flexibility by using the weights over respective range of subspaces.

Keywords: Projective clustering, Karhunen-Loéve Transform, PCA, k-means.

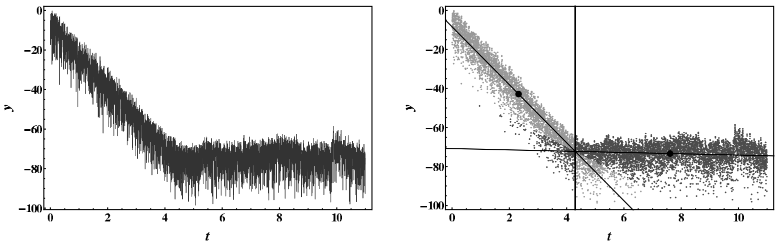
1 Introduction

Projective clustering is a part of the large subspace clustering family, which general aim lies in dividing the given high-dimensional data-set into clusters. For the survey, motivation and further references on subspace clustering we refer to [1,8,11,12].

A typical application of the projective clustering concerns splitting of a given data-set S into clusters with respect to k affine subspaces. A typical illustrative motivation to create such an algorithm is given by a problem to determine linear components of the data obtained in acoustical experiments [6] (see Fig. 1(a)). In general, this kind of data contains two linear components: the first connected with sound (product in experiments) which linearly disappears being absorbed by the walls and the air, and the second consists of noise connected with empty space. To use statistical analysis, we have to extract both of them.

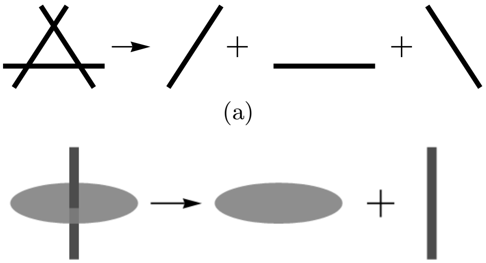
The simplest solution, see [4], lies in a natural modification of k-means: instead of finding k centers which best represent the data, we find k subspaces of given dimension. In other words we change the cost function from the squared-distance from the center of the cluster to the squared distance from the affine space which best represents it and can be found by PCA [5]. To explain it graphically let us consider the following example. Fig. 2(a) represents three lines in the

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 367–378, 2013.



(a) (b)

Fig. 1. Linear component in acoustical data structure. Fig. 1(a) – original data. Fig. 1(b) – outcome from (ω, k)-means algorithm for $k = 2$, $\omega = (0, 1)$. We extract two linear components in data (black dots match clusters centers with the corresponding lines describing those clusters, vertical line separate sound and background noise – after 4.3 s).



(a)

(b)

Fig. 2. Our goal is to create algorithm which will split Fig. 2(b) as two groups of one- and two-dimensional points

plane. The method given in [4] will split the data into three lines. However, it will not work sufficiently well on data given in 2(b), which we would like to split into a line and a circle.

In this paper we generalize the approach from [4] to allow the weights over subspaces of different dimensions, which gives a partial solution to the above mentioned problem. The resulted algorithm we call (ω, k)-means, where ω represents the weights, and k the number of clusters we are interested in. In analogy to the case of k-means, we obtain a version of the Voronoi diagram (see next section). In the simplest form our algorithm needs the number of clusters k and the weight parameter ω (for $\omega = (1, 0, \ldots, 0)$ we obtain the k-means while for $k = 1$ we obtain the PCA).

Detailed investigation of the results of (ω, k)-means allows to determine the optimal dimensions of subspaces, see algorithm in Section 4. The results of this approach we apply to image compression (Section 4, Example 3). In such a case we obtain almost twice the compression of the classical approach.

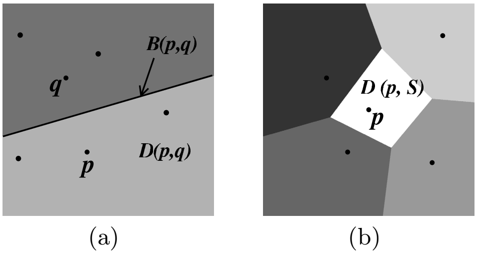


Fig. 3. Graphical presentation of $B(p,q)$, $D(p,q)$ and $D(p,S)$ in $\mathbb{R}^2$

2 Generalized Voronoi Diagram

The Voronoi diagram is one of the most useful data structures in computational geometry, with applications in many areas of science [10]. For the convenience of the reader and to establish the notation we shortly describe the classical version of the Voronoi diagram (for more details see [7]). For $N \in \mathbb{N}$ consider $\mathbb{R}^N$ with the standard Euclidean distance and let S be a finite subset of $\mathbb{R}^N$. For $p, q \in S$ such, that $p \neq q$, let

$$B(p,q) = \{x \in \mathbb{R}^N \colon \|p-x\| = \|q-x\|\}, \tag{1}$$

$$D(p,q) = \{x \in \mathbb{R}^N \colon \|p-x\| < \|q-x\|\}. \tag{2}$$

The hyperplane $B(p,q)$ divides $\mathbb{R}^N$ into two subsets, one containing points which are closer to point p then q ($D(p,q)$), and the second one containing points which are closer to point q then p ($D(q,p)$) – see Figure 3(a).

The set

$$D(p,S) := \bigcap_{q \in S \colon \, q \neq p} D(p,q)$$

of all points that are closer to p than to any other element of S is called the (open) *Voronoi region* [7] of p with respect to S. Set $D(p,S)$ for $N = 2$ is the interior of a convex, possibly unbounded polygon (Figure 3(b)). The points on the contour of $D(p,S)$ are those that have more than one nearest neighbour in S, one of which is p. The union

$$V(S) := \bigcup \partial D(p,S)$$

of all region boundaries is called the Voronoi diagram [7] of S. The common boundary of two Voronoi regions is a *Voronoi edge*. Two edges meet at a Voronoi vertex.

Now we proceed to the description of our modification of the *Voronoi diagram*. We divide the space $\mathbb{R}^N$ with respect to affine subspaces of $\mathbb{R}^N$. For $n \leq N$ let

$$\mathrm{E}_n(\mathbb{R}^N) := \{(v_0, \ldots, v_n) \in (\mathbb{R}^N)^{n+1} \, : \, v_i, v_j \text{ are orthonormal for } i, j > 0, i \neq j\}.$$

Thus v_0 denotes a center of affine space we consider, while $v_1, \dots, v_n$ is the orthonormal base of its "vector part". From the geometrical point of view the element $\mathrm{v} = (v_0, v_1, \dots, v_n) \in \mathrm{E}_n(\mathbb{R}^N)$ represents the affine space

$$v_0 + \mathrm{lin}(v_1, \dots, v_n) = \mathrm{aff}(v_0, v_1, \dots, v_n).$$

We modify equations (1) and (2), by using distance between a point and affine subspace.

Definition 1. *Let $n < N$ and let $\mathrm{v} \in \mathrm{E}_n(\mathbb{R}^N)$, $\omega = (\omega_0, \dots, \omega_n) \in [0,1]^{n+1}$ such that $\sum\limits_{j=0}^{n} \omega_j = 1$ be given. For $x \in \mathbb{R}^N$ let*

$$\mathrm{DIST}_\omega(x; \mathrm{v}) := \left(\sum_{j=0}^{n} \omega_j \ \mathrm{dist}(x; \mathrm{aff}(v_0, \dots, v_j))^2 \right)^{1/2}, \tag{3}$$

where $\mathrm{dist}(x; V)$ *denotes the distance of the point x from the space V.*

In formula (3), $\omega = (\omega_0, \dots, \omega_n)$ is interpreted as vector of weights, where ω_k denotes the weight of the affine subspace of dimension k.

Remark 1. It is easy to notice, that DIST has the following properties:

- for $\mathrm{v} \in \mathrm{E}_n(\mathbb{R}^N)$ and $\omega = (0, \dots, 0, 1) \in [0,1]^{n+1}$ we obtain that $\mathrm{DIST}_\omega(x; \mathrm{v})$ is a distance between the point x and affine space $\mathrm{aff}(\mathrm{v})$;
- if $\mathrm{v}_0 = 0$ and $\omega = (0, \dots, 0, 1)$ then DIST_ω is a distance between point and linear space generated by $(v_1, \dots, v_n)$;
- if $\omega = (1, 0, \dots, 0)$ then DIST_ω is the classical distance between x and v_0:

$$\mathrm{DIST}_\omega(x; \mathrm{v}) = \|x - v_0\|.$$

- if $\omega = \left(\underbrace{0, \dots, 0}_{k}, \underbrace{\frac{1}{l-k}, \dots, \frac{1}{l-k}}_{k-l}, 0, \dots, 0 \right)$ for $k < l$, then DIST_ω describes the mean distance between x and subspaces of dimension from k to l.

Corollary 1. *Formula (3) can be reformulated as follows*

$$\begin{aligned} (\mathrm{DIST}_\omega(x; \mathrm{v}))^2 &= \sum_{j=0}^{n} \omega_j \left(\|x - v_0\|^2 - \sum_{i=1}^{j} \langle x - v_0; v_i \rangle^2 \right) \\ &= \sum_{j=0}^{n} \omega_j \|x - v_0\|^2 - \sum_{j=0}^{n} \omega_j \sum_{i=1}^{j} \langle x - v_0; v_i \rangle^2. \end{aligned}$$

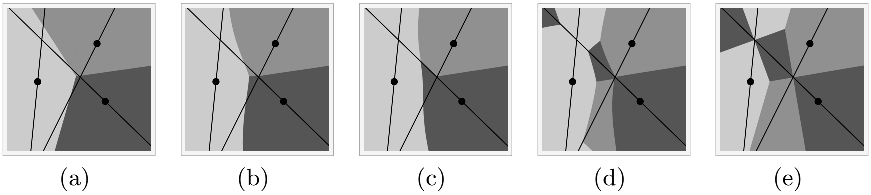


Fig. 4. Generalized Voronoi diagram for clustering of 3 clusters for different weight vectors Fig. 4(a), $\omega = (1, 0)$; Fig. 4(b), $\omega = (\frac{3}{4}, \frac{1}{4})$; Fig. 4(c), $\omega = (\frac{1}{2}, \frac{1}{2})$; Fig. 4(d), $\omega = (\frac{1}{4}, \frac{3}{4})$; Fig. 4(e), $\omega = (0, 1)$

To optimize calculations we define

$$\bar{v}_1 = \langle x - v_0; v_1\rangle^2, \quad \bar{v}_j = \bar{v}_{j-1} + \langle x - v_0; v_j\rangle^2,$$

and since $\sum \omega_j = 1$ *we can simplify our formula to*

$$(\mathrm{DIST}_\omega(x; \mathrm{v}))^2 = \|x - v_0\|^2 - \sum_{j=0}^{n} \omega_j \bar{v}_j.$$

Now we are ready to define our generalization of the Voronoi diagram. Let S be a finite subset of $\mathrm{E}_n(\mathbb{R}^N)$ and $\omega \in [0, 1]^{n+1}$, $\sum \omega_j = 1$, where $n \leq N$. For $\mathrm{p}, \mathrm{q} \in S$ such, that $\mathrm{p} \neq \mathrm{q}$, let

$$B_\omega(\mathrm{p}, \mathrm{q}) := \{z \in \mathbb{R}^N : \mathrm{DIST}_\omega(z; \mathrm{p}) = \mathrm{DIST}_\omega(z; \mathrm{q})\},$$

$$D_\omega(\mathrm{p}, \mathrm{q}) := \{z \in \mathbb{R}^N : \mathrm{DIST}_\omega(z; \mathrm{p}) < \mathrm{DIST}_\omega(z; \mathrm{q})\}.$$

The set $B_\omega(\mathrm{p}, \mathrm{q})$ divides the space $\mathbb{R}^N$ into two subsets, containing points which are closer to p then to q ($D_\omega(\mathrm{p}, \mathrm{q})$) and points which are closer to q then p ($D_\omega(\mathrm{q}, \mathrm{p})$).

Definition 2. *Let* $n \in \mathbb{N}$, $n < N$ *be fixed. Let* S *be a finite subset of* $\mathrm{E}_n(\mathbb{R}^N)$ *and* $\omega \in [0, 1]^{n+1}$, $\sum_{j=0}^{n} \omega_j = 1$ *be given. For* $\mathrm{p} \in S$ *the set*

$$D_\omega(\mathrm{p}, S) := \bigcap_{\mathrm{q} \in S:\ \mathrm{q} \neq \mathrm{p}} D_\omega(\mathrm{p}, \mathrm{q})$$

of all points that are closer to p *than to any other element of* S *is called the (open) generalized Voronoi region of* p *with respect to* S.

Applying this definition we obtain a new type of Voronoi diagram. Figure 4 presents a generalized diagram on the plane for different weights changing from $\omega = (1, 0)$ to $\omega = (0, 1)$. In general we obtain that the boundary sets usually are not polygons but zeros of quadratic polynomials. The same happens in $\mathbb{R}^3$ even for $\omega = (0, 1)$ see the Fig. 5, where we show points with equal distance from two lines.

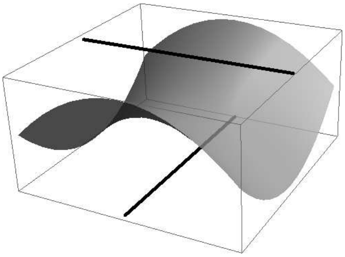


Fig. 5. Generalized Voronoi diagram for $\omega = (0, 1)$ and two lines

3 Generalization of the k-means Method

Clustering is a classical problem of the division of a set $S \subset \mathbb{R}^N$ into separate clusters, or in other words, into sets showing given type of behavior.

One of the most popular and basic method of clustering is the k-means algorithm. By this approach we want to divide S into k clusters $S_1, \ldots, S_k$ with minimal energy. For convenience of the reader and to establish the notation we shortly present the k-means method.

For a cluster S and $r \in \mathbb{R}^N$ we define the cost function

$$E(S, r) := \sum_{s \in S} \|s - r\|^2$$

which we interprets as an energy. We say that the point $\overline{r}$ best "describes" the set S if energy is minimal, more precisely, if

$$\mathrm{E}(S, \overline{r}) = \inf_{r \in \mathbb{R}^N} \{\mathrm{E}(S, r)\}.$$

It is easy to show that barycenter (mean) of S minimizes the function $\mathrm{E}(S, \cdot)$ (for more information see [2,3]). The above consideration can be precisely formulated as follows:

Theorem 1 (k-means). *Let S be a finite subset of $\mathbb{R}^N$. We have*

$$\mathrm{E}(S, \mu(S)) = \inf_{r \in \mathbb{R}^N} \{\mathrm{E}(S, r)\}$$

where $\mu(S) := \frac{1}{\mathrm{card} S} \sum_{s \in S} s$ denotes the barycentre of S.

Thus in the k-means the goal is to find such clustering $S = S_1 \cup \ldots \cup S_k$ that the function $\mathrm{E}(S_1, \ldots, S_k) = \sum_{j=1}^{k} \mathrm{E}(S_j, \mu(S_j))$ is minimal.

In this paper we consider generalization of k-means algorithm similar to that from the previous section concerning the Voronoi diagram. Instead of looking for points which best "describe" clusters we seek n dimensional subspaces of $\mathbb{R}^N$.

Let $S \subset \mathbb{R}^N$ and $\omega \in [0, 1]^{n+1}$, $\sum \omega_j = 1$ be fixed. For $\mathrm{v} \in \mathrm{E}_n(\mathbb{R}^N)$ let

$$\mathrm{E}_\omega(S, \mathrm{v}) := \sum_{s \in S} \mathrm{DIST}^2_\omega(s, \mathrm{v}).$$

We interpret the function $\mathrm{E}_\omega(S, \mathrm{v})$ as an energy of the set S respectively to the subspace generated by v. If the energy is zero, the set S is subset of affine space generated by v. We say that $\overline{\mathrm{v}}$ best "describes" the set S if the energy is minimal, more precisely if

$$\mathrm{E}_\omega(S, \overline{\mathrm{v}}) = \inf_{\mathrm{v} \in \mathrm{E}_n(\mathbb{R}^N)} \{\mathrm{E}_\omega(S, \mathrm{v})\}.$$

To obtain an optimal base we use a classical Karhunen-Loeve̗ transform (called also Principal Component Analysis, shortly PCA), see [5]. The basic idea behind the PCA is to find the coordinate system in which the first few coordinates give us a "largest" possible information about our data.

Theorem 2 (PCA). *Let $S = \{s_1, \ldots, s_m\}$ be a finite subset of $\mathbb{R}^N$. Let*

$$\mathcal{M}(S) := (v_0, \ldots, v_N) \in \mathrm{E}_N(\mathbb{R}^N)$$

be such that

- $v_0 = \mu(S)$*;*
- $v_1, \ldots, v_N$ *are pairwise orthogonal eigenvectors of* $[s_1 - v_0, \ldots, s_m - v_0] \cdot [s_1 - v_0, \ldots, s_m - v_0]^T$ *arranged in descending order with respect to the eigenvalues.*

For every $n < N$ and $\omega \in [0,1]^{n+1}$ we have

$$\mathrm{E}_\omega(S, \mathcal{M}_k(S)) = \inf_{\mathrm{v} \in \mathrm{E}_n(\mathbb{R}^N)} \{\mathrm{E}_\omega(S, \mathrm{v})\},$$

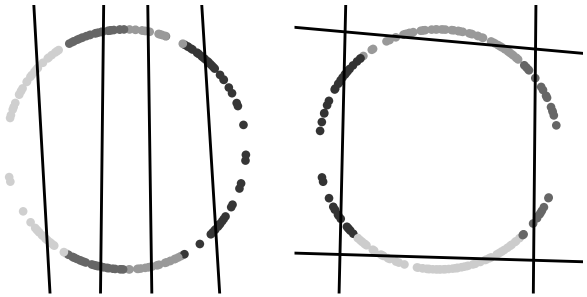
where $\mathcal{M}_k(S) := (v_0, \ldots, v_k)$.

Thus given $\omega \in [0,1]^{n+1}$, $\sum \omega_j = 1$, in (w, k)-means our goal is to find such clustering $S = S_1 \cup \ldots \cup S_k$ that the function

$$\mathrm{E}_\omega(S_1, \ldots, S_k) := \sum_{j=1}^{k} \mathrm{E}_\omega(S_j, \mathcal{M}_n(S)) \tag{4}$$

is minimal. Consequently (ω, k)-means algorithm can be described as follows:

```
stop condition
  choose $\varepsilon > 0$
initial conditions
  choose randomly points $\{\overline{s}_1, \ldots, \overline{s}_k\} \subset S$
  obtain first clustering $(S_1, \ldots, S_k)$ by matching each of the points $s \in S$ to
  the cluster such that $\|s - \overline{s}_j\|^2$ is minimal
repeat
    let $\mathrm{E} = \mathrm{E}_\omega(S_1, \ldots, S_k)$
    compute vectors $\mathrm{v}^1, \ldots, \mathrm{v}^k$, which best "describe" clusters, by the PCA
    method $(\mathrm{v}_j = \mathcal{M}_n(S_j))$
    obtain new clustering $(S_1, \ldots, S_k)$ by adding each of the point $s \in S$ to
    the cluster such that $\mathrm{DIST}_\omega(s, \mathrm{v}_j)$ is minimal
until $\mathrm{E} - \mathrm{E}_\omega(S_1, \ldots, S_k) < \varepsilon$
```



(a) local minimum (b) global minimum

Fig. 6. Circle clustering in $\mathbb{R}^2$ for 4 clusters with $\omega = (0, 1)$. (ω, k)-means method strongly dependents on initial conditions.

As is the case in the classical k-means, our algorithm guarantees a decrease in each iteration but does not guarantee that the result will be optimal. It is easy to notice that the above method has following properties:

1. for $\omega = (1, 0, \ldots, 0)$ we obtain the classical k-means,
2. for $n = 1$ we get Karhunen-Loéve transform.

Example 1. As already mentioned in Section 2, the k-means do not find a global minimum and strongly depends on initial selection of clusters. In our case, this effect can be even more visible. Consider the case of circle C in $\mathbb{R}^2$ with 4 clusters and $\omega = (0, 1)$. The picture, see Figure 6(a), shows clustering obtained by use (ω, k)-means algorithm. Of course it is a local minimum of E_ω, however as we see at Figure 6(b) it is far from being the global minimum.

4 Determining the Dimensions of Subspaces

In this section we present the method of determining the optimal dimensions subspaces which describe clusters. By using various value of parameter ω we are able to compress data by remembering different coordinates in each cluster.

Let k (the number of clusters) and ω (weight parameter) be fixed. As a result of the (ω, k)-means algorithm for the dataset S we obtain k clusters $\{S_1, \ldots, S_k\}$ and k coordinate systems $\{\mathrm{v}^1, \ldots, \mathrm{v}^k\} \subset \mathrm{E}_n(\mathbb{R}^N)$. For $\mathrm{v} \in \mathrm{E}_n(\mathbb{R}^N)$ and $n_0 \leq n$ we define sub–base of dimension n_0 by

$$\mathrm{v}_{n_0} = (v_0, \ldots, v_{n_0}).$$

We choose $n_1, \ldots, n_k \in \{1, \ldots, N\}$ and we compress the data of S by replacing each element $s \in S_i$ by its orthogonal projection on a suitable subspace spanned on v_{n_i}.

Let $S_1,\ldots,S_k$ and $\{\mathrm{v}^1,\ldots,\mathrm{v}^k\} \subset \mathrm{E}_n(\mathbb{R}^N)$ be a result of the (ω,k)-mean algorithm. For parameters $n_1,\ldots,n_k$ we consider the compression error

$$\mathrm{Comp\_err}(n_1,\ldots,n_k) := \left(\sum_{i=1}^{k}\sum_{s\in S_i}\mathrm{dist}^2(s;\mathrm{v}^i_{n_i})\right)^{1/2}.$$

Let $\varepsilon > 0$ be given a maximal allowed error. We want to find the minimal number of parameters to "compress" the data with compression error below ε. Observe that if we approximate S_i by its projection onto subspaces of dimension n_i, the total number of parameters is given by

$$n_1 \cdot \mathrm{card}(S_1) + \ldots + n_k \cdot \mathrm{card}(S_k).$$

The procedure of determining respective dimensions $n_1,\ldots,n_k$ of clusters, such that

$$\mathrm{Comp\_err}(n_1,\ldots,n_k) < \varepsilon$$

can be formulated as follows:

1. Apply the (ω,k)–means algorithm with given k (in general this parameter should be chosen respectively to data structure) and ω (which describe possible dimensions of clusters).
2. In each cluster $S_1,\ldots,S_k$ determinate eigenvalue of covariance matrix
$$\lambda^j_1 \geq \ldots \geq \lambda^j_N \text{ for } j = 1,\ldots,k.$$
3. Put
$$\varLambda := \left\{\lambda^{l_1}_1,\ldots,\lambda^{l_{kN}}_{kN}\right\}.$$
4. Sort the eigenvalues increasingly
$$\varLambda_{(\cdot)} = \left\{\lambda^{l_{(1)}}_{(1)},\ldots,\lambda^{l_{(kn)}}_{(kn)}\right\}.$$
5. Let
$$\bar{n} := \sup\left\{n\colon \sum_{i=1}^{n}\lambda^{l_{(i)}}_{(i)} \cdot m_{l_{(i)}} \leq \varepsilon\right\},$$
where $m_i = \mathrm{card}(S_i)$, for $i = \{1,\ldots,k\}$.
6. We define $n_1,\ldots,n_k$ by
$$n_j = \mathrm{card}\left\{\lambda^{l_(i)}_{(i)}\colon \text{ such that } l_{(i)} = j \text{ and } (i) > \bar{n}\right\}.$$

Before we show that this algorithm gives good accuracy we present following theorem

Lemma 1 ([5]). *Let* $S = \{x_1,\ldots,x_n\}$ *be subset of* $\mathbb{R}^N$. *By* $\{\lambda_1,\ldots,\lambda_n\}$ *we denote eigenvalues corresponding to eigenvectors* $\{v_1,\ldots,v_n\}$ *of matrix* $\mathrm{cov}([x_1,\ldots,x_n])$. *Then*

$$\sum_{x\in S}\mathrm{dist}^2(x,\mathrm{v}_k) = \sum_{i=k+1}^{n}\lambda_i \cdot n,$$

where $\mathrm{v}_k = \{v_0,v_1,\ldots,v_k\}$.

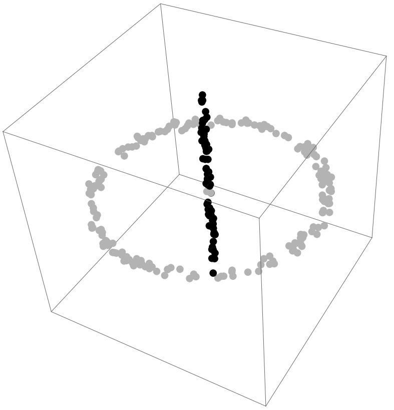


Fig. 7. Clustering with (ω, k)-means for $\omega = (0, \frac{1}{2}, \frac{1}{2})$

Now by simple calculations we have

$$\text{Comp\_err}(n_1, \ldots, n_k) = \left(\sum_{i=1}^{k} \sum_{s \in S_i} \text{dist}(s; \text{v}_{n_i})^2 \right)^{1/2}$$

$$= \left(\sum_{i}^{k} \sum_{j=n_i+1}^{m_i} \lambda_j^i \cdot m_i \right)^{1/2} = \left(\sum_{i=1}^{\bar{n}} \lambda_{(i)}^{l_{(i)}} \cdot m_{l_{(i)}} \right)^{1/2} < \varepsilon.$$

Example 2. Consider the dataset containing points group around a segment (200 points) and circle (200 points) – see Figure 7. In first step we fix $\varepsilon = 2.87$ (which gives 5.5% of total error[1]). Then to start our algorithm we have to choose the parameters k and ω. In our example we want to obtain two clusters, so we fix $k = 2$. Moreover the first cluster should represent the one–dimension data and the second two–dimension. So we put $\omega = (0, \frac{1}{2}, \frac{1}{2})$. Outcome obtained at the end of calculation is presented in Table 1. Cluster S_1 corresponds to points grouped along interval, and the S_2 – along circle.

Now by steps 3–6 we have

$$\Lambda_{(\cdot)} = \{0.001, 0.001, 0.040, 1.885, 2.124, 9.241\},$$

$$\bar{n} = 3,$$

$$n_1 = 1, \quad n_2 = 2.$$

Consequently, we get $2 \cdot 199$ parameters for S_2 and $1 \cdot 201$ for S_1.

At the end of this section we presen our algorithm on the example of the classical Lena picture.

[1] By total error we understand error obtained in the worst case of compression, when we replace each element in each cluster by barycenter of all data.

Table 1. Outcome of the (ω, k)-means in the case of data from Example 2

	S_1	S_2
$\mu(S_i)$	(-0.048, -0.027, 0.0)	(-0.004, 0.002, -0.012)
eigenvector	$\begin{bmatrix} 0.0 & 0.534 & -0.845 \\ 0.0 & -0.845 & -0.534 \\ 1.0 & 0.0 & 0.0 \end{bmatrix}$	$\begin{bmatrix} 0.692 & -0.722 & 0.001 \\ 0.722 & 0.692 & 0.002 \\ 0.003 & 0.0 & -1.0 \end{bmatrix}$
eigenvalue	(9.241, 0.040, 0.001)	(2.124,1.885,0.001)

Example 3. Let us consider the classical Lena image. First, we interpret photo as a matrix. We do this by dividing a picture into disjoint squares (8 by 8) pixels, where each of them is described (in RGB) by using 3 parameters. Consequently we have vectors from $\mathbb{R}^{192}$. Let $\varepsilon = 427$ (which gives 1% of total error) be fixed. We use $k = 5$. Then we have to choose ω. If we do not have any intuition about possible dimension of cluster we can put

$$\omega = \left(\frac{1}{192}, \ldots, \frac{1}{192}\right).$$

Since in picture compression we expect the data to have lower dimensional structure[2], we narrow our consideration to subspaces of dimension between 10–20 by choosing, according to Remark 1,

$$\omega = \left(\underbrace{0, \ldots, 0}_{1-10}, \underbrace{\frac{1}{10}, \ldots, \frac{1}{10}}_{11-20}, \underbrace{0, \ldots, 0}_{21-192}\right).$$

By applying points 3–6 we obtain:

- $1 \cdot 2375$ parameters for the first cluster,
- $2 \cdot 151$ parameters for the second cluster,
- $5 \cdot 880$ parameters for the third cluster,
- $4 \cdot 229$ parameters for the fourth cluster,
- $3 \cdot 461$ parameters for the fifth cluster.

As we see by use our method we have to remember 9376 parameters. If we fix $n_1 = \ldots = n_5$ such that $\text{Comp\_err}(n_1, \ldots, n_5) < \varepsilon$ (4 first eigenvalues for each cluster) we obtain 16384 parameters – which is all most twice as much as in our method.

[2] That why the compression based on Karhunen–Loéve transform or JPG format gives good results.

Sample implementation of (ω, k)-means algorithm prepared in Java programming language is available at [9].

References

1. Agarwal, P.K., Mustafa, N.H.: k-Means projective clustering. In: Proceedings of the Twenty-third ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, pp. 155–165 (2004)
2. Ding, C., He, X.: K-means clustering via principal component analysis. In: Proceedings of the Twenty-first International Conference on Machine Learning, vol. 29 (2004)
3. Fisher, W.: On grouping for maximum homogeneity. Journal of the American Statistical Association, 789–798 (1958)
4. Grim, J.: Multimodal discrete Karhunen-Loéve expansion. In: Institute of Information Theory and Automation AS CR Kybernetika, pp. 329–330 (1986)
5. Jolliffe, I.: Principal component analysis. Encyclopedia of Statistics in Behavioral Science (2002)
6. Kamisiński, T., Rubacha, J., Pilch, A.: The Study of Sound Scattering Structures for the Purposes of Room Acoustic Enhancement. Acta Physica Polonica A, Polska Akademia Nauk. Instytut Fizyki, Warszawa, 83–86 (2010)
7. Klein, R.: Concrete and Abstract Voronoi Diagrams. LNCS, vol. 400. Springer, Heidelberg (1989)
8. Kriegel, H., Kröger, P., Zimek, A.: Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. ACM Transactions on Knowledge Discovery from Data (TKDD) 3(1), 1–58 (2009)
9. Misztal, K., Spurek, P. and Tabor, J.: Implementation of the (ω, k)-means algorithm (2012), `http://www2.im.uj.edu.pl/badania/preprinty/imuj2012/pr1201.zip`
10. Okabe, A.: Spatial tessellations: concepts and applications of Voronoi diagrams. John Wiley & Sons Inc. (2000)
11. Parsons, L., Haque, E., Liu, H.: Subspace clustering for high dimensional data: a review. ACM SIGKDD Explorations Newsletter 6(1), 90–105 (2004)
12. Vidal, R.: Subspace clustering. IEEE Signal Processing Magazine 28(2), 52–68 (2011)

Machine Learning with Known Input Data Uncertainty Measure

Wojciech M. Czarnecki[1] and Igor T. Podolak[2]

[1] Faculty of Mathematics and Computer Science,
Adam Mickiewicz University in Poznan
w.czarnecki@amu.edu.pl

[2] Faculty of Mathematics and Computer Science,
Jagiellonian University
igor.podolak@uj.edu.pl

Abstract. Uncertainty of the input data is a common issue in machine learning. In this paper we show how one can incorporate knowledge on uncertainty measure regarding particular points in the training set. This may boost up models accuracy as well as reduce overfitting. We show an approach based on the classical training with jitter for Artificial Neural Networks (ANNs). We prove that our method, which can be applied to a wide class of models, is approximately equivalent to generalised Tikhonov regularisation learning. We also compare our results with some alternative methods. In the end we discuss further prospects and applications.

Keywords: machine learning, neural networks, classification, clustering, jitter, uncertainty, random variables.

1 Introduction

Uncertainty is a popular phenomenon in several life areas, such as medicine, image processing, linguistics, robotics, etc. Uncertainty types relate to modelled system input, output, and their representation. Previously much attention was paid mainly to the expected output, e.g. by taking into account class label together with its probability [1]. The output label uncertainty has to be approached when using an active learning methodology, e.g. with multiple oracles [2].

In the context of input data uncertainty, most frequently addressed issue is the missing attributes problem. Solutions include, e.g. explicitly modelling the expected risk [3], data imputation using EM [4], use of the grey information theory [5], etc.

On the other hand, the problem of input data uncertainty (meaning that input vectors are enriched with some kind of certainty distribution) is addressed less frequently. This is worth considering in such areas as robotics, biology, medicine, pharmacology, etc. For example, in robotics we may imagine a situation where a robot detecting an object using some methods of measurements, can also estimate its certainty of measurement correctness. It might depend on several factors: sensor type, localisation, time spent on taking the measurement,

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 379–388, 2013.

environment conditions, etc. Analogously, in experimental science, observation induced error can be estimated [6].

In this paper we investigate how one can exploit the measurement uncertainty if it is given directly as a probability density function. First, we shall propose a method of incorporating this kind of knowledge into iterative learning algorithms. Then we shall show some theoretical properties. Some experiments shall be performed as a proof of concept. Conclusions follow.

2 Methods

There are different solutions possible for the problem of learning tasks when the input attribute measurements are inaccurate. The most obvious would be to generate incorrect data. This might prove to be troublesome and the learning process might be intractable.

It is different when the inaccuracies are known in advance and given in the form of probability of distributions of points being sampled in the neighbourhood of the exact attribute value. The distribution might be given in the form of, e.g. a normal distribution with zero mean and a given covariance matrix (see Fig. 1). Given such additional knowledge, and a number generator, we may sample points to be used during learning. As we will show, when using this approach, data points from the whole input space may be labelled, and learning belongs to a class of generalised Tikhonov's regularisation [7].

This method is, naturally, applicable only in the case of iterative methods, like a steepest descent with a defined energy function. Therefore, we shall use, as an example, feed–forward neural networks. It has to be clearly pointed, that other method, e.g. clustering, may benefit from this model.

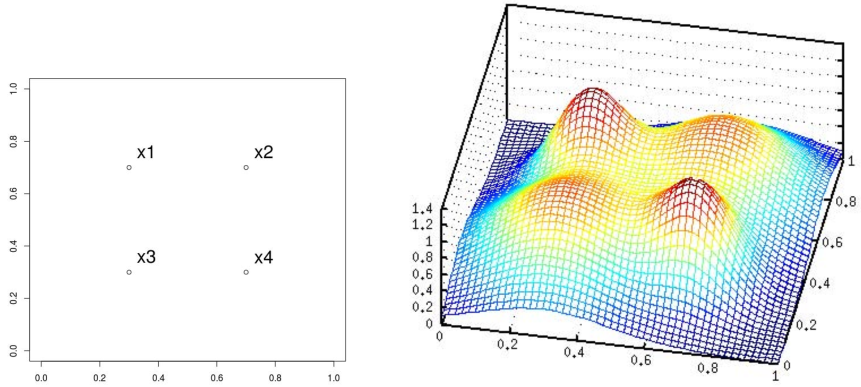


Fig. 1. Sample training set with four data points (on the left) transformed by adding to the $x1$ and $x4$ points normal distribution with zero mean and variance 0.04 and the same distribution with variance 0.01 to the remaining two points (on the right). Labels are omitted for clarity.

2.1 Sampling the Input Space

To provide data needed for learning, we need to sample the input space. Each training example is, actually, a triple (x_i, t_i, f_i) where x_i is a P-dimensional numeric attribute vector, t_i is its target label (either for a regression or classification task), while f_i is a P-dimensional probability density function. This pdf might be, e.g. a normal distribution $\mathcal{N}(0, Q_i)$, where Q_i is the covariance matrix, and gives the likelihood the actual attribute measurement is biased from its true value x_i. The actual data points to be classified by a trained model shall be $x_i + \nu$, where ν is the noise vector. In the following sections we assume that all the distributions are zero-mean and normalised.

We have two basic methods of sampling data, described in Table 1, which may easily be shown to be equivalent.

Table 1. Data sampling methods

draw from f_i individually	draw from sum of f_i
1. draw an index i corresponding to example x_i with label t_i 2. draw a random variable ν with some given probability f_i 3. return a training example $(x_i + \nu, t_i, f_i)$	1. draw a data point $\widetilde{x}$ with probability given with $\sum_i f_i$ 2. draw a label $t = t_i \sim f_i(\widetilde{x}) / \sum_j f_j(\widetilde{x})$ 3. $\nu = \widetilde{x} - x_i$ 4. return a training example $(x_i + \nu, t_i, f_i)$

2.2 Training with Jitter

Training with jitter is a method in which data is corrupted with noise [9]. This approach is designed to help generalisation. When training, the examples are taken from the training set with some small noise term ν added. Let $\widetilde{x} = x_i + \nu$. The target label is taken to be that of x_i, i.e. $t(\widetilde{x}) = t_i$. Thus, the expected value of the target is $E[t(\widetilde{x} - \nu)|\widetilde{x}] = \sum_i t_i P(i|\widetilde{x})$, where $P(i|\widetilde{x})$ denotes the probability that $\widetilde{x}$ is a noisy exemplification of example x_i. In other words, the labels are actually defined for all the points in the attribute space.

As Reed et al. [9] show, the training is equivalent to minimising the loss function $\ell(\widetilde{x}) = \left[\sum_i t_i P(i|\widetilde{x}) - y(\widetilde{x})\right]^2$ (provided a least-squares approach is used). This shows, that this training supports generalisation, which has been proven in several experiments [10,11].

Our method is to extend the training with jitter approach to the case with many different, but known, noise distributions. Such approach needs a theoretical justification, therefore we shall use the Tikhonov regularisation model as a base, which is a well studied and grounded method in machine learning. This implies many well known properties and advantages of this approach. Intuitively, thanks to distribution implied labeling of all input space data points, our method adds regularisation.

2.3 Input Uncertainty Training as a Regularisation Method

Proposition 1. *Let $\mathcal{M}$ be a least squares minimisation procedure. If we use a training set with added zero-mean, normalised random noise coming from known distributions defined separately for each training data point, then there exists a procedure $\mathcal{M}'$ approximately equal to $\mathcal{M}$, such that $\mathcal{M}'$ belongs to a class of generalised Tikhonov's regularisation.*

Proof. Let the training set be $\{(x_i, t_i, f_i | i = 1, \dots, N)\}$, where f_i pdfs are parameterised with Q_i covariance matrices. When using a gradient descent teaching function to train an ANN with weight matrix W, we may define the cost function as

$$E(W) = \sum_{i=1}^{N} \int\limits_{\mathbb{R}^K} \big[t(x_i) - y(x_i + \nu)\big]^2 P(x_i) f_i(x_i + \nu) d\nu, \tag{1}$$

where ν is the noise vector, and $P(x_i) = \int\limits_{\mathbb{R}^K} P(i|x_i + \nu)$. We can easily expand it

$$\begin{aligned}
E(W) &= \textstyle\sum_{i=1}^{N} \int\limits_{\mathbb{R}^K} \big[t^2(x_i) - 2t^2(x_i)y^2(x_i+\nu) + y^2(x_i+\nu)\big] P(x_i) f_i(x_i+\nu) d\nu \\
&= \textstyle\sum_{i=1}^{N} \big[t^2(x_i) \int\limits_{\mathbb{R}^K} f_i(x_i+\nu)d\nu - 2\int\limits_{\mathbb{R}^K} t^2(x_i)y^2(x_i+\nu)f_i(x_i+\nu)d\nu \\
&\qquad + \int\limits_{\mathbb{R}^K} y^2(x_i+\nu)f_i(x_i+\nu)d\nu\big] P(x_i) \\
&= \textstyle\sum_{i=1}^{N} \big[t^2(x_i) \int\limits_{\mathbb{R}^K} f_i(x_i+\nu)d\nu - 2t(x_i)\int\limits_{\mathbb{R}^K} y(x_i+\nu)f_i(x_i+\nu)d\nu \\
&\qquad + \int\limits_{\mathbb{R}^K} y^2(x_i+\nu)f_i(x_i+\nu)d\nu\big] P(x_i)
\end{aligned} \tag{2}$$

If the ν noise is small, we can approximate using the Taylor expansion

$$y(x+\nu) = y(x) + \left(\frac{\partial y}{\partial x}\right)^T \nu + O(\nu^2)$$

with derivatives computed at $\nu = 0$. For sufficiently small ν we can omit the last term. Substituting into (2) and expanding

$$
\begin{aligned}
E(W) \simeq \textstyle\sum_{i=1}^{N} \big[& t^2(x_i) \int\limits_{\mathbb{R}^K} f_i(x_i+\nu)d\nu - 2t(x_i) \int\limits_{\mathbb{R}^K} y(x_i+\nu) f_i(x_i+\nu)d\nu \\
& -2t(x_i) \int\limits_{\mathbb{R}^K} \left(\frac{\partial y}{\partial x}\right)^T f_i(x_i+\nu)d\nu + \int\limits_{\mathbb{R}^K} y^2(x_i+\nu) f_i(x_i+\nu)d\nu \\
& +2 \int\limits_{\mathbb{R}^K} y(x_i+\nu) \left(\frac{\partial y}{\partial x}\right)^T \nu f_i(x_i+\nu)d\nu \\
& + \int\limits_{\mathbb{R}^K} \left(\frac{\partial y}{\partial x}\right)^T \nu\nu^T \left(\frac{\partial y}{\partial x}\right) f_i(x_i+\nu)d\nu \big] P(x_i)
\end{aligned}
$$

Since $\int\limits_{\mathbb{R}^K} f_i(x_i+\nu)d\nu = 1$

$$
\begin{aligned}
E(W) &\simeq \sum_{i=1}^{N} \left[t^2(x_i) - 2t(x_i)y(x_i+\nu) + y^2(x_i+\nu)\right] P(x_i) \\
&\quad + \sum_{i=1}^{N} [\left(\frac{\partial y}{\partial x}\right)^T \nu\nu^T \left(\frac{\partial y}{\partial x}\right)] P(x_i) \\
&= \sum_{i=1}^{N} \left[t(x_i) - y(x_i+\nu)\right]^2 P(x_i) + \sum_{i=1}^{N} [\left(\frac{\partial y}{\partial x}\right)^T Q_i \left(\frac{\partial y}{\partial x}\right)] P(x_i) \\
&= \sum_{i=1}^{N} \left[t(x_i) - y(x_i+\nu)\right]^2 P(x_i) + \sum_{i=1}^{N} \left\| \frac{\partial y}{\partial x} \right\|_{Q_i}^2 P(x_i),
\end{aligned} \tag{3}
$$

where Q_i is the covariance matrix corresponding to the f_i pdf, and $\|x\|_{Q_i}^2$ is a weighted norm $x^T Q_i x$. Equation (3) is a Tikhonov regularisation. □

Observation 1. *There are four different possibilities in case of $\mathcal{N}(0, Q_i)$:*

1. *all pdfs are identical and non-correlated $Q_i = \sigma^2 I$ in that case the regularising term reduces to $\sigma^2 \sum_{i=1}^{N} \|\frac{\partial y}{\partial x}\|^2 P(x_i)$ which corresponds to results in [8], the variance determines the relevance of the regularising term,*
2. *all pdfs are identical, but correlated $Q_i = Q$; the regularising term reduces to $\sum_{i=1}^{N} \|\frac{\partial y}{\partial x}\|_Q^2 P(x_i)$, similarly, the variances determine the regularising factor but with different strength depending on the dimension,*
3. *pdfs are different and uncorrelated $Q_i = \sigma_i^2 I$; the regularising term reduces to $\sum_{i=1}^{N} \sigma_i^2 \|\frac{\partial y}{\partial x}\|^2 P(x_i)$, variance determines the relevance of the regularisation, but its strength depends on the distance from training points: in neighbourhood of training data with higher variance, this term is more important; in areas of lower certainty the function smoothness becomes more important than its exact fitting,*
4. *pdfs are correlated and different, the regularising term has the basic form $\sum_{i=1}^{N} \|\frac{\partial y}{\partial x}\|_{Q_i}^2 P(x_i)$, the strength depends now both on the distance from training points and the dimensions.*

3 Experiments

During our experiments we have used the Encog Machine Learning Framework [12] for which we developed classes to treat the input data as random variables of known distributions. Sampling is derived using the Apache Commons Mathematics Library [13] to ensure high quality of this step. We focus on the three layered feed forward neural network with sigmoidal activation functions as our model, trained using classic backpropagation algorithm with momentum, as it is a simple, well known model that can be trained in the iterative fashion. Training set was resampled from the f_i distributions after each iteration (see Fig. 2 for details). In fact, we do not need the exact closed form equations of f_i but rather we just require a sampling method (some blackboxes s_i that provides us with samples $x_i + f$, with f coming from f_i).

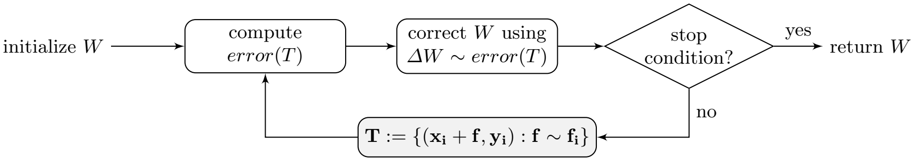

initialize W
compute $error(T)$
correct W using $\Delta W \sim error(T)$
stop condition?
yes
return W
no
$\mathbf{T} := \{(\mathbf{x_i} + \mathbf{f}, \mathbf{y_i}) : \mathbf{f} \sim \mathbf{f_i}\}$

Fig. 2. Flowchart of the proposed method. Classical iterative ANNs learning techniques include repeated weights (W) corrections according to the current *error* computed on the training set T (white blocks). In our model, the only required modification is to resample the T set from respective distributions f_i after each iteration (gray block).

Three toy experiments are described below, namely (σ^2_{ik} is the variance of the k-th example in the i-th class):

1. Training set consisting of 4 points, two of one class (triangles) located in $(0.3, 0.3)$ and $(0.7, 0.7)$ and two of the second class (circles) located in $(0.3, 0.7)$ and $(0.7, 0.3)$. Assumed distributions are zero mean Gaussian distributions with variance $\sigma^2_1 = 0.04$ and $\sigma^2_2 = 0.01$ respectively.
2. Training set consisting of 2 points, one class (triangles) located in $(0.3, 0.3)$ and second class (circles) located in $(0.7, 0.7)$. Assumed distributions are zero mean Gaussian distributions with variance $\sigma^2_1 = 0.09$ and $\sigma^2_2 = 0.01$ respectively.
3. Training set consisting of 10 points, five of one class (triangles) located in $(0.3+0.1k, 0.3)$ and five of the second class (circles) located in $(0.3+0.1k, 0.7)$ for $k \in \{0, 1, 2, 3, 4\}$. Assumed distributions are zero mean Gaussian distributions with variances $\sigma^2_{1k} = 0.025k^2$ and $\sigma^2_{2k} = 0.036k^2$ respectively.

Fig. 3 shows results of our first experiment for points classification from $[0, 1]^2$. Additional information regarding input data point uncertainty helps this model in defining more adequate decision boundary. Four points used in this example

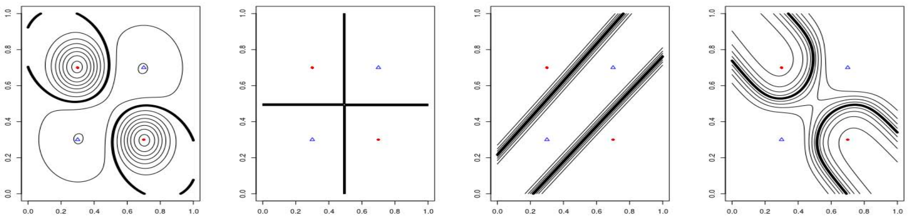


Fig. 3. Results of experiment 1, from left: Bayesian classification, Voronoi diagram, neural network classification using simple learning (discarding distribution information), our method. Thick lines indicate decision boundaries.

are fully symmetric, therefore decision boundaries found by simple neural network are parallel lines between elements of different classes. Different variances in each of the class forced neural network to bend its decision boundary towards points with higher certainty (as it is more probable that points in the unknown parts of space are parts of the second distribution).

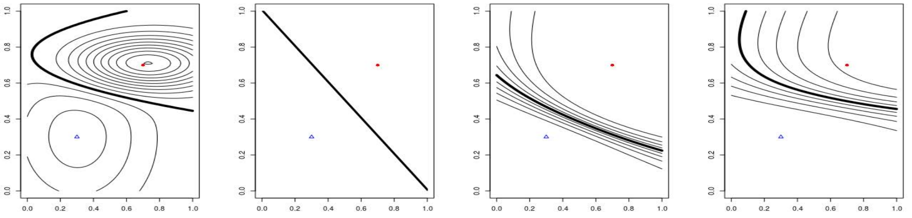


Fig. 4. Results of experiment 2, from left: Bayesian classification, Voronoi diagram, neural network classification using simple learning (discarding distribution information), our method. Thick lines indicate decision boundaries.

Fig. 4 captures similar observations like the previous one, but as the Voronoi diagram for this dataset is also a result of the support vector machine (SVM) classifier trained on such data — it shows how this additional information about relative difference in our certainty about particular data points affect the decision boundary. In this particular example model obtained by training neural network is a better approximation of the underlying distribution (generalisation) than ones returned by both SVM and an ANN.

With more data points available in the training set the effect of uncertainty measure is even more noticeable. Fig. 5 shows result of our third experiment, where one can observe the overfitting problem of the classical neural network while sampling method achieves nice, smooth decision boundary approximating underlying distribution. In particular, Fig. 6 shows results of four different runs of experiment 3, where one can see strength of the regularisation induced by uncertainty measure in comparison to the overfitting and complete indeterministic behaviour of simple neural network learning.

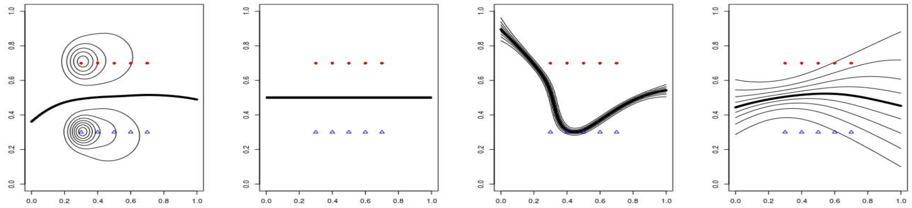


Fig. 5. Results of experiment 3, from left: Bayesian classification, Voronoi diagram, neural network classification using simple learning (discarding distribution information), our method. Thick lines indicate decision boundaries.

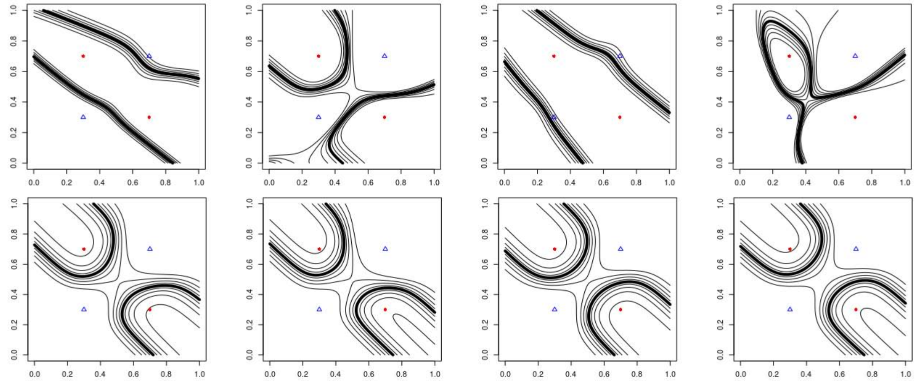


Fig. 6. Example of overfitting of a simple ANN in experiment 1 (top row) and almost indistinguishable results of multiple runs of our sampling method (bottom row)

Fig. 7 shows mean squared error during one of the experiments 1 and 3 runs using random sampling method as compared to simple training. As one can see - learning curve of the experiments 1 dataset behaves quite similarly - after short amount of time it rapidly decreases and stays in the local minimum. Although the main difference is the curve smoothness - rapid changes in its value make it unreasonable to use simple thresholding as a stop condition for our method.

There are at least three possible ways to overcome this limitation:

- train as long as possible, as Proposition 1 and experiments clearly show that such training is strongly regularised so overfitting is highly unlikely,
- use some statistics of errors instead of its value (even moving averages behaves well in our experiments),
- if we have knowledge about analytic form of our distributions sufficient to compute the regularisation term from Eq. (3) then (according to Proposition 1) we can train as long as regularised form of error equation would be higher than some threshold.

It is worth noting that such an approach actually speeds up the convergence as greedy optimisation of the smoothed function is much more rapid (see right part of Fig. 7 for reference).

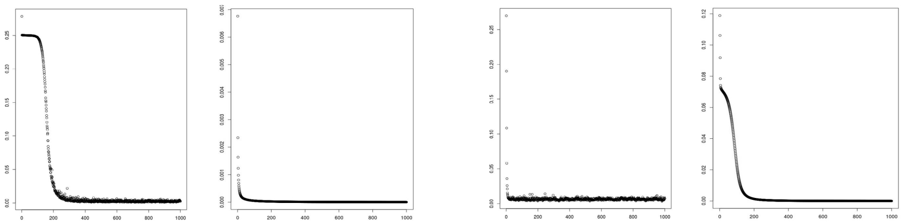


Fig. 7. Comparison of the error curves during training, from the left: experiment 1-our method, experiment 1-simple training, experiment 3-our method, experiment 3-simple training. On x axis - training step, on y axis - mean squared error on training data.

4 Conclusions and Future Work

In this paper we have proposed methods of incorporating knowledge on input data uncertainty into the learning process for a range of machine learning models, ANNs in particular. We have proved, both theoretically and empirically, that this approach is valid and yields valuable results. In particular, expressing it as a generalised Tikhonov regularisation shows its immunity to the overfitting problem.

It remains an open question what are the exact conditions which the data pdfs have to fulfil in order to insure convergence to the minimum of the regularising functional. In particular, it is obvious that uniform distribution, e.g. with $n = 2$, may lead to oscilations. Therefore, can we use discrete distributions in general?

The future work will concentrate on application of this methodology to clustering problems, e.g. Kohonen or K-means algorithms, with particular interest in convergence requirements. We also want to extend the active learning paradigm in direction of widening the spectrum of queries [14]. In addition to just asking for a label of a particular data point, we could also query for a corrected measurement, which should have higher certainty and smaller variance. Both have applications in, e.g. robotics.

We also plan to investigate applicability of such model in complex classifiers like Hierarchical Classifier with Overlapping Classes (HCOC [15]) which could use it as a node level supervised classifier, with decresing noise variance on deeper tree levels.

Acknowledgment. This work is partially financed by the National Science Centre grant *Hierarchical Classifier model. Fusion of supervised learning and non-supervised clustering into sub-problems. Theoretical problems and practical examples* (N N519 654480).

References

1. Niaf, E., Flamary, R., Lartizien, C., Canu, S.: Handling uncertainties in SVM classification. In: IEEE Statistical Signal Processing Workshop, pp. 757–760 (2011)
2. Ni, E.A., Ling, C.X.: Active Learning with c-Certainty. In: Tan, P.-N., Chawla, S., Ho, C.K., Bailey, J. (eds.) PAKDD 2012, Part I. LNCS, vol. 7301, pp. 231–242. Springer, Heidelberg (2012)
3. Pelckmans, K., De Brabanter, J., Suykens, J.A.K., De Moor, B.: Handling missing values in support vector machine classifiers. Neural Networks 18, 684–692 (2005)
4. Zhang, S.S., Wu, X., Zhu, M.: Efficient missing data imputation for supervised learning. In: 9th IEEE Int. Conf. on Cognitive Informatics, pp. 672–679 (2010)
5. Han, B., Xiao, S., Liu, L., Wu, Z.: New methods for filling missing values by grey relational analysis. In: Int. Conf. on Artificial Intelligence, Management Science and Electronic Commerce, pp. 2721–2724 (2011)
6. Coleman, H.W., Steele, W.G.: Experimentation, validation and uncertainty analysis for engineers. John Wiley and Sons (2009)
7. Tikhonov, A.N., Arsenin, V.Y.: Solutions of ill–posed problems. V.H. Winston (1977)
8. Bishop, C.M.: Training with noise is equivalent to Tikhonov regularization. Neural Computation 7(1), 108–116 (1995)
9. Reed, R., Oh, S., Marks, R.J.: Regularization using jittered training data. In: IEEE Int. Joint Conf. on Neural Networks, pp. 147–152. IEEE Press (1992)
10. Sietsma, J., Dow, R.J.F.: Creating artificial neural networks that generalise. Neural Networks 4(1), 1481–1497 (1990)
11. Weigand, A.S., Rumelhart, D.E., Huberman, B.A.: Generalization by weight elimination applied to currency exchange rate prediction. In: Int. Joint Conf. on Neural Networks, pp. 1–837 (1991)
12. Heaton, J.: Programming neural networks with Encog 3 in Java. Heaton Research Inc. (2011)
13. Apache Commons Mathematics Library, `http://commons.apache.org/math`
14. Settles, B.: Active Learning, Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers (2012)
15. Podolak, I., Roman, A.: Theoretical Foundations and Experimental Results for a Hierarchical Classifier with Overlapping Clusters. Computational Intelligence 29(2), 357–388 (2013)

Learning Algorithms in the Detection of Unused Functionalities in SOA Systems

Ilona Bluemke and Marcin Tarka

Institute of Computer Science, Warsaw University of Technology,
Nowowiejska 15/19, 00-665 Warsaw, Poland
I.Bluemke@ii.pw.edu.pl

Abstract. The objective of this paper is to present an application of learning algorithms to the detection of anomalies in SOA system. As it was not possible to inject errors into the "real" SOA system and to analyze the effect of these errors, a special model of SOA system was designed and implemented. In this system several anomalies were introduced and the effectiveness of algorithms in detecting them were measured. The results of experiments can be used to select efficient algorithm for anomaly detection. Two algorithms: K-means clustering and Kohonen networks were used to detect the unused functionalities and the results of this experiment are discussed.

1 Introduction

With the growth of computer networking, electronic commerce, and web services, security of networking systems has become very important. Many companies now rely on web services as a major source of revenue. Computer hacking poses significant problems to these companies, as distributed attacks can make their systems or services inoperable for some period of time. As this happens often, an entire area of research, called Intrusion Detection, is devoted to detect these activities.

Nowadays many system are based on Service Oriented Architecture (SOA) [1,2] idea. A system based on SOA provides functionalities as a suite of interoperable services that can be used within multiple, separate systems from several business domains. SOA also provides a way for consumers of services, such as web-based applications, to be aware of available SOA-based services. Service-orientation requires loose coupling of services with operating systems, and other technologies that underly applications. SOA separates functionality into distinct units, or services, which developers make accessible over a network in order to allow users to combine and reuse them in the production of applications.

The objective of this paper is to present the detection of anomalies in SOA systems by learning algorithms. Related work is presented in section 2 and in section 3, a special model of SOA system which was used in experiments, is presented. In this systems several anomalies were introduced. Four algorithms: Chi-Square statistics, k-means clustering, emerging patterns and Kohonen networks were used to detect anomalies. Detection of anomalies by k-means and Kohonen networks is presented in section 4 and some conclusions are given in section 5.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 389–400, 2013.

2 Related Work

Many anomaly detection algorithms have been proposed in the literature. They differ according to the information used for analysis and according to techniques that are used to detect deviations from normal behavior. Lim and Jones in [3] proposed two types of anomaly detection techniques based on employed techniques: the learning model method and the specification model.

The *learning* approach is based on the application of machine learning techniques, to automatically obtain a representation of normal behaviors from the analysis of system activities. The *specification-based* approach requires that someone manually provides specifications of correct behavior. Approaches that concern the model construction are presented in Fig. 1.

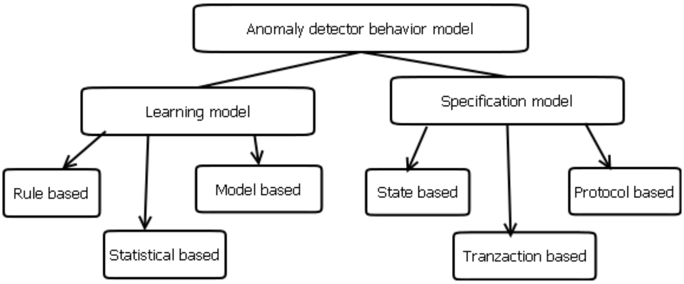


Fig. 1. Taxonomy of anomaly detection behavioral model (based on [3])

The *specification* approach depends more on human observation and expertise than on mathematical models. It was first proposed by C. Ko et. al. [4] and uses a logic based description of expected behavior to construct a base model. This specification-based anomaly detector monitors multiple system elements, ranging from application to network traffic.

In the *protocol based* approach [5] a normal use model is built from the protocol specification e.g. TCP/IP. Lemonnier [5] proposed a protocol anomaly filter able to specifically analyze a protocol and model the normal usage of a specific protocol. This technique can be seen as a filter looking for protocol misuse. The protocol could be interpreted as any official set of rules describing the interaction between the elements of a computer system.

Many protocol anomaly detectors are built as *state* machines [6]. Each state corresponds to a part of the connection, such as a server waiting for a response from client. The transitions between the states describe the legal and expected changes between states. Besides, Z. Shan et. al. [7] uses network state based model approach to describe intrusion attacks.

In the *transaction* based approach the "positive" behavior is formally described. The desired actions and sequence of actions are specified by the definition of transactions. Such explicit definition makes the transaction an integral part of security policy. In the research proposed by R. Buschkes et. al. [8] the detection of anomalies is based on the definition of correct transactional behavior.

The *learning* model must be trained on the specific network. In the training phase, the behavior of the system is observed and logged and machine learning techniques are used to create a profile of normal behaviors. In the process of creating an effective anomaly detection model: *rule-based*, *model-based*, and *statistical-based* approaches have been adopted to create the baseline profile.

Rule-based systems used in anomaly detection describe the normal behavior of users, networks and/or computer systems by a set of rules. These predefined rules typically look for the high-level state change patterns observed in the audit data.

SRI International's Next-generation Intrusion Detection Expert System (NIDES) [9] is an innovative statistical algorithm for anomaly detection, and an expert system that encodes known intrusion scenarios. The features considered by NIDES are related to user activity, for instance, CPU and file utilization. The information collected by NIDES is compared with the long - term profile of analyzed system and used for the learning process based on Chi-squared statistic. Owens et. al. present in [10] an adaptive expert system for intrusion detection based on fuzzy sets.

In the *model based* anomaly detector the intrusions are analyzed at a higher level of abstraction than the audit records of the rule based approach. Execution is restricted to pre-computed patterns of expected behavior. Different types of models are used to characterize the normal behavior of the monitored system like *data mining*, *neural networks*, *pattern matching, etc.*

Data mining extracts the behavioral models from a large amount of data. Intrusion detection systems require frequent adaptation to resist new attacks, so data mining techniques that can adaptively build new detection models are very useful. An example of data mining system was proposed by Lee et. al. and is presented in [11]. The key idea is to mine network audit data, then use the patterns to compute inductively learned classifiers that can recognize anomalies and known intrusions.

A *neural network* learns the user profile in the training process. Training may use data more abstract than the audit records. The fraction of incorrectly predicted events constitutes the variance of the user behavior. [12] describes anomalies detection in information system based on Kohonen networks. There are several libraries supporting building neural networks e.g. [13 - 15]. Other neural networks based systems for intrusion detection are also described in [16- 19].

In the *pattern matching* approach, learning is used to build a traffic profile for a given network. Traffic profiles are built using features such as packet loss, link utilization, number of collisions. Normal behavior is captured as templates and tolerance limits are set based on different levels of standard deviation. The usage of emerging patterns in anomaly detection is described in [20].

First *statistical based* anomaly detection was proposed by Denning and Neumann [21] in 1985. The anomaly detector observes subjects and generates profiles for them that represent their behavior. The widely known techniques in statistics can often be applied; e.g. data points that lie beyond a multiple of the standard deviation on either side of the mean might be considered anomalous. There are many statistical techniques like *Bayesian statistics*, *covariance matrices and Chi-square statistics* [22] for the profiling of anomaly detection. Example of *Chi-square statistic* in anomaly detection is described in [23]. Statistical approaches disadvantage is the insensitivity to the order of occurrence of events. Sequential interrelationships among events should be considered for more accurate detection. It is also difficult to determine a threshold above which an anomaly should be considered.

Commercial products applying rule based, statistical based, model based and neural networks approach to detected anomalies are briefly described in [3].

3 Research Model

As we were not allowed to use real SOA system and inject anomalies into it, a special system was implemented. The business idea of this system VTV (Virtual TV) is presented in Fig. 2. The exemplary company is a virtual TV provider. The company does not have its own technical infrastructure and is associating TV digital provider with the telecommunication operator. The receiving equipment is delivered to the client by one of courier companies. The company is also using two applications: Customer Relationship Management (CRM) containing all clients data and a storage management system.

The VTV system simulates a real SOA system and enables to inject typical anomalies into its regular operation. Configurable are frequencies of : single services calls, group of services calls, processes calls, and services in a context. More details can be found in [24].

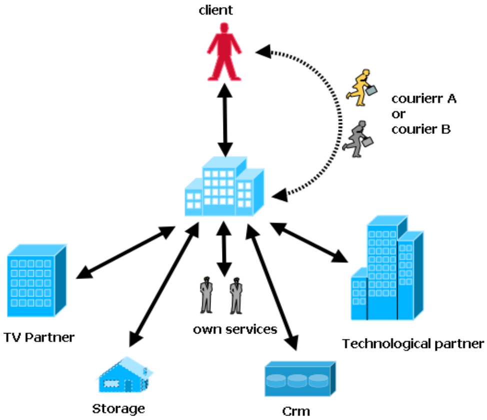


Fig. 2. The business relations of an exemplary company

The architecture of VTV system is presented in Fig. 3. The *request generator* simulates activities of clients. It generates different types of request (e.g. create, modify, deactivate) for available services (e.g. TV or hardware). Depending of the value of request some requests can be identified as very important. The generated address of a client determines the currier group. The configurable parameters of the request generator enable to simulate real operation and to inject some anomalies: e.g. request for courier, hardware, TV services can be set, ratio of create, modify, deactivate requests in generated requests can be chosen.

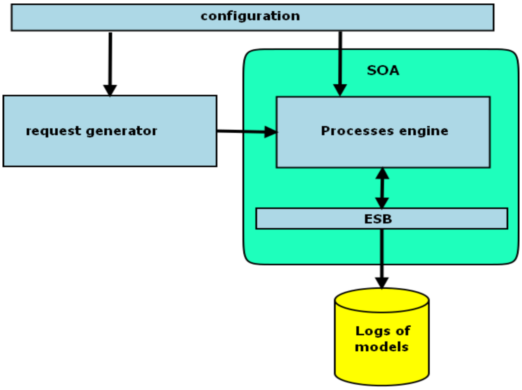


Fig. 3. Architecture of VTV system

The generated request is transferred to the *business processes engine* (Fig.3) which composes processes from services available on Enterprise Service Bus (ESB). Outputs of processes are logs. Logs of model contain information similar to logs from monitoring real SOA systems.

3.1 Environment of Experiment

All experiments were conducted on PC with Intel Core 2 Duo T7200, 512 Mb of memory under Fedora Core operating system. The examined algorithms are using text input files prepared from logs of the VTV system (Fig.3). In this log file information like number of requests, name of processes, name of services called, execution time are written. These logs files are then transformed by scripts, implemented in R [25] environment, into transactions or summarized reports which are input to detection algorithms. The flow of data from logs of the model is shown in Fig. 4.

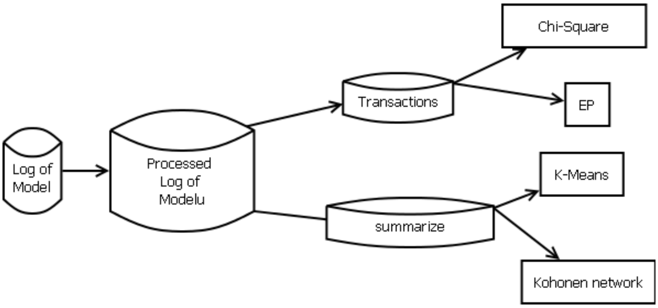


Fig. 4. Flow of data from logs

The implementation of algorithms for anomaly detection *k-means*, *Chi-Square* and *Kohonen* networks algorithms was made in R environment while the *emerging patterns* algorithm, which is more complicated, was implemented in C++.

k-means algorithm was prepared based on [26], for *emerging patterns* algorithm information from [20, 27] were used, *Chi-Square* detector was taken from [23].

4 Experiment

The research system presented in section 3 was used to explore four cases typical for SOA systems i.e.: change in the frequency of service calls, change in the frequency of a group of services, change of the context of services calls and unused functionalities.

For each of the above listed cases it was expected, that anomalies detector provides information useful for the maintenance team. Four learning algorithms (section 2) were used in the anomalies detector: *Chi-square statistic*, *Kohonen network*, *Emerging patterns* and *k-means clustering*.

Each of the above algorithms represents different approach to anomalies detection. The goal of the experiment was to examine advantages and disadvantages of each of these algorithms. Anomalies detection is a kind of clustering with two types of clusters grouping normal and abnormal behaviors. Correctly identified anomaly is an abnormal event which was introduced by purpose by case' scenario and was assigned to the abnormal cluster. Identified as anomalies other events or not recognized anomalies are treated as errors. The values measured in experiments are:

- FP (false positive) – number of incorrectly identified anomalies,
- FN (false negative) - number of not recognized anomalies,
- TP (true positive) - number of correctly identified normal behaviors,
- TN (true negative) - number of incorrectly identified normal behaviors.

Good anomalies detector should generate small number of errors. To compare the quality of detectors often sensitivity and specificity measures are used. The sensitivity and specificity are defined accordingly as:

$$sensitivit\ y = \frac{TP}{TP + FN} \tag{1}$$

and

$$specificit\ y = \frac{TN}{TN + FP} \tag{2}$$

The relation between specificity and sensitivity – ROC (Receiver Operating Characteristics) curve [28] can be used to compare the quality of models. ROC as a technique to analyze data was introduced during the second world war to identify if signal seen on a radar were coming from an enemy, an alliance or if it were noise. Currently ROC curves are used to analyze different types of data e.g. radiological data.

The construction of ROC curves during experiments was as follows:

1. create entities with algorithm' parameters
2. for each entity :
    - perform experiment
    - calculate specificity and sensitivity
    - mark the point on the diagram
3. draw the line connecting points.

For each examined algorithm also the learning time was calculated.

4.1 Plan for Experiment

The examination of anomaly detection algorithms was based on four test cases: changes in frequency of service and groups of services calls, change of the context of services calls, and lacking functionalities. Each of these cases simulates one type of anomaly typical for SOA.

The experiment was conducted in following steps:

1. Create data for regular behavior
2. For each of test case' scenario :
   - Create data for abnormal behavior in this scenario
   - For each algorithm execute:
     - Using regular data perform the learning phase
     - Perform detection phase on data for abnormal behavior
     - Evaluate the quality of detection
3. Compare algorithms.

Below the results for the scenario "*not used functionalities*" are presented. The detection of this kind of anomalies is very important in distributed systems especially in SOA systems. Some functions may be not used as a result of errors in routing algorithm or in business logic. If not used services were dedicated to some cluster this cluster could be used for other purposes. In ours research' environment this anomaly was obtained by forcing the request generator not to generate courier delivery services by setting parameter `Courier_share` to zero [24]. In [29] the detection of anomalies in the frequency of service calls by K-means clustering algorithm and emerging patterns are described.

4.2 Results of Experiments

The goal of examination was to find high level of detection with minimal number of false alarms. If the ideal detection was not possible preferred were the results with no false alarms. In practice, if system is generating many false alarms the user will neglect any alarm.

k-means Algorithm

k-means clustering [26] is a clustering analysis algorithm that groups objects based on their feature values into *k*disjoint clusters. Objects that are classified into the same cluster have similar feature values. *k*is a positive integer number specifying the number of clusters, and has to be given in advance. All objects are assigned to their closest cluster according to the ordinary Euclidean distance metric.

At the beginning the summarized reports used by the algorithm were created for all logs in the system. The results are given in Table 1. The maximal distance from centroid of cluster for a cluster member is denoted as *maxL*.

Table 1. Results for clustering algorithm

row	k	maxL	logSize	TP	TN	FP	FN	Sensitivity	Specificity
1	4	1	150	0	1	14	23	1	0.6
2	4	2	150	0	3	12	23	0	0.2
3	4	2.3	150	23	9	6	0	1	0.6
4	4	2.5	150	23	3	12	0	1	0.2
5	5	2	150	23	5	10	0	1	0.33
6	5	2.3	150	23	6	9	0	1	0.4
7	5	2.5	150	23	8	7	0	1	0.53
8	4	1	350	10	4	3	0	1	0.57
9	4	1.5	350	10	5	2	0	1	0.71
10	4	2	350	0	5	2	10	0	0.71
11	4	2.5	350	0	5	2	10	0	0.71
12	4	3	350	0	6	1	10	0	0.86

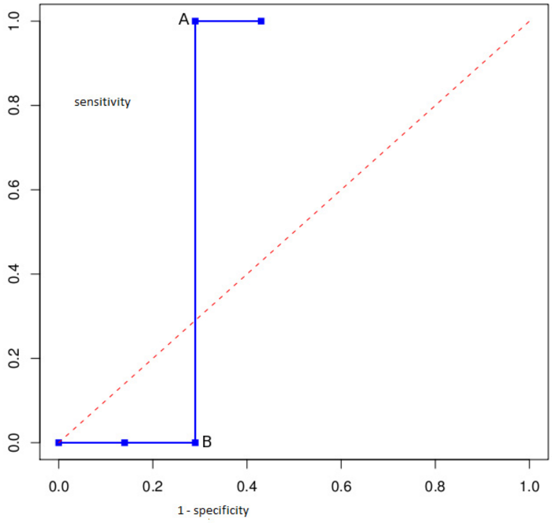


Fig. 5. ROC curve for k-means algorithm with 350 training logs

Initially the experiment was conducted for 150 logs in training set. The anomaly wasn’t satisfactorily detected, the numbers of not identified anomalies were high (e.g. 23, first row in Table 1), also the numbers of incorrectly identified anomalies were high (e.g. 14 first row in Table 1). In the training data there were many summarization reports in which the number of services not called in regular data, was close to zero, so the lack of calls of these services was treated as normal behavior. Next, the number of logs was increased to 350. The results improved (rows 8-12 Table 1).

In Fig.5 the ROC curve is shown for 350 training logs. The high change in sensitivity is seen – from zero to one. Point A was obtained by decreasing the value of `maxL` parameter from the value in point B, till the level in which zero was not assigned to the group with the minimal centroid.

The *k-means* clustering algorithm can be used in the detection of not used functionalities if the number of training data is high. In [29] we have shown that this algorithm was not suitable in detecting the change in frequencies of single services calls.

Kohonen Networks

A neural network is a set of simple, highly interconnected units called neurons. Neurons transform a set of inputs to a set of desired outputs. The result of the transformation is determined by the characteristics of the elements and the weights associated with the interconnections among them, therefore by adjusting the weight the output can be controlled. The process of updating the weights and thresholds is called learning.

Training usually takes the form of presenting the network with set of typical inputs vectors. In unsupervised learning mode (self - organizing map [30] - Kohonen networks), the inputs are presented for training; the adjustments are made so that the network is able to recognize inputs as belonging to a particular profile.

The training utilizes competitive learning. When a training example is fed to the network, its Euclidean distance to all weight vectors is computed. The neuron whose weight vector is most similar to the input is called the best matching unit (BMU). The weights of the BMU and neurons close to it in the lattice are adjusted towards the input vector. The magnitude of the change decreases with time and with distance (within the lattice) from the BMU. This process is repeated for each input vector for a number of cycles. The network winds up associating output nodes with groups or patterns in the input data set.

The results for Kohonen network algorithm are shown in Table 2. Column `Lambda` – toleration ration, shows the restraint to the distance from BMU. In rows 1-5 the numbers of not recognized (FN) and of incorrectly identified anomalies (FP) are high. After increasing the number of elements in the training phase the results of detection significantly improved. In rows 6-11 only few anomalies were not recognized.

Table 2. Results for Kohonen network algorithm

row	N	Lambda	Logsize	TP	TN	FP	FN	Sensitivity	Specificity
1	25	0.5	150	358	331	137	398	0.47	0.707
2	25	1	150	324	366	102	432	0.43	0.782
3	25	1.5	150	254	393	75	502	0.34	0.840
4	25	2	150	163	427	41	593	0.22	0.912
5	25	2.5	150	72	443	25	698	0.1	0.95
6	25	0.2	350	12	77	41	0	1	0.65
7	25	0.5	350	12	137	67	0	1	0.67
8	25	1	350	12	172	32	0	1	0.84
9	25	1.5	350	11	191	13	1	0.92	0.94
10	25	2	350	8	190	14	4	0.67	0.93
11	25	2.5	350	6	198	6	6	0.5	0.97

In Fig. 6 the ROC curve for 350 training logs is shown. Point A is optimal, with the sensitivity= 0.92 and specificity= 0.94 and the costs of false qualification of anomaly and false qualification of normal behavior are equal. These values may be improved by increasing the training data. Point B represents the case in which all anomalies were detected, the specificity was equal to 0.84.

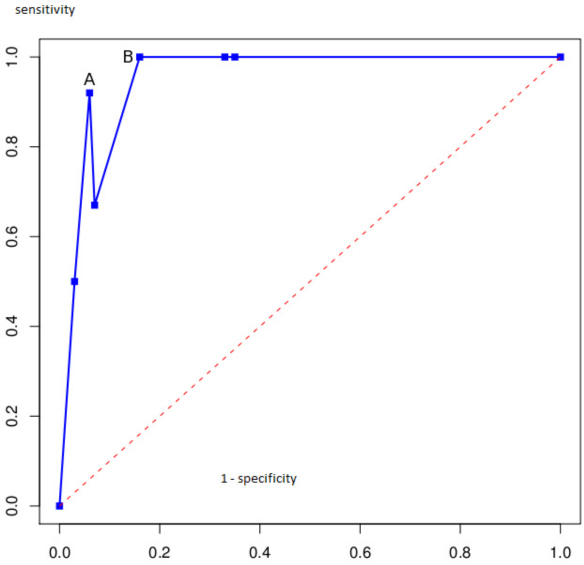


Fig. 6. ROC curve for Kohonen networks algorithm with 350 training logs

The above presented data show that Kohonen networks can be used to detect the unused functionalities but large training data must be provided.

5 Conclusions

In this paper some results of the detection of one anomaly – unused functionalities in SOA system are presented. An experiment was conducted in an environment designed to introduce several types of anomalies. This environment, described in section 3, enables the detection of anomalies by four learning algorithms, from different types (section 2): *emerging patterns*, *k-means* clustering, *Kohonen* networks and statistical *Chi-Square*. In this paper the results of only two: *k-means* clustering and *Kohonen* networks are presented. Both algorithms were able to satisfactorily detect unused functionalities while two others i.e. *Chi-Square* and *emerging patterns* appeared to be inappropriate in the detection of unused functionalities.

In [29] the results of detecting other anomaly - the change in frequencies of service calls was described. The least accurate in the detection of this kind of anomaly was the *k-means* clustering algorithm and the best was *emerging pattern* algorithm. Kohonen algorithm also produced quite good results.

The results presented in this paper and in [29] show that in anomalies detection in SOA systems different algorithms may appear most suitable for different type of anomaly so further research should be conducted. The exemplary SOA system (section 2) enables to conduct other experiments examining the suitability of learning algorithms in the detection of other anomalies. The results of these experiments will be available soon.

Acknowledgments. We are very grateful to the reviewers for many valuable remarks.

References

1. BPEL Standard, `http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.html` (access July 2011)
2. SOA manifesto, `http://www.soa-manifesto.org` (access July 2011)
3. Lim, S.Y., Jones, A.: Network Anomaly Detection System: The State of Art of Network Behavior Analysis. In: Proc. of the Int. Conf. on Convergence and Hybrid Information Technology 2008, pp. 459–465 (2008), doi:10.1109/ICHIT2008.249
4. Ko, C., Ruschitzka, M., Levitt, K.: Execution monitoring of security-critical programs in distributed systems: a specification-based approach. In: Proc. of IEEE Symposium on Security and Privacy, Oakland, CA, USA (1997)
5. Lemonnier, E.: Protocol Anomaly Detection in Network-based IDSs. Defcom white paper (2001)
6. Sekar, R., Gupta, A., Frullo, J., Shanbag, T., Tiwari, A., Yang, H., Zhou, S.: Specification-based anomaly detection: A New Approach for Detecting Network Intrusions. In: ACM Computer and Communication Security Conference, Washington, DC, USA (2002)
7. Shan, Z., Chen, P., Xu, Y., Xu, K.: A Network State Based Intrusion Detection Model. In: Proc. of the 2001 Int. Conf. on Computer Networks and Mobile Computing, ICCNMC 2001 (2001)
8. Buschkes, R., Borning, M., Kesdogan, D.: Transaction-based Anomaly Detection. In: Proc. of the Workshop on Intrusion Detection and Network Monitoring, Santa Clara, California, USA (1999)
9. Anderson, D., Frivold, T.: Valdes: A Next-generation Intrusion Detection Expert System (NIDES) Summary. Computer Science Laboratory, SRI-CSL-95-07 (May 1995)
10. Owens, S., Levary, R.: An adaptive expert system approach for intrusion detection. International Journal of Security and Networks 1, 3–4 (2006)
11. Lee, W., Stolfo, S.J.: Data mining approaches for intrusion detection. In: Proc. of the 7th USENIX Security Symposium (1998)
12. Bivens, A., Palagrini, C., Smith, R., Szymański, B., Embrechts, M.: Network-based intrusion detection using neural networks. In: Proc. Intelligent Eng. Systems through Neural Networks, ANNIE 2002, St. Louis, MO, vol. 12, pp. 579–584. ASME Press, NY (2002)
13. C Neural network library, `http://franck.fleurey.free.fr/NeuralNetwork/`
14. NeuroBox, `http://www.cdrnet.net/projects/neuro/`
15. Fast Artificial Neural Network Library, `http://sourceforge.net/projects/fann/`
16. Ryan, J., Lin, M., Miikkulainen, M.: Intrusion Detection with Neural Networks. In: Advances in Neural Information Processing Systems, vol. 10 (1998)

17. Ghosh, A.K., Schwartzbard, A.: A Study in Using Neural Networks for Anomaly and Misuse Detection. In: Proc. of the 8th USENIX Security Symposium, Washington D.C., USA (1999)
18. Han, S.-J., Cho, S.-B.: Evolutionary Neural Networks for Anomaly Detection Based on the Behaviour of a Program. IEEE Transactions on Systems, Man and Cybernetics (2006)
19. Bivens, A., et al.: Network-based intrusion detection using neural networks. In: Proc. of Intelligent Engineering Systems through Artificial Neural Networks, ANNIE 2002, St.Luis, MO, vol. 12, pp. 579–584. ASME press, New York (2002)
20. Ceci, M., Appice, A., Caruso, C., Malerba, D.: Discovering Emerging Patterns for Anomaly Detection in Network Connection Data. In: An, A., Matwin, S., Raś, Z.W., Ślęzak, D. (eds.) ISMIS 2008. LNCS (LNAI), vol. 4994, pp. 179–188. Springer, Heidelberg (2008)
21. Denning, D., Neumann, P.: Requirements and Model for IDES-A Real-Time Intrusion-Detection Expert System. SRI Project 6169, SRI International, Menlo Park, CA (1985)
22. Masum, S., Ye, E.M., Chen, Q., Noh, K.: Chi-square statistical profiling for anomaly detection. In: Proceedings of the 2000 IEEE Workshop on Information Assurance and Security (2000)
23. Ye, N., Chen, Q.: An anomaly detection technique based on a chi-square statistic for detecting intrusions into information systems. Qual. Reliab. Eng. Int. 17, 105–112 (2001)
24. Tarka, M.: Anomaly detection in SOA systems. Msc Thesis, Institute of Computer Science, Warsaw University of Technology (2011) (in polish)
25. The R Project for Statistical Computing, `http://gcc.gnu.org/` (access September 2011)
26. Munz, G., Li, S., Carle, G.: Traffic Anomaly Detection Using K-Means Clustering. Wilhelm Schickard Institute for Computer Science, University of Tuebingen (2007)
27. Guozhu, D., Jinyan, L.: Efficient Mining of Emerging Patterns: Discovering Trends and Differences. Wright State University, The University of Melbourne (2007)
28. Hanley, J.A.: Receiver operating characteristic (ROC) methodology: the state of the art. Crit Rev Diagn Imaging (1989)
29. Bluemke, I., Tarka, M.: Detection of anomalies in a SOA system by learning algorithms. In: Zamojski, W., Mazurkiewicz, J., Sugier, J., Walkowiak, T., Kacprzyk, J. (eds.) Complex Systems and Dependability. AISC, vol. 170, pp. 69–85. Springer, Heidelberg (2012)
30. Kohonen, T.: The self-organizing map. Proc. IEEE 78(9), 1464–1480 (1990)

Construction of Sequential Classifier Using Confusion Matrix

Robert Burduk and Pawel Trajdos

Department of Systems and Computer Networks, Wroclaw University of Technology,
Wybrzeze Wyspianskiego 27, 50-370 Wroclaw, Poland
robert.burduk@pwr.wroc.pl

Abstract. This paper presents the problem of building the decision scheme in the multistage pattern recognition task. This task can be presented using a decision tree. This decision tree is built in the learning phase of classification. This paper proposes a split criterion based on the analysis of the confusion matrix. Specifically, we propose the division associated with an incorrect classification. The obtained results were verified on the data sets form UCI Machine Learning Repository and one real-life data set of the computer-aided medical diagnosis.

Keywords: Multistage classifier, sequential classifier, confusion matrix.

1 Introduction

The classification task may be divided according to its complexity. There are two groups here: one-step and multistage approach. In the one-step approach there is no division into smaller tasks classification. However, a multistage (sequential) approach breaks up a complex decision into a collection of several simpler decisions [1–4]. Many algorithm build a tree structure in the learning process [5, 6]. In other approaches the decision tree structure is fixed before the learning process [7]. Generally, the synthesis of the multistage classifier is a complex problem. It involves a specification of the following components [4, 8]:

- design of a decision tree structure [9],
- selection of features used at each non-terminal node of the decision tree [10–13],
- the choice of decision rules for performing the classification [14].

In particular, this paper discuses a way to design a decision tree structure. The split criterion is based on the confusion matrix. The potential division of the node is associated with the analysis of misclassification in the learning process. In the experiment decision rules are chosen arbitrarily in the entire tree.

The content of the work is as follows. Section 2 introduces the idea of the hierarchical (sequential) classifier. In Section 3 we describe the proposed split criterion. In the next section we present the results of the experiments verified on data sets form UCI repository and one real-life data set of the computer-aided medical diagnosis. The last section concludes the paper.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 401–407, 2013.

2 Hierarchical Classifier

The hierarchical classifier contains a sequence of actions [15, 16]. These actions are simple classification tasks executed in the individual nodes of the decision tree. Some specific features are measured on every non-leaf node of the decision tree. At the first nonleaf node features x_0 are measured, at the second features x_1 are considered and so on. Every set of features comes from the whole vector of features. In every node of the decision tree the classification is executed according to the specific rule. The decisions $i_0, i_1, ..., i_N$ are the results of recognition in the suitable node of the tree. The design of a decision tree structure is based on split criterion.

In our task of classification the number of classes is equal to NC. The terminal nodes are labeled with the number of the classes from $M = 1, 2, ..., NC$, where M is the set of labels classes. The non-terminal nodes are labeled by numbers of 0, NC+1, NC+2 reserving 0 for the root-node. Let us introduce the notation for the received model of the multistage recognition [8]:

- $\overline{\mathcal{M}}$ – the set of internal (nonleaf) nodes,
- $\mathcal{M}_i$ – the set of class labels attainable from the i-th node ($i \in \overline{\mathcal{M}}$),
- $\mathcal{M}^i$ – the set of nodes of the immediate descendant node i ($i \in \overline{\mathcal{M}}$),
- m_i – the node of the direct predecessor of the i-th node ($i \neq 0$).

In each interior node the recognition algorithm is used. It maps observation subspace to the set of the immediate descendant nodes of the i-th node [17, 18]:

$$\Psi_i : X_i \to \mathcal{M}^i, \quad i \in \overline{\mathcal{M}} \; . \tag{1}$$

This approach minimizes the misclassification rate for the particular nodes of a tree. The decision rules at each node are mutually independent. In experiment the decision rules are chosen arbitrarily in the entire tree. Each of the classifiers used in the nodes of the tree takes a decision based on the full set of attributes available in the training set.

In our method of induction, the classification tree is a regular binary tree. This means that on each of the tree nodes there is a leaf or a node has two children.

Induction of the decision tree is performed by the top-down method. This means that it is initiated by the classifier located in the root of the tree. Using the proposed criterion the decision is made whether to continue the division. The process is repeated for the subsequent child nodes of the tree, until the state wherein the nodes in the tree can no longer be divided.

3 Split Criteria

The division of the internal node will be made on the basis of the multidimensional confusion matrix. Specifically, we propose the division associated with an incorrect classification. This division is binary, which means that the node that will meet the criterion of the split will have two child nodes. One of them represents a new internal decision tree node. The second one represents the label of a class that met the appropriate condition. The internal node of the decision tree is analysed in detail via the multidimensional confusion matrix. The columns of the confusion matrix correspond to the predicted labels (decisions made by the classifier in the internal node). The rows correspond to the true class labels. In this matrix the diagonal elements represent the overall performance of each label. The off-diagonal elements represent the errors related to each label.

Now we present the split criterion. For every class labels from internal node we create the $L \times L$ dimensional confusion matrix. Now we calculate the set of factors $W(k_l)$, where $l = 1, 2, ..., L$ is the number of class labels, according to the formula:

$$W(k_l) = \sum_{m=1, m \neq l}^{L} w_{l,m} + \sum_{m=1, m \neq l}^{L} w_{m,l}. \tag{2}$$

The example of the confusion matrix is presented in Tab. 1.

Table 1. The confusion matrix for the nonleaf node i

	estimated			
	k_1	k_2	$\dots$	k_L
k_1	$w_{1,1}$	$w_{1,2}$	$\dots$	$w_{1,L}$
true k_2	$w_{2,1}$	$w_{2,2}$	$\dots$	$w_{2,L}$
$\vdots$	$\vdots$	$\vdots$		$\vdots$
k_L	$w_{L,1}$	$w_{L,2}$	$\dots$	$w_{L,L}$

The division of the node occurs when

$$\left| \max_{l \in 1,2,\ldots,L} W(k_l) - \min_{l \in 1,2,\ldots,L} W(k_l) \right| > T, \tag{3}$$

where $T \in [0, 1]$ is a fixed threshold value. At the threshold one does not make a division of node. If we set this value at the beginning of the experiment, it indicates that the classification process is performed in the one-stage approach.

4 Experiments

In the experiential research 10 data sets were tested. Nine data sets come from UCI Machine Learning Repository [19]. The tenth comes from the Surgical Clinic Wroclaw Medical Academy and describes the acute abdominal pain diagnosis problem. A set of all the available features was used for all data sets, however, for the acute abdominal pain data set the selection of features has been made in accordance with the suggestions from another work on the topic [20, 21].

Table 2. Description of data sets selected for the experiments

Data set	example	attribute	class
Acute Abdominal Pain	476	31	8
Breast Tissue	106	10	6
Ecoli	336	7	8
Glass Identification	214	10	6
Irys	150	4	3
Lung Cancer	31	52	3
Seeds	210	7	3
Vertebral Column	310	6	3
Wine	178	13	3
Yeast	1484	8	10

The numbers of attributes, classes and available examples of the investigated data sets are presented in Tab. 2.

Tab. 3 presents the mean error for $5 - NN$ (5-nearest neighbor) classifier for the selected values of the parameter T. In Tab. 4 we presented the average ranks for all experiments. The average ranks are calculated on the basis of the Friedman test [22]. Each column in this table is attributed to one test. This means that the one-step approach $T = 1$ was compared in order to propose in the work the sequential approach. In Tab. 4 the lowest average ranks in each group are shown in bold.

Table 3. Avarage error for 5-NN classifier

Data set	T=0.02	T=0.04	T=0.02	T=0.4	T=0.5	T=1
Acute	0.161	0.161	0.163	0.165	0.168	0.163
Breast	0.428	0.419	0.438	0.489	0.457	0.412
Ecoli	0.128	0.131	0.124	0.131	0.131	0.131
Glass	0.336	0.344	0.333	0.327	0.34	0.343
Irys	0.03	0.035	0.03	0.03	0.035	0.035
Lung	0.472	0.525	0.537	0.458	0.487	0.557
Seeds	0.112	0.112	0.112	0.112	0.112	0.112
Vert.	0.184	0.181	0.178	0.178	0.181	0.184
Wine	0.302	0.303	0.297	0.322	0.325	0.319
Yeast	0.431	0.437	0.44	0.457	0.444	0.443

All classifiers from the group NN (Nearest Neighbor), with properly chosen values T, may improve the quality of classification. The value of this improvement, however, is not significant from the statistical point of view. For the post-hoc Bonferroni-Dunn [23] test the critical difference (CD) for the 27 values of the parameter T and 10 data sets is equal $CD = 11,7$. This CD is calculated at $\alpha = 0.05$. Although the differences of the average rank do not exceed CD, the obtained results can be considered promising because they are close to this value (in particular 5-NN).

Table 4. Average ranks from Friedman test

T	3-NN	5-NN	7-NN	9-NN	SVM
0	15,45	14,1	19	15,85	15,85
0.02	13.95	**7.8**	15.1	16.9	15.85
0.04	**9.95**	13.15	17	15.8	15.85
0.06	14.2	13.05	16.7	16.4	15.85
0.08	11.85	11.15	14	11.6	15.85
0.1	12.1	13.45	15.15	13.95	14.5
0.12	16.85	10.95	19.75	11.2	14.5
0.14	12.65	19.1	11.7	**7.85**	14.9
0.16	12.25	12	14.35	16.15	13.2
0.18	15.75	14.25	15.5	15.1	13.2
0.2	15.4	8.55	11.7	10.95	13.2
0.22	12.85	14.7	**7.45**	16.15	13.2
0.24	11.05	13.55	16.15	13.65	13.2
0.26	18.1	16.5	16.1	14.85	14.55
0.28	16	12.7	14.3	9.85	14.55
0.3	13.55	13.8	15.35	16.5	14.55
0.32	12	12.85	11.5	14.15	13.7
0.34	14	12.75	13.7	16.4	13.15
0.36	15.2	15.35	13.2	15.9	**13.15**
0.38	10.7	13.55	14.3	14.5	**13.15**
0.4	16.05	15.75	14.95	9.65	**13.15**
0.42	13.15	13.55	14.7	14.35	**13.15**
0.44	12.9	18	10.15	12.75	**13.15**
0.46	16.6	15.5	12.75	14.1	**13.15**
0.48	13.3	17.95	11.1	13.65	**13.15**
0.5	16.15	16.95	11.7	14.45	**13.15**
1	16	17	10.65	15.35	**13.15**

5 Conclusions

In the paper we propose a split criterion based on analizing the confusion matrix. Specifically, we propose the division associated with an incorrect classification. This criterion is used in the design of a decision tree structure in the multistage classifier. With a fulfilled criteria a binary split of the analyzed decision node is carried out. If we set $T = 1$, it indicates that the classification process is performed in the one-stages approach. Which means that it does not creat a decision tree.

Experiments done in the work show that we have obtain promising results. The proposed approach improves the quality of classification for $k-NN$ group of classifiers. In some cases, the difference of mean ranks obtained by Friedman test is close to the critical difference. In the future work we can use different division criteria in order to design a sequential classifier. For example, the separable linearization [24] or MacArthur's [25] overlapping niches model can be used in the split criterium.

Acknowledgments. The work was supported in part by the statutory funds of the Department of Systems and Computer Networks, Wroclaw University of Technology and by the by The Polish National Science Centre under the grant N N519 650440 which is being realized in years 2011–2014.

References

1. Kołakowska, A., Malina, W.: Fisher Sequential Classifiers. IEEE Transaction on Systems, Man, and Cybernecics – Part B Cybernecics 35(5), 988–998 (2005)
2. Mui, J., Fu, K.S.: Automated classification of nucleated blood cells using a binary tree classifier. IEEE Trans. Pattern Anal. Mach. Intell. PAMI-2, 429–443 (1980)
3. Podolak, I.T.: Hierarchical classifier with overlapping class groups. Expert Syst. Appl. 34(1), 673–682 (2008)
4. Safavian, S.R., Landgrebe, D.: A survey of decision tree classifier methodology. IEEE Trans. Systems, Man Cyber. 21(3), 660–674 (1991)
5. Penar, W., Woźniak, M.: Experiments on classifiers obtained via decision tree induction methods with different attribute acquisition cost limit. Advances in Soft Computing 45, 371–377 (2007)
6. Quinlan, J.R.: Induction on Decision Tree. Machine Learning 1, 81–106 (1986)
7. Manwani, N., Sastry, P.S.: Geometric decision tree. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 42(1), 181–192 (2012)
8. Kurzyński, M.: Decision Rules for a Hierarchical Classifier. Pat. Rec. Let. 1, 305–310 (1983)
9. Woźniak, M.: A hybrid decision tree training method using data streams. Knowledge and Information Systems 29(2), 335–347 (2010)
10. Choraś, M.: Image feature extraction methods for ear biometrics–A survey. In: 6th International Conference on Computer Information Systems and Industrial Management Applications, CISIM 2007, pp. 261–265. IEEE (2007)
11. Choraś, R.S.: Content-based retrieval using color, texture, and shape information. In: Progress in Pattern Recognition, Speech and Image Analysis, pp. 619–626 (2003)
12. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. The Journal of Machine Learning Research 3, 1157–1182 (2003)
13. Rejer, I.: Genetic Algorithms in EEG Feature Selection for the Classification of Movements of the Left and Right Hand. In: Burduk, R., Jackowski, K., Kurzynski, M., Wozniak, M., Zolnierek, A. (eds.) CORES 2013. AISC, vol. 226, pp. 579–589. Springer, Heidelberg (2013)
14. Burduk, R., Zmyślony, M.: Decomposition of classification task with selection of classifiers on the medical diagnosis example. In: Corchado, E., Snášel, V., Abraham, A., Woźniak, M., Graña, M., Cho, S.-B. (eds.) HAIS 2012, Part II. LNCS, vol. 7209, pp. 569–577. Springer, Heidelberg (2012)
15. Burduk, R.: Classification error in Bayes multistage recognition task with fuzzy observations. Pattern Analysis and Applications 13(1), 85–91 (2010)
16. Kurzyński, M.: On the Multistage Bayes Classifier. Pattern Recognition 21, 355–365 (1988)
17. Berger, J.: Statistical Decision Theory and Bayesian Analysis. Springer, New York (1993)
18. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. John Wiley and Sons (2000)

19. Frank, A., Asuncion, A.: UCI machine learning repository (2010)
20. Burduk, R., Woźniak, M.: Different decision tree induction strategies for a medical decision problem. Central European Journal of Medicine 7(2), 183–193 (2010)
21. Kurzyński, M.: Diagnosis of acute abdominal pain using three-stage classifier. Computers in Biology and Medicine 17(1), 19–27 (1987)
22. Trawiński, B., Smetek, M., Telec, Z., Lasota, T.: Nonparametric statistical analysis for multiple comparison of machine learning regression algorithms. International Journal of Applied Mathematics and Computer Science 22(4), 867–881 (2012)
23. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. The Journal of Machine Learning Research 7, 1–30 (2006)
24. Bobrowski, L., Topczewska, M.: Separable Linearization of Learning Sets by Ranked Layer of Radial Binary Classifiers. In: Burduk, R., Jackowski, K., Kurzynski, M., Wozniak, M., Zolnierek, A. (eds.) CORES 2013. AISC, vol. 226, pp. 131–140. Springer, Heidelberg (2013)
25. MacArthur, R.: On the relative abundance of bird species. Proc. Natl. Acad. Sci. USA 43, 293–295 (1957)

Growing Neural Gas – A Parallel Approach

Lukáš Vojáček[1] and Jiří Dvorský[2]

[1] IT4Innovations Centre of Excellence
Ostrava, Czech Republic
lukas.vojacek@vsb.cz

[2] Department of Computer Science,
VŠB – Technical University of Ostrava, 17. listopadu 15,
708 33 Ostrava, Czech Republic
jiri.dvorsky@vsb.cz

Abstract. The paper deals with the high dimensional data clustering problem. One possible way to cluster this kind of data is based on Artificial Neural Networks (ANN) such as SOM or Growing Neural Gas (GNG). The learning phase of the ANN, which is time-consuming especially for large high-dimensional datasets, is the main drawback of this approach to data clustering. The parallel modification, Growing Neural Gas, and its implementation on the HPC cluster is presented in the paper. Some experimental results are also presented.

Keywords: growing neural gas, high-dimensional dataset, high performance computing.

1 Introduction

Recently, the issue of high-dimensional data clustering has arisen together with the development of information and communication technologies which support growing opportunities to process large data collections. High dimensional data collections are commonly available in areas like medicine, biology, information retrieval, web analysis, social network analysis, image processing, financial transaction analysis and many others.

Two main challenges should be solved to process high-dimensional data collections. One of the problems is the fast growth of computational complexity with respect to growing data dimensionality. The second one is specific similarity mea-surement in a high-dimensional space. As presented in [1], Beyer et al. for any point in a high-dimensional space the expected distance, computed by Euclidean measure to the closest and to the farthest point, shrinks with growing dimensionality. These two reasons reduce the effectiveness of clustering algorithms on the above-mentioned high-dimensional data collections in many real applications.

The authors propose an effective data clustering algorithm which is based on *Growing Neural Gas* (GNG) [7]. The computational complexity is resolved by the parallel implementation of GNG. Some technical problems have to be resolved in

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 408–419, 2013.

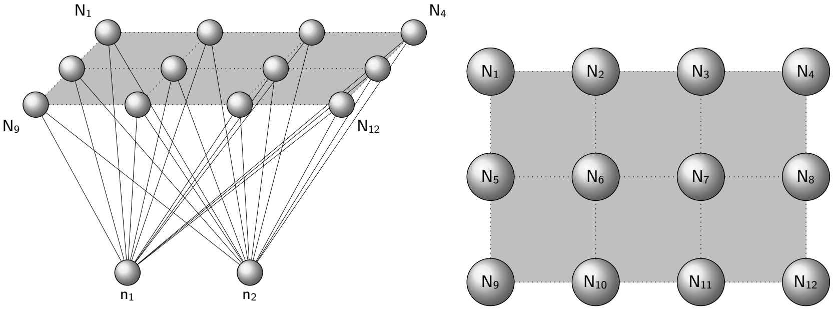
order to effectively train such kind of neural network using an *High Performance Computing* (HPC) cluster with MPI. The traditional serial approach to training GNG is also considered in the paper. The serial learning GNG algorithm is used for benchmarking the parallel version of GNG. In other words, parallel GNG has to produce the same network and should be an order of magnitude faster in the ideal case.

2 Artificial Neural Networks

2.1 Self Organizing Maps

Self Organizing Maps (SOM), also known as Kohonen maps, were proposed by Teuvo Kohonen in 1982 [4]. SOM is a kind of artificial neural network that is trained by unsupervised learning. Using SOM, the input space of training samples can be represented in a lower-dimensional (often two-dimensional) space [5], called *map*. SSuch model is efficient in structure visualization due to its feature of topological preservation using a neighbourhood function.

SOM consists of two layers of neurons (see Fig. 1): an *input layer* that receives and transmits the input information and an *output layer*, the map that represents the output characteristics. The output layer is commonly organized as a two-dimensional rectangular grid of nodes, where each node corresponds to one neuron. Both layers are feed-forward connected. Each neuron in the input layer is connected to each neuron in the output layer. A real number, or weight, is assigned to each of these connections.



(a) Global view on SOM structure (b) SOM output layer

Fig. 1. Basic Schema of SOM

2.2 Growing Neural Gas

The principle of this neural network is an undirected graph which need not be continuous. Generally, there are no restrictions on the topology. The graph is generated and continuously updated by competitive Hebbian Learning[6,9].

According to the pre-set conditions, new neurons are automatically added and connections between neurons are subject to time and can be removed. GNG can be used for vector quantization by finding the code-vectors in clusters [3], biologically influenced [10], image compression, disease diagnosis.

GNG works by modifying the graph, where the operations are the addition and removal of neurons and edges between neurons. An example of the operation is shown in Figure 2

To understand the functioning of GNG, it is necessary to define the algorithm. The algorithm described by Algorithm 1 is based on the original algorithm [2] [3], but it is modified for better continuity in the SOM algorithm. The description of the algorithm has been divided for convenience into two parts. In the first part of Algorithm 1 the overall functionality is described. The second part of Algorithm 2 describes one iteration, which means the descriptions of variables without dependence on time.

Remark The notation used in the paper is briefly listed in Table 1.

Algorithm 1. Growing Neural Gas algorithm

```
1. Initialization of network. Two neurons $\mathsf{N}_1$ and $\mathsf{N}_2$ are created, $\mathsf{E} = \{\mathsf{e}_{12}\}$. Weight
   vectors $w_1(t)$ and $w_2(t)$ are initialized to random values $w_{kj}(t) \in [0, 1]$.
2. Select arbitrary unused input data vector.
3. Perform the one learning iteration according to the algorithm described in Algorithm 2.
4. Reduce error value $e_i$ for all neurons $\mathsf{N}_i$ using factor $\beta$.
5. Returns to step 2, until all input data vector have been used.
6. If $t < T$ return to step 2.
```

3 Parallelization

Parallelization focuses on the equally distribution of neurons, where new neurons are allocated to the process with the lowest number of neurons. The advantage of this distribution is constant workload processes. The disadvantage is increase communication between processes.

After analysing the GNG learning algorithm we identified the one most processor time-consuming area. This part was selected as a candidate for the possible parallelization. The selected area is:

Finding BMU – this part of GNG learning can be significantly accelerated by dividing the GNG output layer into smaller pieces. Each piece is then assigned to an individual computation process. The calculation of the Euclidean distance among the individual input vector and all the weight vectors to find BMU in a given part of the GNG output layer is the crucial point of this part of GNG learning. Each process finds its own, partial, BMU in its part of the GNG output layer. Each partial BMU is then compared with

Table 1. Notation used in the paper

Symbol	Description
M	Number of input vectors
n	Dimension of input vectors, number of input neurons, dimension of weight vectors in GNG output layer neurons
N	Current number of neurons in GNG output layer
N_{max}	Maximum allowed number of neurons in GNG output layer
n_i	i-th input neuron, $i = 1, 2, \ldots, n$
N_i	i-th output neuron, $i = 1, 2, \ldots, N$
e_{ij}	edge between neurons N_i and N_j for some $i, j = 1, \ldots, N$, where $i \neq j$.
E	set of all edges in GNG
G	undirected graph describing topology of GNG, $G(\{\mathsf{N}_1, \ldots, \mathsf{N}_N\}, \mathsf{E})$
T	Number of epochs
t	Current epoch, $t = 1, 2, \ldots, T$
X	Set of input vectors, $X \subset \mathbb{R}^n$
$\boldsymbol{x}(t)$	Current input vector in epoch t, arbitrarily selected vector from set X $\boldsymbol{x}(t) \in X$, $\boldsymbol{x}(t) = (x_1, x_2, \ldots, x_n)$
$\boldsymbol{w_k}(t)$	Weight vector of neuron N_k, $k = 1, 2, \ldots, N$ $\boldsymbol{w_k}(t) \in \mathbb{R}^n$, $\boldsymbol{w_k}(t) = (w_{1k}, w_{2k}, \ldots, w_{nk})$
N_{c_1}	The first Best Matching Unit (BMU_1), winner of learning competition
N_{c_2}	The second Best Matching Unit (BMU_2), the second best matching neuron in learning competition
$\boldsymbol{w}_{c_1}(t)$	Weight vector of BMU_1
$\boldsymbol{w}_{c_1}(t)$	Weight vector of BMU_2
l_{c_1}	Learning factor of BMU_1
l_{nc_1}	Learning factor of BMU_1 neighbours
e_i	Local error of output neuron N_i, $i = 1, 2, \ldots, N$
α	Error e_i reduction factor
β	Neuron error reduction factor
γ	Interval of input patterns to add a new neuron
a_{max}	Maximum edges age
a_{ij}	Age of edge e_{ij}

Algorithm 2. One iteration of the Growing Neural Gas algorithm

1. Find neurons BMUs neurons N_{c_1} and N_{c_2}.
2. Update the local error e_{c_1} of neuron N_{c_1}

$$e_{c_1} = e_{c_1} + \|\boldsymbol{w}_{c_1} - \boldsymbol{x}\|^2 \tag{1}$$

3. Update the weight vector $\boldsymbol{w}_{c_1}$ of neuron N_{c_1}

$$\boldsymbol{w}_{c_1} = \boldsymbol{w}_{c_1} + \boldsymbol{l}_{c_1}(\boldsymbol{x} - \boldsymbol{w}_{c_1}) \tag{2}$$

4. For all neurons N_k where exists edge e_{c_1k} (N_{c_1} neighbourhood)
   (a) Update the weights $\boldsymbol{w}_k$ using l_{nc_1} learning factor

$$\boldsymbol{w}_k = \boldsymbol{w}_k + l_{nc_1}(\boldsymbol{x} - \boldsymbol{w}_{\boldsymbol{k}}) \tag{3}$$

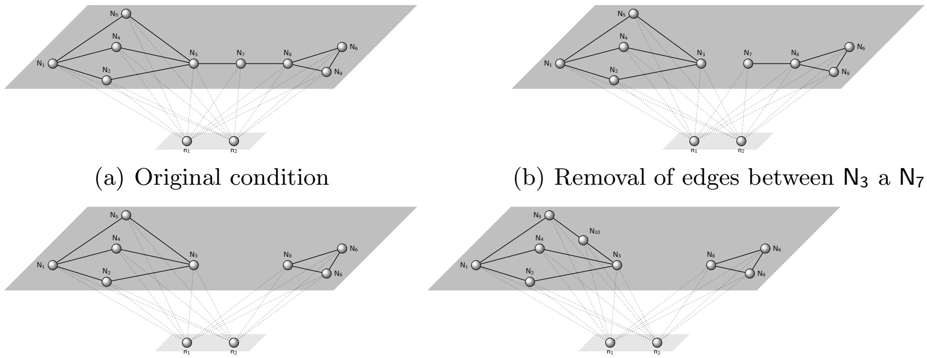
   (b) Increase age a_{kc_1} of edge e_{c_1k}

$$a_{kc_1} = a_{kc_1} + 1 \tag{4}$$

5. If there is no edge between neurons N_{c_1} and N_{c_2}, then create such edge. If the edge exists, the age is set to 0.
6. If any edge has reached the age of a_{max}, it is removed.
7. If there is a neuron without connection to any edge, the neuron is then removed.
8. If the number of processed input vectors in the current iteration has reached the whole multiple of the value γ and the maximum allowed number of output neurons is not reached, add a new neuron N_{N+1}. The location and error of the new neuron is determined by the following rules:
   (a) Found neuron N_b(NBE) which has the biggest error e_b.
   (b) Found neuron N_c(NSE) among neighbours of neuron N_b and has the biggest error e_c among these neighbours.
   (c) Create a new neuron N_{N+1} and the value of w_n is set as:

$$\boldsymbol{w}_{N+1} = \frac{1}{2}(\boldsymbol{w}_b + \boldsymbol{w}_c) \tag{5}$$

   (d) Creating edges between neurons N_b and N_{N+1}, and also between neurons N_c and N_{N+1}.
   (e) Removed edge between neurons N_b and N_c.
   (f) Reduction of error value in neurons N_b and N_c using the multiplying factor α. Error for neuron N_{N+1} is equal to the new error of neuron N_b.



(a) Original condition

(b) Removal of edges between N_3 a N_7

(c) Removal of edges between N_7 a N_8, remove neuron N_7

(d) Added neuron N_{10} between N_5 a N_3

Fig. 2. GNG examples

other BMUs obtained by other processes. Information about the BMU of the whole network is then transmitted to all the processes to perform the updates of the BMU neighbourhood.

A detailed description of our approach to the parallelization process is described in Fig. 3.

The parallelization of GNG learning was performed on an HPC cluster, using *Message Passing Interface* (MPI) technology. MPI technology is based on effective communication between processes. That means that one application can run on many cores. The application uses MPI processes which run on individual cores. The processes are able to send and receive messages and data, communicate etc. Detailed information about HPC and MPI technology is provided, for example, in [8][1].

3.1 Description of Proposed Approach

This subsection includes a detailed description of our approach. Initially, each process reads the training input vectors and on the first process are create two neurons. During the calculation, new neurons are equally distributed on the processes. We do not store the complete GNG graph, we only store parts of it in the appropriate computational nodes.

This approach results in the following significant advantage: neurons are equally distributed among the processes corresponding to the used cores, contrary to the sequential approach, where the whole graph is allocated to only one core (and where there may not be enough memory). For a more precise

[1] A specification of MPI is available on the web: `http://www.mpi-forum.org/` The functions used in the experiment with a link to MPI.Net are available on the web: `http://osl.iu.edu/research/mpi.net`

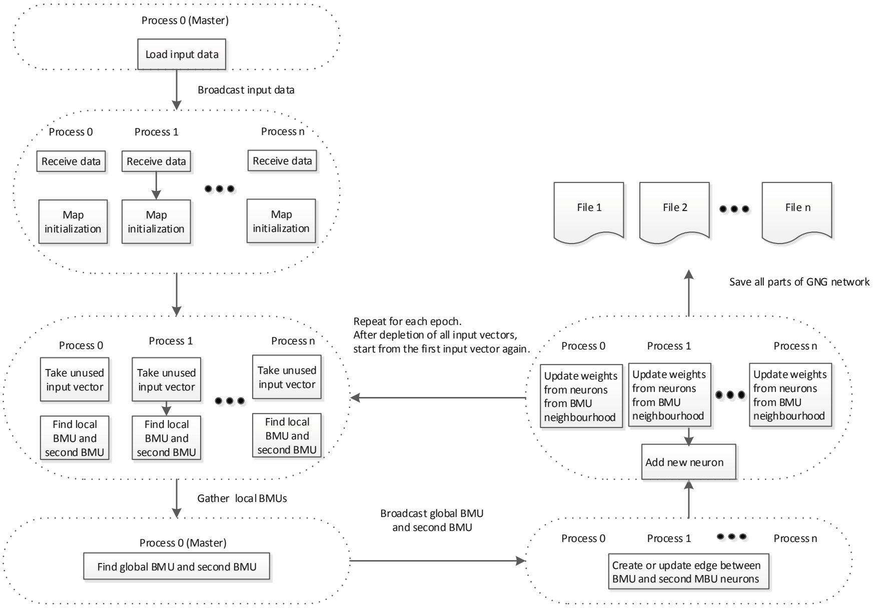

Process 0 (Master)
Load input data
Broadcast input data
Process 0
Process 1
Process n
Receive data
Receive data
Receive data
Map initialization
Map initialization
Map initialization
File 1
File 2
File n
Save all parts of GNG network
Repeat for each epoch.
After depletion of all input vectors,
start from the first input vector again.
Process 0
Process 1
Process n
Take unused input vector
Take unused input vector
Take unused input vector
Find local BMU and second BMU
Find local BMU and second BMU
Find local BMU and second BMU
Process 0
Process 1
Process n
Update weights from neurons from BMU neighbourhood
Update weights from neurons from BMU neighbourhood
Update weights from neurons from BMU neighbourhood
Add new neuron
Gather local BMUs
Broadcast global BMU and second BMU
Process 0 (Master)
Find global BMU and second BMU
Process 0
Process 1
Process n
Create or update edge between BMU and second MBU neurons

Fig. 3. Parallel Algorithm

illustration: consider having three computation servers, each with 16GB of memory. If we use the sequential GNG version, we can only use 16GB of memory on one server. But in the case of the parallel SOM version, we can use all the memory, all 48GB. (The graph can be up to three times larger.)

In the main computational phase, one BMU and a second BMU are founded for each input vector. Thus, each processor needs to compute its local BMU and second BMU within its neurons, after which, each local BMU and second BMU (and their position in the network) are shifted onto one process using the MPI function *GatherFlattened* to determine the global BMU and second BMU. It is possible to use another MPI functions as well, which can provide this selection at one time, but after testing we have found that the experiments took much more time than our presented approach. A global winning BMU and second BMU are then distributed using the MPI function *Broadcast* on all the processes. Now if there is a edge between the first BMU and the second BMU then age is set to zero otherwise creates edge between this two neurons. Next, the neighbourhood of the BMU in each process is known and, consequently, the weights of the neurons matching are actualized. If the condition is met for adding a new neuron, the process with the lowest number of neurons add a new neuron. A detailed description of adding a new neuron can be found in Fig. 4. This procedure is repeated until all the input training vectors are exhausted (until we have finished one epoch).

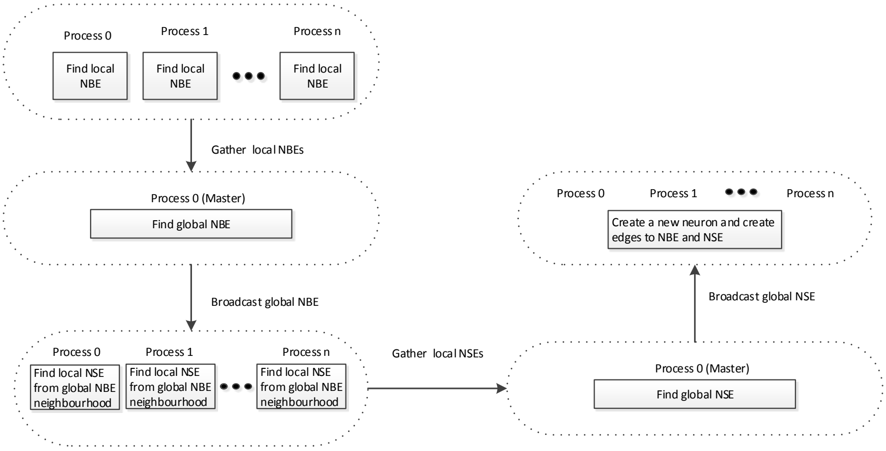


Fig. 4. Parallel Algorithm for adding a neuron

4 Experiments

4.1 Experimental Datasets and Hardware

Two datasets were used in the experiments. The first dataset was commonly used in Information Retrieval – *Medlars*. The second one was the test data for the elementary benchmark for clustering algorithms[11].

Medlars Dataset. The Medlars dataset consisted of 1,033 English abstracts from a medical science[2]. The 8,567 distinct terms were extracted from the Medlars dataset. Each term represents a potential dimension in the input vector space. The term's level of significance (weight) in a particular document represents a value of the component of the input vector. Finally, the input vector space has a dimension of 8,707, and 1,033 input vectors were extracted from the dataset.

Clustering Dataset. Three training data collections called TwoDiamonds, Lsun and Hepta from the Fundamental Clustering Problems Suite (FCPS) are used. A short description of the selected dataset used in our experiments is given in Table 2.

Experimental Hardware All the experiments were performed on a Windows HPC server 2008 with 6 computing nodes, where each node had 8 processors with 12 GB of memory. The processors in nodes were Intel Xeon 2.27GHz.

[2] The collection can be downloaded from `ftp://ftp.cs.cornell.edu/pub/smart`. The total size of the dataset is approximately 1.03 MB.

Table 2. Fundamental Clustering Problems Suite – selected datasets

Name	Cases	#Vars	#Clusters	Main Clustering Problem
Target	770	2	6	outlying clusters
Lsun	400	2	3	different variances in clusters
TwoDiamonds	800	2	2	touching clusters

The topology with the connection between the head node and computing nodes can be found on the web[3] (topology number four). The Enterprise and Private Networks link speed was 1Gbps, the Application link speed was 20Gbps.

4.2 First Part of the Experiment

The first part of the experiment was oriented towards a comparison of the standard GNG algorithm and parallel approach to this GNG learning algorithm. The Medlars dataset was used for the experiment. A parallel version of the learning algorithm was run using 2, 8, 16, 24 and 32 MPI processes. The records with an asterisk (*) represents the results for only one process i.e. this is the original serial learning algorithm and there is no network communication.

GNG parameters are the same for all experiments and are as follows $\gamma = 200$, $e_w = 0.05$, $e_n = 0.006$, $\alpha = 0.5$, $\beta = 0.0005$, $a_{max} = 88$, M = 1000, $\delta = 100$. The achieved computing time is presented in Table 3.

Table 3. Computing Time with Respect to Number of Cores, Standard GNG Algorithm, Dataset Medlars

Cores	Computing Time [hh:mm:ss]
1*	00:35:41
2	00:17:50
8	00:04:47
12	00:03:56
16	00:03:09
24	00:02:45
32	00:02:32

As we can see from Table 3, the computing time depends on the number of used cores as well. With a growing number of processors, the computation effectiveness increases, and the computational time is sufficiently reduced.

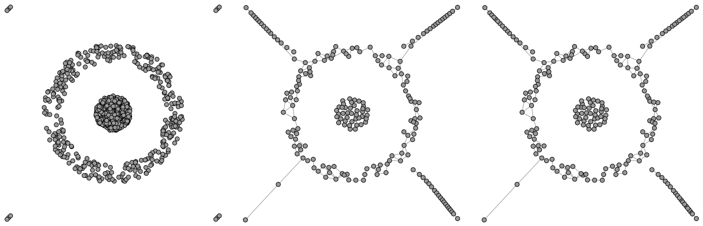
4.3 Second Part of the Experiment

The second part of the experiments was oriented towards comparing the results obtained by the parallel and standard GNG algorithm. The Clustering dataset

[3] http://technet.microsoft.com/en-us/library/cc719008(v=ws.10).aspx

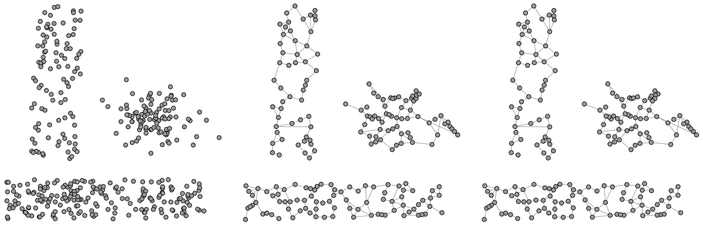
was used for the experiment. The parallel version of the learning algorithm was run using 16 MPI processes.

GNG parameters are similar to the previous experiment. There are two changes M = 500 and $\delta = 25$.



(a) Input data (b) Standard GNG (c) Parallel GNG using 16 cores

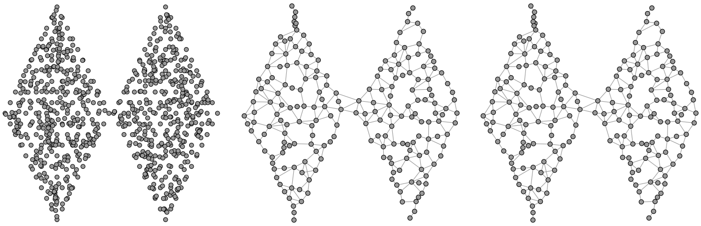
Fig. 5. Results of dataset *Target*



(a) Input data (b) Standard GNG (c) Parallel GNG using 16 cores

Fig. 6. Results of dataset *Lsun*

In Figures 5(a), 6(a) and 7(a) there are a layout view input data, which are used for training GNG. Outputs of standard GNG algorithm are in Figures 5(b), 6(b) and 7(b), which are the same as in the parallel version. i.e. both versions produce the same network.



(a) Input data (b) Standard GNG (c) Parallel GNG using 16 cores

Fig. 7. Results of dataset *TwoDiamonds*

5 Conclusion

In this paper the parallel implementation of the GNG neural network algorithm is presented. The achieved speed-up was very good and the results from the standard and parallel version of GNG are same. So we can say that the operation of the parallel version is correct. However, the effectiveness of a parallel solution is dependent on the division of the output layer. An improper division may cause the communication between processes to be very time consuming.

In future work we intend to focus on the sparse date, use combinations of neural networks for improved result and improved acceleration.

Acknowledgement. This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and by the Development of human resources in research and development of latest soft computing methods and their application in practice project, reg. no. CZ.1.07/2.3.00/20.0072 funded by Operational Programme Education for Competitiveness, co-financed by ESF and state budget of the Czech Republic.

References

1. Beyer, K., Goldstein, J., Ramakrishnan, R., Shaft, U.: When is nearest neighbor meaningful? In: Beeri, C., Bruneman, P. (eds.) ICDT 1999. LNCS, vol. 1540, pp. 217–235. Springer, Heidelberg (1998)
2. Fritzke, B.: A growing neural gas network learns topologies. In: Advances in Neural Information Processing Systems 7, pp. 625–632. MIT Press (1995)
3. Holmström, J.: Growing Neural Gas Experiments with GNG, GNG with Utility and Supervised GNG. Master's thesis, Uppsala University (August 30, 2002)
4. Kohonen, T.: Self-Organization and Associative Memory, 3rd edn. Springer Series in Information Sciences, vol. 8. Springer, Heidelberg (1984)
5. Kohonen, T.: Self Organizing Maps, 3rd edn. Springer (2001)

6. Martinetz, T.: Competitive hebbian learning rule forms perfectly topology preserving maps. In: Gielen, S., Kappen, B. (eds.) ICANN 1993, pp. 427–434. Springer, London (1993), http://dx.doi.org/10.1007/978-1-4471-2063-6_104
7. Martinetz, T., Schulten, K.: A "neural-gas" network learns topologies. Artificial Neural Networks 1, 397–402 (1991), http://web.cs.swarthmore.edu/~meeden/DevelopmentalRobotics/fritzke95.pdf
8. Pacheco, P.: Parallel Programming with MPI, 1st edn. Morgan Kaufmann (1996)
9. Prudent, Y., Ennaji, A.: An incremental growing neural gas learns topologies. In: Proceedings of the 2005 IEEE International Joint Conference on Neural Networks, IJCNN 2005, vol. 2, pp. 1211–1216 (July 4-August 2005)
10. Sledge, I., Keller, J.: Growing neural gas for temporal clustering. In: 19th International Conference on Pattern Recognition, ICPR 2008, pp. 1–4 (2008)
11. Ultsch, A.: Clustering with SOM: U*C. In: Proc. Workshop on Self-Organizing Maps, Paris, France, pp. 75–82 (2005)

Modified Moment Method Estimator for the Shape Parameter of Generalized Gaussian Distribution for a Small Sample Size

Robert Krupiński

West-Pomeranian University of Technology in Szczecin,
Chair of Signal Processing and Multimedia Engineering,
ul. 26-Kwietnia 10, 71-126 Szczecin, Poland
rkrupinski@wp.pl

Abstract. The moment method (MM) estimator for the shape parameter of generalized Gaussian distribution (GGD) assume asymptotic case when there is available infinite number of observations, but with the smaller sample size the variance of the estimator increases and the moment method equation may not converge to a real solution for some sample sets. The higher order moments can be expanded into series in the moment method equation leading to a drop in the relative mean square error (RMSE) and assuring a solution for a smaller sample size comparing to the moment method without modification.

Keywords: estimation, generalized Gaussian distribution, moment method.

1 Introduction

The moment method [1] is very popular method used for the estimation of the power shape parameter of GGD. The most popular approach is when the two moments are set to the values $m_1 = 1$ and $m_2 = 2$ (the Mallat's method), but the higher order moments are also used [2]. Most of the already known estimation methods for the shape parameter of GGD assume that the sample size is large, but in the real case, a set of limited size is only available. A very high value of variance for a very small sample size makes the estimation methods very inaccurate.

The computation of GGD parameters on small sample size is different from on larger ones in a maximum likelihood (ML, [3]) framework as was shown in [4]. The ML method has a lower variance for a small sample size than the MM method [5], but it is complex and time consuming. The complexity of ML can be reduced with the One Moment (OM) method [6]. The comparison of ML and the Mallat's method for the estimation of GGD shape parameter in the range $0.3 - 3$ is presented in [7], where the latter one is not accurate for the whole range of the shape parameter.

The presented approach in this article will not overcome other estimation methods for a small sample size, as it is fully based on the MM framework, but

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 420–429, 2013.

it will enhance the convergence of the MM method for the higher order moments and a small sample size.

In Section 2 the higher order moments are expanded into series in the moment method equation for a small sample size. In Section 3 the numerical results for the integer and real higher order moments are presented.

2 Material and Methods

The probability density function (PDF) of the continuous random variable of GGD [8] takes the form

$$f(x) = \frac{\lambda(p,\sigma) \cdot p}{2 \cdot \Gamma(\frac{1}{p})} e^{-[\lambda(p,\sigma) \cdot |x-\mu|]^p}, \tag{1}$$

where $\Gamma(z) = \int_0^\infty t^{z-1} e^{-t} dt, z > 0$, is the Gamma function [9], p denotes the shape parameter, μ is the location parameter and λ relates to the variance of the distribution and can be calculated from

$$\lambda(p,\sigma) = \frac{1}{\sigma} \left[\frac{\Gamma(\frac{3}{p})}{\Gamma(\frac{1}{p})} \right]^{\frac{1}{2}}, \tag{2}$$

where σ denotes the standard deviation. The special case of GGD can be observed when the shape parameter equals $p = 1$ and $p = 2$, which corresponds to Laplacian and Gaussian distributions respectively. For $p \to 0$, $f(x)$ approaches an impulse function, and for $p \to \infty$, $f(x)$ approaches a uniform distribution. For $\mu = 0$, the probability density function is centralized around zero.

From the definition of absolute moment $E[|X|^m] = \int_{-\infty}^{\infty} |x|^m \cdot f(x)dx$, for GGD the following is obtained [5]

$$E_m = \frac{\Gamma\left(\frac{m+1}{p}\right)}{\lambda^m \cdot \Gamma\left(\frac{1}{p}\right)}, \tag{3}$$

where m can be a real number.

The E_m estimated value of the moment can be acquired from the equation

$$\hat{E}_m = \frac{1}{N} \cdot \sum_{i=1}^{N} |x_i|^m \tag{4}$$

and where N denotes the number of observed variables, and $\{x_1, x_2, \ldots, x_N\}$ is the collection of N i.i.d zero-mean random variables.

Eq. (3) for two different moment values m_1 and m_2 and eliminating λ leads to:

$$\frac{\Gamma\left(\frac{m_2+1}{p}\right) \Gamma\left(\frac{1}{p}\right)^{\frac{m_2}{m_1}-1}}{\Gamma\left(\frac{m_1+1}{p}\right)^{\frac{m_2}{m_1}}} = \frac{E_{m_2}}{(E_{m_1})^{\frac{m_2}{m_1}}}, \tag{5}$$

which is the method for the estimation of the shape parameter based on two moments.

The moments (Eq. (4)) in the rhs of Eq. (5) can be expanded into series. Each component $|x_i|^m$ in the sum (Eq. (4)) can be represented in the form $(1 \pm y)^m$, which can be expanded into series for $m > 0$ and $|y| \leq 1$. For $x_i \in < 0, 1 >$, it can be written (by Binomial Theorem)

$$\begin{aligned} &(x_i)^m = (1 + (x_i - 1))^m = \\ &1 + m \cdot (x_i - 1) + \tfrac{m\cdot(m-1)}{2!} \cdot (x_i - 1)^2 + . \\ &\tfrac{m\cdot(m-1)\cdot(m-2)}{3!} \cdot (x_i - 1)^3 + \ldots \end{aligned} \tag{6}$$

For $x_i \in < -1, 0)$, it takes the form

$$\begin{aligned} &(-x_i)^m = (1 - (x_i + 1))^m = \\ &1 - m \cdot (x_i + 1) + \tfrac{m\cdot(m-1)}{2!} \cdot (x_i + 1)^2 - . \\ &\tfrac{m\cdot(m-1)\cdot(m-2)}{3!} \cdot (x_i + 1)^3 + \ldots \end{aligned} \tag{7}$$

For $x_i > 1$, it can be written

$$\begin{aligned} &(x_i)^m = |\lfloor x_i \rfloor|^m \cdot (1 + \left(\tfrac{x_i}{|\lfloor x_i \rfloor|} - 1\right))^m = \\ &|\lfloor x_i \rfloor|^m \cdot \left\{ 1 + m \cdot \left(\tfrac{x_i}{|\lfloor x_i \rfloor|} - 1\right) + \tfrac{m\cdot(m-1)}{2!} \cdot \left(\tfrac{x_i}{|\lfloor x_i \rfloor|} - 1\right)^2 + . \right. \\ &\left. \tfrac{m\cdot(m-1)\cdot(m-2)}{3!} \cdot \left(\tfrac{x_i}{|\lfloor x_i \rfloor|} - 1\right)^3 + \ldots \right\} \end{aligned} \tag{8}$$

The operator $\lfloor x \rfloor$ rounds the elements of x toward zero, resulting in an integer. For $x_i < -1$, it can be written

$$\begin{aligned} &(-x_i)^m = |\lfloor x_i \rfloor|^m \cdot (1 - \left(\tfrac{x_i}{|\lfloor x_i \rfloor|} + 1\right))^m = \\ &|\lfloor x_i \rfloor|^m \cdot \left\{ 1 - m \cdot \left(\tfrac{x_i}{|\lfloor x_i \rfloor|} + 1\right) + \tfrac{m\cdot(m-1)}{2!} \cdot \left(\tfrac{x_i}{|\lfloor x_i \rfloor|} + 1\right)^2 - . \right. \\ &\left. \tfrac{m\cdot(m-1)\cdot(m-2)}{3!} \cdot \left(\tfrac{x_i}{|\lfloor x_i \rfloor|} + 1\right)^3 + \ldots \right\} \end{aligned} \tag{9}$$

The equations (6)–(9) are used in the approximation of the moment (Eq. (4)) and used in Eq. (5) for the solution of p. Taking into account only a limited number of series components, the influence of outliers on the MM equation can be alleviated for a small sample size and the higher order moments and consequently reduce RMSE and improve the convergence.

3 Calculation

The equations from the article were validated with the GGD generator with fixed variance to unity and varying shape parameter in the range $p \in < 0.3, 6 >$.

The relative mean square error was applied for the comparison of the estimators output. RMSE was calculated from the equation

$$RMSE = \frac{1}{M}\sum_{i=1}^{M}\frac{(\hat{p}-p)^2}{p^2}, \tag{10}$$

where $\hat{p}$ is a value estimated by the model and p is a real value of a shape parameter. M denotes the number of repetitions.

A small sample size may lead to difficulties with the root finding of the MM method (Eq. (5)), therefore, the stop condition was set to $tolX = 1e-4$ and $tolY = 1e-4$. When at least one solution is not available for the specified conditions, then the plots have the missing points for such a case.

Fig. 1 depicts RMSE for the MM method of the estimation of the shape parameter p of GGD with a varying shape value p and a selected fixed sample size $N = 10^4$. Each iteration was repeated $M = 10^3$. For a relatively large sample size, RMSE is not constant for all shape parameters of GGD for the MM method. The RMSE value depends on the selected moments m_1 and m_2 and is relatively small. As an example, the integer moments were selected $m_1 = 4$, $m_2 = 8$ and the real ones: $m_1 = 3.5$, $m_2 = 5.5$. The higher order moments exhibit the better RMSE performance of MM for the higher values of p.

In the case of a small sample size, the increase of RMSE as well as the reduction of a convergence range are observed (Fig. 2). The higher order moments

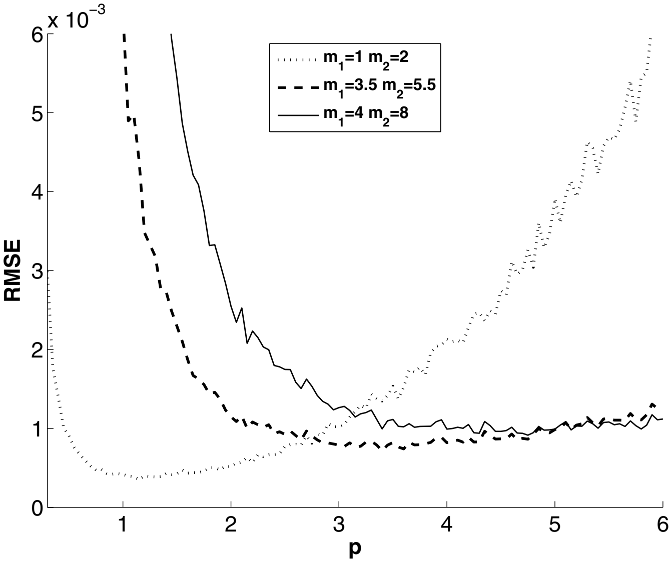


Fig. 1. Comparison of RMSE for the MM method of the estimation of the shape parameter p of GGD with a varying shape value p and a selected fixed sample size $N = 10^4$ (a relatively large sample size) for selected moments

assure convergence of the MM equation for the range with higher values of power shape parameter of GGD, whereas the Mallat's method may not have a solution there. The range of convergence of the MM equation for the Mallat's method also reduced. Fig. 2 depicts the plots for $N = 100$ and $N = 200$ and each iteration was repeated $M = 10^4$. The same repetition count is kept for the rest of calculations.

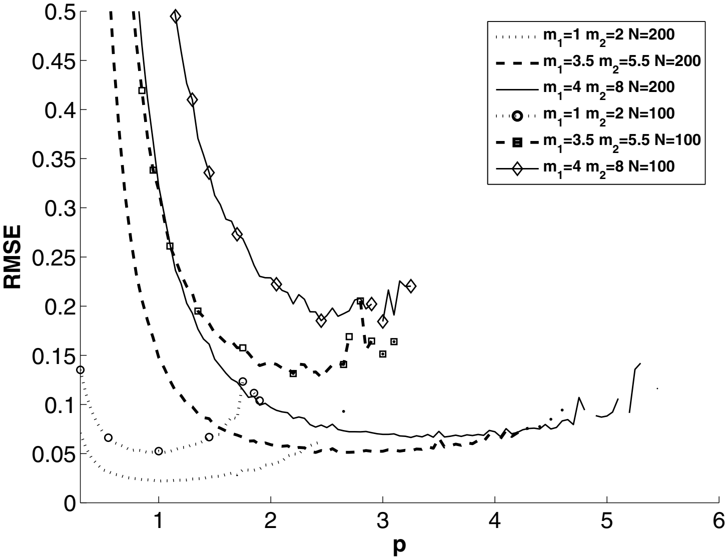


Fig. 2. Comparison of RMSE for the MM method of the estimation of the shape parameter p of GGD with a varying shape value p and a selected fixed sample size $N = 100$ and $N = 200$ for selected moments

By the application of Eq. (6)–(9) the influence of outliers on the MM equation can be alleviated for a small sample size and the higher order moments and the range of convergence can be extended toward the smaller sample sizes. The expansion of $|x_i|^m$ in Eq. (4) up to and including a k-th component, i.e., $(\frac{x_i}{\lfloor x_i \rfloor} \pm 1)^k$ will be denoted as $E_m = f(k)$. Fig. 3 depicts that selecting only a few components in the moments calculation (Eq. (6)–(9)) RMSE is stable and smaller than in the usual calculation. The selected expansions $E_{m1} = f(3)$, $E_{m2} = f(6)$ and $E_{m1} = f(3)$, $E_{m2} = f(8)$ are plotted for the integer moments $m_1 = 4$, $m_2 = 8$ and for the shape parameter $p = 6$ and exhibit better RMSE performance below $N = 300$.

The same behavior for the integer moments $m_1 = 4$ and $m_2 = 8$ and for the shape parameter $p = 5$ is observed. For the selected expansions $E_{m1} = f(3)$, $E_{m2} = f(6)$ and $E_{m1} = f(3)$, $E_{m2} = f(8)$, RMSE is stable and smaller than in the usual calculation below $N = 260$ (Fig. 4).

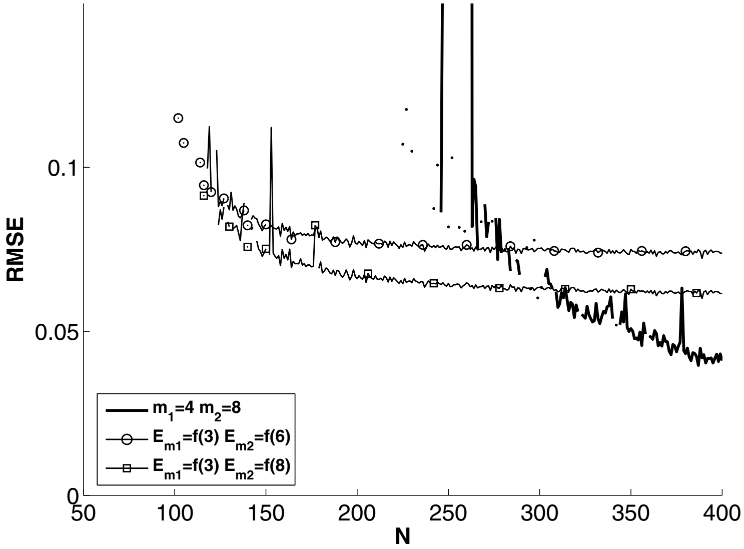


Fig. 3. Comparison of RMSE for the MM method ($m_1 = 4$, $m_2 = 8$) and selected expansions of the estimation of the shape parameter p of GGD with a varying sample size N and for the selected fixed value $p = 6$ in the GGD generator

The selected expansions $E_{m1} = f(3)$, $E_{m2} = f(6)$ and $E_{m1} = f(3)$, $E_{m2} = f(8)$ and the MM method for the integer moments $m_1 = 4$ and $m_2 = 8$ for a small sample size $N = 169$ and for a wider range of p are compared in Fig. 5. It can be noticed that the expansions $E_{m1} = f(3)$, $E_{m2} = f(6)$ and $E_{m1} = f(3)$, $E_{m2} = f(8)$ resulted in smaller RMSE and assured convergence for wider range of p.

In the case of a small sample size for the real higher order moments, the convergence of the MM equation can be also improved toward the observation smaller in size. Fig. 6 depicts a few components in the moments calculation (Eq. (6)–(9)) comparing to the whole moment calculation ($m_1 = 3.5$, $m_2 = 5.5$) over a range of varying sample size N for the fixed value $p = 6$ in the GGD generator. It can be also noted that RMSE is longer stable for $E_{m1} = f(3)$ and $E_{m2} = f(6)$ toward smaller N and has a smaller value too.

For the moments $m_1 = 3.5$ and $m_2 = 5.5$ and for the shape parameter $p = 5$, one of the expansions $E_{m1} = f(3)$ and $E_{m2} = f(6)$ has smaller RMSE and is stable for longer range of N than the full moments, whereas another expansion $E_{m1} = f(3)$ and $E_{m2} = f(4)$ gains RMSE smaller and is still stable below $N = 250$ (Fig. 7).

The selected expansions $E_{m1} = f(3)$, $E_{m2} = f(4)$ and $E_{m1} = f(3)$, $E_{m2} = f(6)$ and the MM method for the real moments $m_1 = 3.5$, $m_2 = 5.5$ for a small sample size $N = 300$ and for a wider range of p are compared in Fig. 8. It can be noticed that the expansions resulted in smaller RMSE and assured

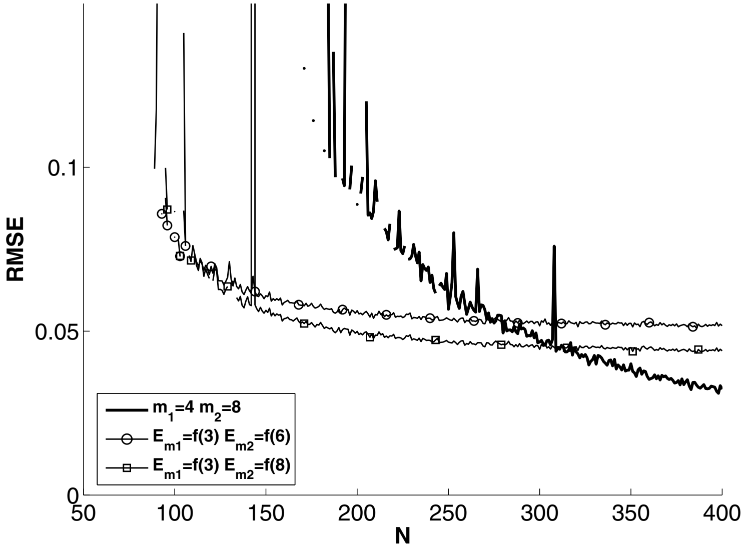


Fig. 4. Comparison of RMSE for the MM method ($m_1 = 4$, $m_2 = 8$) and selected expansions of the estimation of the shape parameter p of GGD with a varying sample size N and for the selected fixed value $p = 5$ in the GGD generator

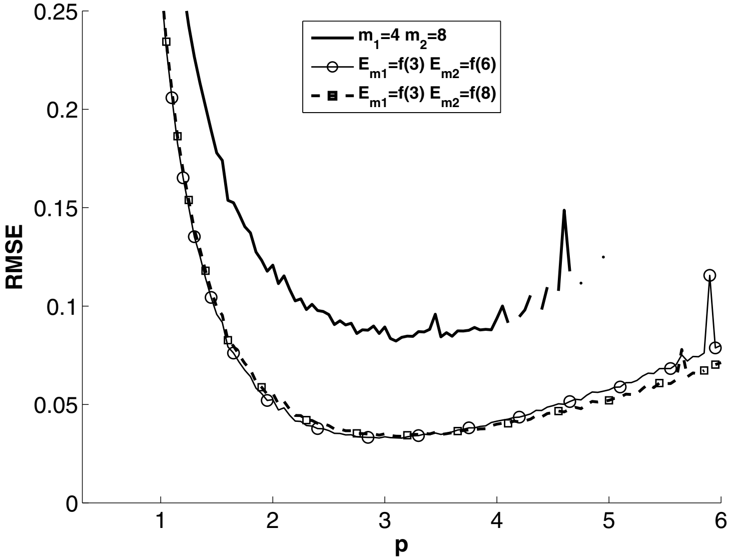


Fig. 5. Comparison of RMSE for the MM method ($m_1 = 4$, $m_2 = 8$) and selected expansions of the estimation of the shape parameter p of GGD with a varying shape value p and a selected fixed sample size $N = 169$

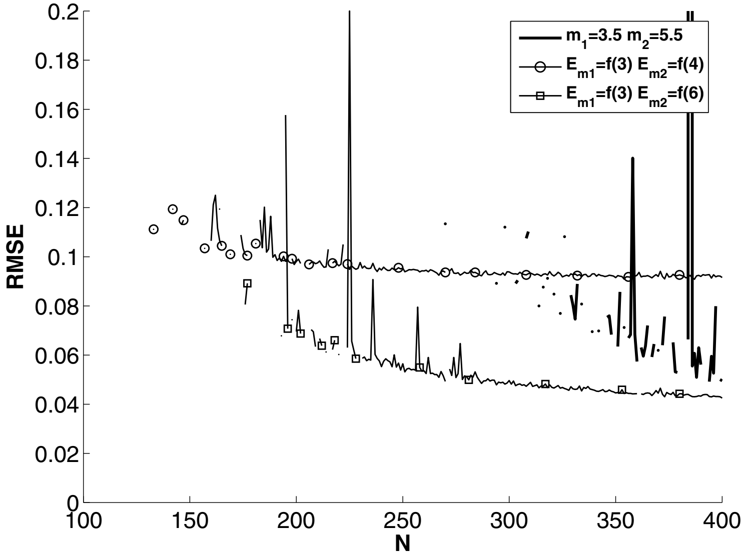


Fig. 6. Comparison of RMSE for the MM method ($m_1 = 3.5$, $m_2 = 5.5$) and selected expansions of the estimation of the shape parameter p of GGD with a varying sample size N and for the selected fixed value $p = 6$ in the GGD generator

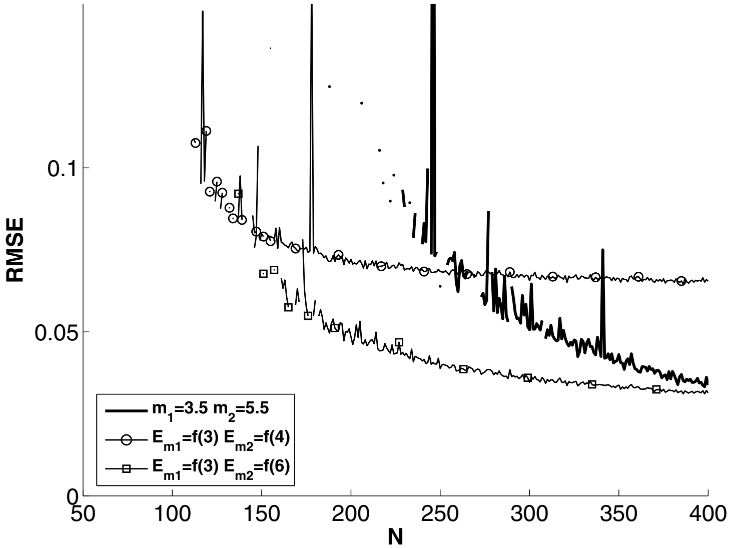


Fig. 7. Comparison of RMSE for the MM method ($m_1 = 3.5$, $m_2 = 5.5$) and selected expansions of the estimation of the shape parameter p of GGD with a varying sample size N and for the selected fixed value $p = 5$ in the GGD generator

convergence for wider range of p. For the one of the expansions, i.e., $E_{m1} = f(3)$ and $E_{m2} = f(4)$ the increase of RMSE is observed comparing to the normal moments calculation, but the results are stable for the wider p range.

The same approach can be applied for other higher order moments both integer and real, but not all expansions lead to better performance.

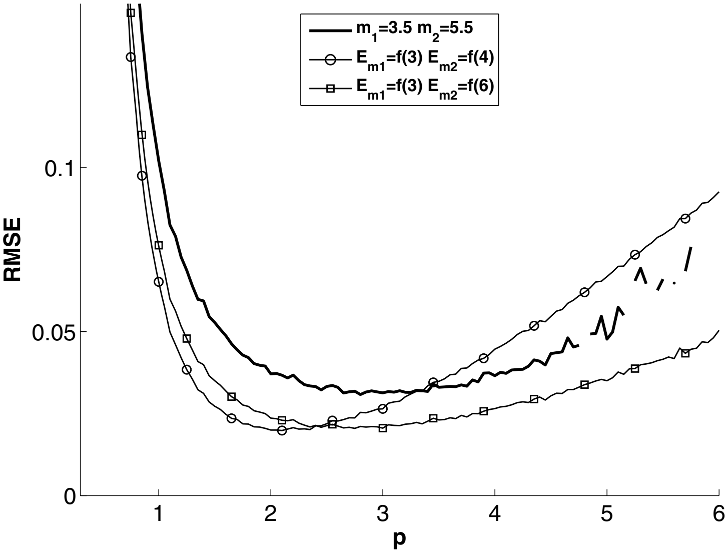


Fig. 8. Comparison of RMSE for the MM method ($m_1 = 3.5$, $m_2 = 5.5$) and selected expansions of the estimation of the shape parameter p of GGD with a varying shape value p and a selected fixed sample size $N = 300$

4 Conclusions

The article focuses on the estimation of the shape parameter of GGD when a small size is only available. The experimental results and the comparison with already known MM method, which show that the method proposed is very useful from a practical point of view. The smaller sample size the variance of the MM estimator increases and the moment method equation may not converge to a real solution for some sample sets. For the higher order moments, the presented approach allows to lower RMSE and ensure a real solution for the smaller sample sets of GGD for the modified MM method.

References

1. Mallat, S.G.: A Theory of Multiresolution Signal Decomposition: The Wavelet Representation. IEEE Trans. Pattern Anal. Machine Intell. 11, 674–693 (1989)
2. Novey, M., Adali, T., Roy, A.: A complex generalized Gaussian distribution – characterization, generation, and estimation. IEEE Transactions on Signal Processing 58, 1427–1433 (2010)
3. Du, Y.: Ein sphärisch invariantes Verbunddichtemodell für Bildsignale. Archiv für Elektronik und Übertragungstechnik AEÜ-45, 148–159 (1991)
4. Meignen, S., Meignen, H.: On the modeling of small sample distributions with generalized Gaussian density in a maximum likelihood framework. IEEE Transactions on Image Processing 15, 1647–1652 (2006)
5. Varanasi, M.K., Aazhang, B.: Parametric generalized Gaussian density estimation. J. Acoust. Soc. Amer. 86, 1404–1415 (1989)
6. Krupiński, R., Purczyński, J.: Modeling the distribution of DCT coefficients for JPEG reconstruction. Image Commun. 22, 439–447 (2007)
7. Krupiński, R., Purczyński, J.: Approximated fast estimator for the shape parameter of generalized Gaussian distribution. Signal Process 86, 205–211 (2006)
8. Box, G., Tiao, G.C.: Bayesian Inference in Statistical Analysis. Addison Wesley, Reading (1973)
9. Olver, F.W.J.: Asymptotics and special functions. Academic Press, New York (1974)

Trajectory Estimation for Exponential Parameterization and Different Samplings

Ryszard Kozera[1], Lyle Noakes[2], and Piotr Szmielew[1]

[1] Warsaw University of Life Sciences - SGGW
Faculty of Applied Informatics and Mathematics
Nowoursynowska str. 159, 02-776 Warsaw, Poland
[2] Department of Mathematics and Statistics
The University of Western Australia
35 Stirling Highway, Crawley W.A. 6009, Perth, Australia
ryszard_kozera@sggw.pl, lyle.noakes@maths.uwa.edu.au,
p.szmielew@ieee.org

Abstract. This paper discusses the issue of fitting reduced data $Q_m = \{q_i\}_{i=0}^m$ with piecewise-quadratics to estimate an unknown curve γ in Euclidean space. The interpolation knots $\{t_i\}_{i=0}^m$ with $\gamma(t_i) = q_i$ are assumed to be unknown. Such *non-parametric interpolation* commonly appears in computer graphics and vision, engineering and physics [1]. We analyze a special scheme aimed to supply the missing knots $\{\hat{t}_i^\lambda\}_{i=0}^m \approx \{t_i\}_{i=0}^m$ (with $\lambda \in [0,1]$) - the so-called *exponential parameterization* used in computer graphics for curve modeling. *A blind uniform guess*, for $\lambda = 0$ coupled with more-or-less uniform samplings yields a linear convergence order in trajectory estimation. In addition, for ε-uniform samplings ($\varepsilon \geq 0$) and $\lambda = 0$ an extra acceleration $\alpha_\varepsilon(0) = \min\{3, 1+2\varepsilon\}$ follows [2]. On the other hand, for $\lambda = 1$ *cumulative chords* render a cubic convergence order $\alpha(1) = 3$ within a general class of admissible samplings [3]. A recent theoretical result [4] is that for $\lambda \in [0,1)$ and more-or-less uniform samplings, sharp orders $\alpha(\lambda) = 1$ eventuate. Thus no acceleration in $\alpha(\lambda) < \alpha(1) = 3$ prevails while $\lambda \in [0,1)$. Finally, another recent result [5] proves that for all $\lambda \in [0,1)$ and ε-uniform samplings, the respective accelerated orders $\alpha_\varepsilon(\lambda) = \min\{3, 1+2\varepsilon\}$ are independent of λ. The latter extends the case of $\alpha_\varepsilon(\lambda = 0) = 1+2\varepsilon$ to all $\lambda \in [0,1)$. We revisit here [4] and [5] and verify their sharpness experimentally.

Keywords: Interpolation, numerical analysis, computer graphics and vision.

1 Introduction

The sampled data points $Q_m = \{q_i\}_{i=0}^m$ with $\gamma(t_i) = q_i \in \mathbb{R}^n$ define the pair $(\{t_i\}_{i=0}^m, Q_m)$ commonly coined as *non-reduced data*. We also require here that $t_i < t_{i+1}$ and $q_i \neq q_{i+1}$ hold. Moreover, assume that $\gamma : [0,T] \to \mathbb{R}^n$ (with $0 < T < \infty$) is sufficiently smooth (specified later) and that it defines a regular curve $\dot{\gamma}(t) \neq \mathbf{0}$. In order to estimate the unknown curve γ with an arbitrary

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 430–441, 2013.

interpolant $\bar{\gamma} : [0, T] \to \mathbb{R}^n$ it is necessary to assume that $\{t_i\}_{i=0}^m \in V_G^m$, i.e. that the following *admissibility condition* is satisfied:

$$\lim_{m \to \infty} \delta_m = 0, \quad \text{where} \quad \delta_m = \max_{0 \leq i \leq m-1} (t_{i+1} - t_i). \tag{1}$$

We omit here the subscript m in δ_m by setting $\delta = \delta_m$. In this paper, two substantial subfamilies of V_G^m are discussed.

The *first one* $V_{mol}^m \subset V_G^m$ includes *more-or-less uniform samplings* [6], [7]:

$$\beta\delta \leq t_{i+1} - t_i \leq \delta, \tag{2}$$

for some $\beta \in (0, 1]$. The left inequality in (2) excludes samplings with distance between consecutive knots smaller then $\beta\delta$. The right inequality follows from (1). Condition (2), as shown in [6], can be replaced by the equivalent condition (3) holding for each $i = 0, 1, \ldots m-1$ and some constants $0 < K_1 \leq K_2$:

$$\frac{K_1}{m} \leq t_{i+1} - t_i \leq \frac{K_2}{m}. \tag{3}$$

The *second subfamily* $V_\varepsilon^m \subset V_G^m$ is that of *ε-uniform samplings* [2]:

$$t_i = \phi(\frac{iT}{m}) + O(\frac{1}{m^{1+\varepsilon}}), \tag{4}$$

where $\varepsilon \geqslant 0$, $\phi : [0, T] \to [0, T]$ is smooth and $\dot{\phi} > 0$ (so that $t_i < t_{i+1}$). Clearly, the smaller ε gets, the bigger distortion of uniform distribution occurs (modulo ϕ). The case when $\varepsilon = 0$ needs a special attention so that the inequality $t_i < t_{i+1}$ holds. However, the latter is asymptotically guaranteed for all ε positive. Note that each ε-uniform sampling with $\varepsilon > 0$ is also more-or-less uniform [6].

2 Problem Formulation and Motivation

We say that the family $F_\delta : [0, T] \to \mathbb{R}^n$ satisfies $F_\delta = O(\delta^\alpha)$ if $\|F_\delta\| = O(\delta^\alpha)$, where $\|\cdot\|$ denotes the Euclidean norm. Another words there are constants $K > 0$ and $\delta_0 > 0$ such that $\|F_\delta\| \leq K\delta^\alpha$, for all $\delta \in (0, \delta_0)$ and $t \in [0, T]$ - [8].

A standard result for *non-reduced data* $(\{t_i\}_{i=0}^m, Q_m)$ for piecewise r-degree polynomial $\bar{\gamma} = \tilde{\gamma}_r$ reads [6], [9]:

Theorem 1. *Let $\gamma \in C^{r+1}$ be a regular curve $\gamma : [0, T] \to \mathbb{R}^n$ with knot parameters $\{t_i\}_{i=0}^m \in V_G^m$ given. Then a piecewise r-degree Lagrange polynomial interpolation $\tilde{\gamma}_r$ used with $\{t_i\}_{i=0}^m$ known, yields a sharp estimate:*

$$\tilde{\gamma}_r = \gamma + O(\delta^{r+1}). \tag{5}$$

By (5) piecewise-quadratics (-cubics) $\tilde{\gamma}_2$ ($\tilde{\gamma}_3$) yield *cubic* (*quartic*) order error terms.

In many applications in computer graphics and computer vision, engineering or physics, the so-called *reduced data* Q_m are encountered (see e.g. [1], [10], [11], or [12]), where the knots $\{t_i\}_{i=0}^m$ are unknown and have to be first guessed somehow. A family of the so-called *exponential parameterization* $\{\hat{t}_i\}_{i=0}^m \approx \{t_i\}_{i=0}^m$ is commonly used for curve modeling [11], [13]:

$$\hat{t}_0 = 0, \qquad \hat{t}_{i+1} = \hat{t}_i + \|q_{i+1} - q_i\|^{\lambda}, \tag{6}$$

where $0 \leq \lambda \leq 1$ and $i = 0, 1, \ldots, m-1$. The cases when $\lambda \in \{1, 0.5, 0\}$, yield *cumulative chords*, *centripetal* or *blind uniform parameterizations*, respectively.

We call a piecewise r-degree polynomial based on (6) and Q_m as $\hat{\gamma} = \hat{\gamma}_r : [0, \hat{T}] \to \mathbb{R}^n$, where $\hat{T} = \sum_{i=0}^{m-1} \|q_{i+1} - q_i\|^{\lambda}$. Note that in case of any reduced data Q_m for asymptotics estimation of γ by $\hat{\gamma}_r$, *a re-parameterization* $\psi : [0, T] \to [0, \hat{T}]$ synchronizing both domains of γ and $\hat{\gamma}_r$, needs to be defined (see e.g. [6]).

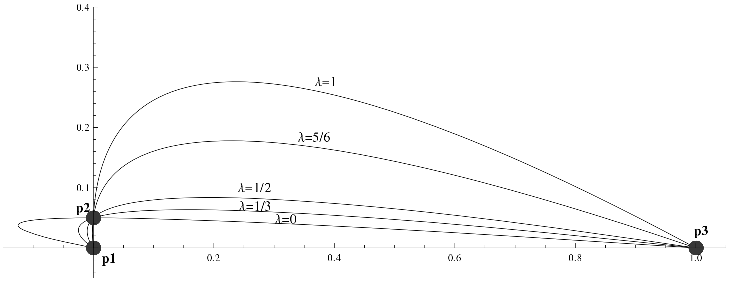


Fig. 1. Interpolating three points $Q_2 = \{(0,0), (0,0.05), (1,0)\}$ with $\hat{\gamma}_2$, for $\lambda = 0, 1/3, 1/2, 5/6, 1 \in [0,1]$

Example 1. Figure 1 shows different $\hat{\gamma}_2$ passing through Q_2 with various $\lambda \in \{0, 1/3, 1/2, 5/6, 1\}$ set in (6). Such curves' fluctuation given different knots and interpolation schemes is commonly exploited for sparse data in 2D and 3D computer graphics in the context of curve modeling - see [10], [11], or [12]. □

Example 2. Another application, elucidating the influence of knots selection on interpolation stems from the computer vision field. Figure 2 shows the image of the same knee joint section. The goal is to isolate from such image the kneecap and to find its area, amounting here to $A = 5237$ pixels. The interpolation points Q_m positioned on the boundary are selected e.g. by the physician (here $m = 5$). Of course, the internal parametrization of the kneecap boundary (i.e. some curve γ) remains unknown. Upon invoking $\hat{\gamma}_2$ (with three quadratic segments) coupled with guessed knots in accordance with (6) we obtain different estimates of γ by $\hat{\gamma}_2$ and consequently various kneecap area A_λ approximations. Namely for $\lambda \in \{0, 1/4, 1/2, 3/4, 1\}$ the following $A \approx A_\lambda \in \{5197, 5209, 5234, 5293, 5376\}$ (in pixels) hold, respectively. The centripetal parameterization (i.e. for $\lambda = 1/2$) on this specific sparse data Q_m yields the best result. □

More *real data examples* emphasizing the importance of the knots' selection for a given interpolation scheme in *computer graphics* (light-source motion estimation or image rendering), *computer vision* (image segmentation or video compression), *geometry* (trajectory, curvature or area estimation) or in *engineering and physics* (fast particles' motion estimation) can be found e.g. in [1].

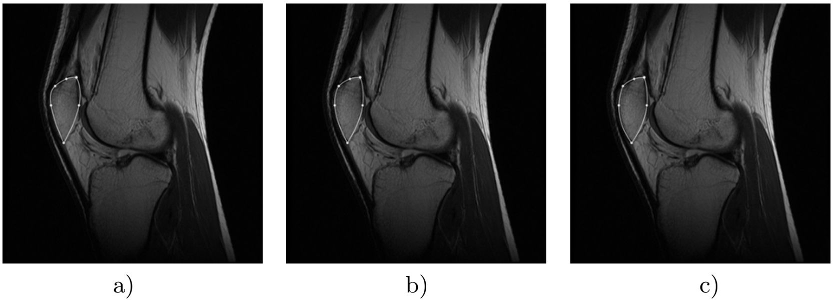


Fig. 2. Isolating the kneecap with $\hat{\gamma}_2$, for a) $\lambda = 0$, b) $\lambda = 0.5$, c) $\lambda = 1$

2.1 Uniform Parameterization - $\lambda = 0$

The case when $\lambda = 0$, transforms (6) into to blind *uniform* knots' guesses $\hat{t}_i = i$. For $r = 2$ and $\lambda = 0$ in (6) the following holds [2]:

Theorem 2. *Let the unknown $\{t_i\}_{i=0}^m$ be sampled ε-uniformly, where $\varepsilon > 0$ and $\gamma \in C^4$. Then there is a uniform piecewise-quadratic Lagrange interpolant $\hat{\gamma}_2 : [0, \hat{T} = m] \to \mathbb{R}^n$, calculable in terms of Q_m (with $\hat{t}_i = i$) and piecewise C^∞ re-parameterization $\psi : [0, T] \to [0, \hat{T}]$ such that sharp estimates hold:*

$$\hat{\gamma}_2 \circ \psi = \gamma + O(\delta^{\min\{3,1+2\varepsilon\}}). \tag{7}$$

Th. 2 with $\alpha_{\varepsilon>0}(0) = \min\{3, 1 + 2\varepsilon\}$ extends to $\varepsilon = 0$ provided $\{t_i\}_{i=0}^m$ satisfies $t_i < t_{i+1}$ and falls also into more-or-less uniformity (2). The latter renders linear convergence order $\alpha_{\varepsilon=0}(0) = 1$ - see [6]. Evidently, for ε-uniform samplings there is an acceleration from $\alpha_{\varepsilon=0}(0) = 1$ via $\alpha_{0<\varepsilon<1}(0) = 1 + 2\varepsilon$ to $\alpha_{\varepsilon\geq1}(0) = 3$.

2.2 Cumulative Chords - $\lambda = 1$

The opposite case when $\lambda = 1$ in (6) renders *cumulative chords* [11], [12]. This choice of $\{\hat{t}_i\}_{i=0}^m$ uses the geometry of Q_m and gives better trajectory estimation (at least for $r = 2, 3$) as opposed to $\lambda = 0$ and (7) [3]:

Theorem 3. *Let γ be a regular C^k curve in $\mathbb{R}^n$, where $k \geqslant r + 1$ and $r = 2, 3$ sampled according to (1). Let $\hat{\gamma}_r : [0, \hat{T} = \sum_{i=0}^{m-1} \|q_{i+1} - q_i\|] \to \mathbb{R}^n$ be the cumulative chord piecewise-quadratic(-cubic) interpolant defined by Q_m and $\lambda = 1$ in (6). Then there is a piecewise-C^r re-parameterization $\psi : [0, T] \to [0, \hat{T}]$, with*

$$\hat{\gamma}_r \circ \psi = \gamma + O(\delta^{r+1}). \tag{8}$$

The asymptotics from Th. 2 and Th. 3 are sharp - see [3] and [6]. For $r = 2$ and $\lambda = 1$, formula (8) yields the cubic order $\alpha(1) = 3$ which not only improves (7) but also matches the non-reduced data case (5) (with $r = 2, 3$).

2.3 Exponential Parameterization - $\lambda \in [0,1]$

Recent research by [4] extends the results from Th. 2 (where $\lambda = 0$) and Th. 3 (where $\lambda = 1$ with $r = 2$) to the remaining cases of exponential parameterization (6) i.e. to $\lambda \in [0,1]$. As proved in [4], for more-or-less uniform samplings (2), (6) and $r = 2$ any choice of $\lambda \in [0,1)$ does not improve the asymptotics for γ approximation. In fact, for all $\lambda \in [0,1)$ we have $\alpha(\lambda) = 1$. Indeed we obtain [4]:

Theorem 4. *Suppose γ is a regular C^3 curve in $\mathbb{R}^n$ sampled more-or-less uniformly (2). Let $\hat{\gamma}_2 : [0, \hat{T} = \sum_{i=0}^{m-1} \|q_{i+1} - q_i\|^{\lambda}] \to \mathbb{R}^n$ be the piecewise-quadratic interpolant defined by Q_m and (6) (with $\lambda \in [0,1]$). Then there is a piecewise-C^∞ re-parameterization $\psi : [0,T] \to [0,\hat{T}]$, such that for $\lambda \in [0,1)$ we have:*

$$\hat{\gamma}_2 \circ \psi = \gamma + O(\delta). \tag{9}$$

In addition, for either $\{t_i\}_{i=0}^m$ uniform or $\lambda = 1$ used with samplings (1) the following holds:

$$\hat{\gamma}_2 \circ \psi = \gamma + O(\delta^3). \tag{10}$$

Both (9) and (10) are sharp (proved analytically). Th. 4 underlines discontinuity of $\alpha(\lambda)$ at $\lambda = 1$ with the jump by 2 in respective convergence orders. Another unexpected fact comes from the proof of Th. 4. Namely, a natural candidate for a re-parameterization, i.e. a Lagrange quadratic $\psi_i : [t_i, t_{i+2}] \to [\hat{t}_i, \hat{t}_{i+2}]$ satisfying $\psi_i(t_{i+j}) = \hat{t}_{i+j}$ (for $j = 0,1,2$) can be a non-injective function [4].

The most recent result [5] shows that for ε-uniform samplings (4) (with $\varepsilon > 0$) the asymptotics established in Th. 4 improves from $\alpha(\lambda) = 1$ to $\alpha_{\varepsilon>0}(\lambda) = \min\{3, 1+2\varepsilon\}$, for each $\lambda \in [0,1)$. Indeed the following holds [5]:

Theorem 5. *Suppose γ is a regular C^4 curve in $\mathbb{R}^n$ sampled according to the ε-uniformity condition (4) with $\varepsilon > 0$. Let $\hat{\gamma}_2 : [0, \hat{T} = \sum_{i=0}^{m-1} \|q_{i+1} - q_i\|^{\lambda}] \to \mathbb{R}^n$ be the piecewise-quadratic interpolant defined by Q_m and (6) (with $\lambda \in [0,1)$). Then there is a piecewise-C^∞ re-parameterization $\psi : [0,T] \to [0,\hat{T}]$, such that:*

$$\hat{\gamma}_2 \circ \psi = \gamma + O(\delta^{\min\{3,1+2\varepsilon\}}). \tag{11}$$

By Th. 4, formula (11) extends to $\varepsilon = 0$ (with $\lambda \in [0,1)$) if extra condition (2) on 0-uniform sampling is imposed. The case $\lambda = 1$ by Th. 4 yields $\hat{\gamma}_2 \circ \psi = \gamma + O(\delta^3)$.

Again (11) is proved analytically to be sharp. Clearly, by Th. 5 an extra acceleration (11) in convergence rates for all $\lambda \in [0,1]$ and (4) with $\varepsilon > 0$ occurs. The latter coincides with Th. 2 holding for $\lambda = 0$. The formula (11) is only dependent on ε, not on λ. It should be pointed out that by [5], for each $\varepsilon > 0$ the quadratic ψ_i defines a genuine re-parameterization of $[t_i, t_{i+2}]$ into $[\hat{t}_i, \hat{t}_{i+2}]$.

2.4 Aim of This Research

In this paper we verify experimentally *the sharpness* of asymptotics for trajectory estimation claimed by Th. 4 and Th. 5. By *sharpness* we understand the existence of at least one curve $\gamma \in C^r([0,T])$ (with r set accordingly) sampled with some admissible samplings $\{t_i\}_{i=0}^m \in V_G^m$ for which the asymptotic estimates in question are exactly matched. The tests conducted herein are confined merely to the planar and spatial curves. It should, however be emphasized that all quoted herein Theorems 1-5 admit regular curves in $\mathbb{R}^n$. Some motivation standing behind the applications of interpolating reduced data is also here presented. More examples of real reduced n-dimensional data Q_m which can be fitted with piecewise-quadratics $\hat{\gamma}_2$ or any other interpolation schemes based on exponential parameterization (6) can be found e.g. in [1].

3 Experiments

All tests presented in this paper are performed in *Mathematica 9.0* [14] on a 2.4GHZ Intel Core 2 Duo computer with 8GB RAM. Note that since $T = \sum_{i=1}^m (t_{i+1} - t_i) \leq m\delta$ the following holds $m^{-\alpha} = O(\delta^\alpha)$, for $\alpha > 0$. Hence, the verification of any asymptotics expressed in terms of $O(\delta^\alpha)$ can be performed by examining the claims of Th. 4 or Th. 5 in terms of $O(1/m^\alpha)$ asymptotics.

Note that for a parametric regular curve $\gamma : [0,T] \to \mathbb{R}^n$, $\lambda \in [0,1]$ and m varying between $m_{min} \leq m \leq m_{max}$ the i-th component of the error for γ estimation is defined here as follows:

$$E_m^i = \sup_{t \in [t_i, t_{i+2}]} \|(\hat{\gamma}_{2,i} \circ \psi_i)(t) - \gamma(t)\| = max_{t \in [t_i, t_{i+2}]} \|(\hat{\gamma}_{2,i} \circ \psi_i)(t) - \gamma(t)\|,$$

as $\tilde{E}_m^i(t) = \|(\hat{\gamma}_{2,i} \circ \psi_i)(t) - \gamma(t)\| \geq 0$ is continuous over each compact subinterval $[t_i, t_{i+2}] \subset [0,T]$. The maximal value E_m of $\tilde{E}_m(t)$ (the track-sum of $\tilde{E}_m^i(t)$), for each $m = 2k$ (here $k = 1, 2, 3, \ldots, m/2$) is found by using *Mathematica* optimization built-in functions: *Maximize* or *FindMinimum* (the latter applied to $-\tilde{E}_m(t)$). From the set of *absolute errors* $\{E_m\}_{m=m_{min}}^{m_{max}}$ the numerical estimate $\bar{\alpha}(\lambda)$ of genuine order $\alpha(\lambda)$ is subsequently computed by using *a linear regression* applied to the pair of points $(\log(m), -\log(E_m))$ (see also [6]). Since piecewisely $deg(\hat{\gamma}_2) = 2$ the number of interpolation points $\{q_i\}_{i=0}^m$ is odd i.e. $m = 2k$ as indexing runs over $0 \leq i \leq m$. The *Mathematica* built-in functions *LinearModelFit* renders the coefficient $\bar{\alpha}(\lambda)$ from the computed regression line $y(x) = \bar{\alpha}(\lambda)x + b$ based on pairs of points $\{(\log(m), -\log(E_m))\}_{m=m_{min}}^{m_{max}}$. Finally, recall that as justified in Th. 5 any ε-uniform sampling with $\varepsilon > 0$ gives asymptotically ψ_i as re-parameterization of $[t_i, t_{i+2}]$ into $[\hat{t}_i, \hat{t}_{i+2}]$. Once $\varepsilon = 0$, one may apply a simple computational test (for $m = m_{max}$) by verifying whether either $\psi_i^{(1)}(t_i) \geq 0$ and $\psi_i^{(1)}(t_{i+2}) > 0$ or $\psi_i^{(1)}(t_i) > 0$ and $\psi_i^{(1)}(t_{i+2}) \geq 0$ hold over each subinterval $[t_i, t_{i+2}]$. The latter combined with the linearity of $\psi_i^{(1)}$ guarantees that ψ_i is a re-parameterization. More discussion on the issue of enforcing ψ_i to be a re-parameterization can be found in [5].

3.1 Fitting Reduced Data for Planar Curves

The testing commences with the simplest possible curve, i.e. a straight line.

Example 3. Consider *a regular straight line*: $\gamma_l(t) = \left(t/\sqrt{5}, 2t/\sqrt{5}\right) \subset \mathbb{R}^2$ for $t \in [0,1]$, sampled with $\delta_1 = i/m$ according to ε-uniform sampling (4) (with $\varepsilon > 0$):

$$t_{i+1} - t_i = \delta_1(1 + \delta_1^{\varepsilon}) \quad \text{and} \quad t_{i+2} - t_i = \delta_1(1 - \delta_1^{\varepsilon}), \tag{12}$$

where $t_0 = 0$ and $t_m = 1$. The plot of γ_l sampled by (12), with $\varepsilon = 0.5$ and $m = 12$ is shown in Figure 3. Recalling (1), note that here $\delta = \delta_1(1 + \delta_1^{\varepsilon})$. As $\varepsilon > 0$, the quadratic ψ_i is a re-parameterization [5]. The linear regression applied to $m_{min} = 100 \leq m \leq m_{max} = 120$ yields the estimates for $\bar{\alpha}_{\varepsilon}(\lambda) \approx \alpha_{\varepsilon}(\lambda) = \min\{3, 1+2\varepsilon\}$ which are presented in Table 1. An inspection of Table 1 confirms *the sharpness or nearly sharpness* of Th. 5. □

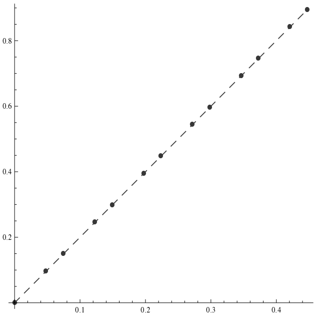


Fig. 3. The plot of the line γ_l sampled as in (12), for $m = 12$ and $\varepsilon = 0.5$

Table 1. Computed $\bar{\alpha}_{\varepsilon}(\lambda) \approx \alpha_{\varepsilon}(\lambda) = 1 + 2\varepsilon$ for γ_l and sampling (12) interpolated by $\hat{\gamma}_2$ with some discrete values $\lambda \in [0,1)$ and $\varepsilon \in (0,1]$

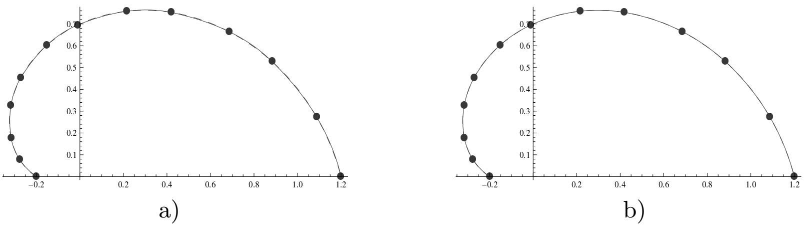
λ	$\varepsilon = 0.1$	$\varepsilon = 0.33$	$\varepsilon = 0.5$	$\varepsilon = 0.7$	$\varepsilon = 0.9$	$\varepsilon = 1$
$\alpha_{\varepsilon}(\lambda)$	1.20	1.66	2.00	2.40	2.80	3.00
0.00	1.47	1.80	2.10	2.46	2.85	3.04
0.10	1.45	1.80	2.10	2.46	2.85	3.04
0.33	1.42	1.80	2.10	2.46	2.85	3.04
0.50	1.39	1.80	2.10	2.46	2.85	3.04
0.70	1.37	1.79	2.10	2.47	2.85	3.04
0.90	1.36	1.79	2.10	2.47	2.85	3.04

We pass now to the next example involving a spiral curve in $\mathbb{R}^2$.

Example 4. Consider now the following regular *spiral curve* $\gamma_{sp1} : [0,1] \to \mathbb{R}^2$: $\gamma_{sp1}(t) = ((t+0.2)\cos(\pi(1-t)), (t+0.2)\sin(\pi(1-t)))$, sampled according to the following ε-uniform sampling (4):

$$t_i = \frac{i}{m} + \frac{(-1)^{i+1}}{m^{1+\varepsilon}}, \tag{13}$$

with $t_0 = 0$ and $t_m = 1$. Figure 4 shows γ_{sp1} (a dashed line) and $\hat{\gamma}_2$ (a continuous line) sampled by (13) with $\varepsilon = 0.7$, $m = 12$ and $\lambda \in \{0, 1\}$. The difference between both $\hat{\gamma}_2$ and γ_{sp1} on sparse data Q_{12} is minor as they both overlap.



a)

b)

Fig. 4. The plot of the spiral γ_{sp1} sampled as in (13) (a dashed line) and interpolant $\hat{\gamma}_2$ (a continuous line), for $m = 12$ and $\varepsilon = 0.7$ with either a) $\lambda = 0$ or b) $\lambda = 1$

The Th. 5, for $\varepsilon > 0$ yields $\psi_i : [t_i, t_{i+2}] \to [\hat{t}_i, \hat{t}_{i+2}]$ as a re-parameterization. The case of $\varepsilon = 0$ renders (13) as more-or-less uniform (3) with $K_1 = 1/3$ and $K_2 = 5/3$. Sufficient conditions for ψ_i to be a re-parameterization are formulated in [5]. The latter enables to verify the validity of Th. 4 also for $\varepsilon = 0$. The linear regression applied to $m = 100 \leq m \leq m_{max} = 120$ renders computed $\bar{\alpha}_\varepsilon(\lambda) \approx \alpha_\varepsilon(\lambda) = \min\{3, 1 + 2\varepsilon\}$ $(\varepsilon \geq 0)$, which are listed in Table 2. Visibly, *the sharpness* or *nearly sharpness* of Th. 4 and Th. 5 is confirmed in Table 2. □

Table 2. Estimated $\bar{\alpha}_\varepsilon(\lambda) \approx \alpha_\varepsilon(\lambda) = 1 + 2\varepsilon$ for γ_{sp1} and sampling (13) interpolated by $\hat{\gamma}_2$ with $\lambda \in [0, 1]$ and $\varepsilon \in [0, 1]$

λ	$\varepsilon = 0.0$	$\varepsilon = 0.1$	$\varepsilon = 0.33$	$\varepsilon = 0.5$	$\varepsilon = 0.7$	$\varepsilon = 0.9$	$\varepsilon = 1.0$
$\alpha_\varepsilon(\lambda)$	1.000	1.200	1.660	2.000	2.400	2.800	3.000
0.00	0.981	1.286	1.716	2.023	2.419	2.96	3.004
0.10	0.983	1.282	1.718	2.029	2.435	2.97	3.005
0.33	0.985	1.277	1.726	2.051	2.496	2.93	3.016
0.50	0.988	1.276	1.740	2.081	2.584	3.01	3.017
0.70	0.995	1.283	1.778	2.178	2.782	2.94	3.030
0.90	1.036	1.354	2.051	2.271	3.005	2.89	3.031
$\alpha_\varepsilon(1)$	3.000	3.000	3.000	3.000	3.000	3.000	3.000
1.00	3.057	2.990	2.996	3.007	3.016	2.88	3.031

The last example involving curve in $\mathbb{R}^2$ refers to another spiral.

Example 5. Let *a planar regular convex spiral* $\gamma_{sp} : [0, 5\pi] \to \mathbb{R}^2$: $\gamma_{sp}(t) = ((6\pi - t)\cos(t), (6\pi - t)\sin(t))$ be sampled according to (13) (rescaled by factor 5π) with $t_0 = 0$ and $t_m = 5\pi$. Figure 5 illustrates γ_{sp} (a dashed line) and $\hat{\gamma}_2$ (a continuous line) coupled with (13), for $\varepsilon = 0.33$, $m = 22$ and $\lambda \in \{0, 1\}$.

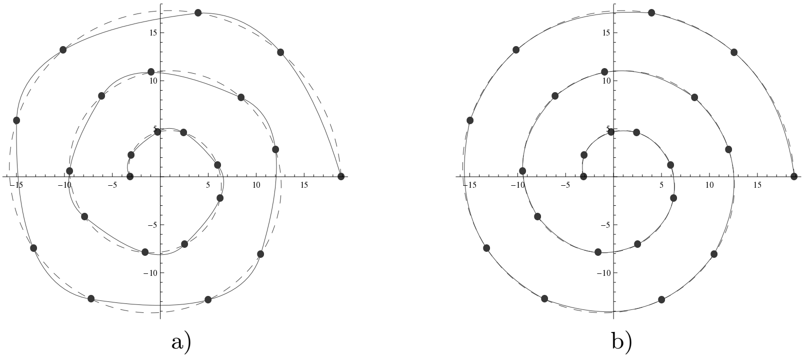


Fig. 5. The plot of the spiral γ_{sp} sampled as in (13) (a dashed line) and interpolant $\hat{\gamma}_2$ (a continuous line), for $m = 22$ and $\varepsilon = 0.33$ with either a) $\lambda = 0$ or b) $\lambda = 1$

The difference between γ_{sp} and $\hat{\gamma}_2$ on reduced data Q_{22} is transparent (at least for $\lambda = 0$). As explained previously, the sampling (13) enforces ψ_i to be a re-parameterization for $\varepsilon > 0$. For $\varepsilon = 0$ one needs to resort to the sufficient conditions for $\psi_i^{(1)} > 0$ to hold (see [5]). In order to estimate the relevant coefficients $\alpha_\varepsilon(\lambda)$ a linear regression is again applied to $100 = m_{min} \leq m \leq m_{max} = 120$. The numerical results are listed in Table 3. Some computed $\alpha_\varepsilon(\lambda)$ from Table 3 exceed convergence orders claimed by Th. 5. However, the first column in Table 3 shows *the sharpness* of Th. 4. □

Table 3. Estimated $\bar{\alpha}_\varepsilon(\lambda) \approx \alpha_\varepsilon(\lambda) = 1 + 2\varepsilon$ for γ_{sp} and sampling (13) interpolated by $\hat{\gamma}_2$ with $\lambda \in [0, 1]$ and $\varepsilon \in [0, 1]$

λ	$\varepsilon = 0.0$	$\varepsilon = 0.1$	$\varepsilon = 0.33$	$\varepsilon = 0.5$	$\varepsilon = 0.7$	$\varepsilon = 0.9$	$\varepsilon = 1.0$
$\alpha_\varepsilon(\lambda)$	1.000	1.200	1.660	2.000	2.400	2.800	3.000
0.00	0.990	1.303	1.799	2.223	2.851	2.986	3.008
0.10	0.990	1.299	1.812	2.342	2.911	2.986	3.007
0.33	0.991	1.296	1.872	2.252	2.966	3.011	3.024
0.50	0,995	1.303	1.989	2.711	2.995	3.020	3.022
0.70	1.013	1.355	2.377	2.930	3.020	3.024	3.023
0.90	1.291	2.092	2.986	2.043	3.033	3.033	3.024
$\alpha_\varepsilon(1)$	3.000	3.000	3.000	3.000	3.000	3.000	3.000
1.00	3.000	2.866	2.901	2.938	2.976	2.995	3.000

3.2 Fitting Reduced Data for Spatial Curves

The next example deals with the reduced data Q_m obtained by sampling the regular spatial curve in $\mathbb{R}^3$.

Example 6. We verify now the sharpness of Th. 4 and Th. 5 for *a quadratic elliptical helix*: $\gamma_h(t) = (2\cos(t), \sin(t), t^2)$, with $t \in [0, 2\pi]$ and sampled ε-uniformly (4) (here $\phi = id$) according to:

$$t_i = \begin{cases} \frac{2\pi i}{m} & \text{if } i \text{ even}, \\ \frac{2\pi i}{m} + \frac{2\pi}{2m^{1+\varepsilon}} & \text{if } i = 4k+1, \\ \frac{2\pi i}{m} - \frac{2\pi}{2m^{1+\varepsilon}} & \text{if } i = 4k+3. \end{cases} \tag{14}$$

The last knot t_m is set to 2π. Figure 6 illustrates the curve γ_h sampled in accordance with (14) for $\varepsilon = 0.5$ and $m = 22$. For $\varepsilon > 0$, by Th. 5 each quadratic $\psi_i : [t_i, t_{i+2}] \to [\hat{t}_i, \hat{t}_{i+2}]$ is a re-parameterization. Note that $\varepsilon = 0$ in (14) yields also more-or-less uniform sampling (3) with $K_1 = \pi$ and $K_2 = 3\pi$. The latter is stipulated by Th. 4. The sufficient conditions for $\{t_i\}_{i=0}^m$ to yield ψ_i as re-parameterization are specified in [5]. The linear regression is used here for $m_{min} = 100 \leq m \leq m_{max} = 120$. The corresponding computed estimates $\bar{\alpha}_\varepsilon(\lambda) \approx \alpha_\varepsilon(\lambda) = \min\{3, 1 + 2\varepsilon\}$ are shown in Table 4. The experiments are consistent with the asymptotics from Th. 5. Thus the sharpness of (11) is also generically herein confirmed. Note also that the estimated convergence orders $\bar{\alpha}_{\varepsilon=0}(\lambda)$ are substantially faster than those claimed by Th. 4. □

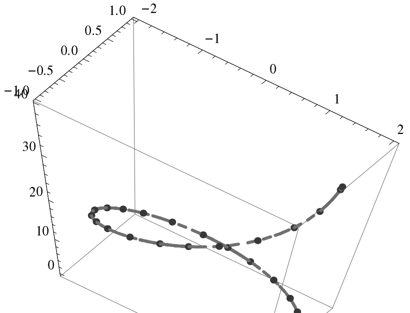


Fig. 6. The plot of the helix γ_h sampled as in (14), for $m = 22$ and $\varepsilon = 0.5$

The linear regression is used to estimate the asymptotic convergence rates $\alpha_\varepsilon(\lambda)$ for sufficiently large m. The estimates may sometimes be misleading when m is not sufficiently large.

Table 4. Estimated $\bar{\alpha}_\varepsilon(\lambda) \approx \alpha_\varepsilon(\lambda) = 1 + 2\varepsilon$ for γ_h and sampling (14) interpolated by $\hat{\gamma}_2$ with $\lambda \in [0, 1]$ and $\varepsilon \in [0, 1]$

λ	$\varepsilon = 0.0$	$\varepsilon = 0.1$	$\varepsilon = 0.33$	$\varepsilon = 0.5$	$\varepsilon = 0.7$	$\varepsilon = 0.9$	$\varepsilon = 1.0$
$\alpha_\varepsilon(\lambda)$	1.00	1.20	1.66	2.00	2.40	2.80	3.00
0.10	2.99	1.26	1.74	2.09	2.54	2.97	3.01
0.33	2.85	1.24	1.72	2.07	2.93	2.93	2.95
0.50	3.24	1.23	1.70	2.06	3.01	3.01	3.04
0.70	3.21	1.20	1.64	2.94	2.94	2.94	3.19
0.90	3.21	1.15	2.89	2.89	2.89	2.89	3.22
$\alpha_\varepsilon(1)$	3.00	3.00	3.00	3.00	3.00	3.00	3.00
1.00	3.21	2.89	2.91	2.92	2.93	2.88	3.21

4 Conclusions

In this paper we discussed the problem of trajectory estimation via piecewise--quadratic interpolation based on reduced data Q_m. In particular, the exponential parameterization (6) which depends on parameter $\lambda \in [0, 1]$ is herein examined. The latter is commonly used in computer graphics for curve modeling - see e.g. [9], [10], [11] or [12]. Special cases of (6) with $\lambda = 0$ (see e.g. [2]) or $\lambda = 1$ (see e.g. [3] or [6]) have been studied in the literature. Recent results from [4] and [5] with full mathematical proofs analyze the asymptotics in question for the remaining cases of $\lambda \in (0, 1)$.

Th. 4 claims that the if $\{t_i\}_{i=0}^m$ is more-or-less uniform (2) there is no acceleration in trajectory estimation once λ varies within the interval $[0, 1)$. The convergence orders are constant, i.e. $\alpha(\lambda) = 1$ for all $\lambda \in [0, 1)$. In addition, there is a discontinuity at $\lambda = 1$ with a jump to $\alpha(1) = 3$ (valid for the general class of admissible samplings (1)).

It is known [2] that further acceleration can be achieved for ε-uniform samplings (4) (for $\varepsilon > 0$) and $\lambda = 0$ by reaching $\alpha_\varepsilon(0) = \min\{3, 1 + 2\varepsilon\}$. The most recent result by [5] extends the latter to arbitrary $\lambda \in [0, 1)$ and $\varepsilon \geq 0$. The case $\varepsilon = 0$ is also admitted provided the curve γ is sampled according to (2). As established in Th. 5 the acceleration amounting to $\alpha_\varepsilon(\lambda) = \min\{3, 1 + 2\varepsilon\}$ is merely dependent on ε (not on $\lambda \in [0, 1]$). Visibly the discontinuity in $\alpha_\varepsilon(\lambda)$ at $\lambda = 1$ is removed for $\varepsilon \geq 1$. It should also be emphasized that the proof of Th. 5 shows that the Lagrange quadratic $\psi_i : [t_i, t_{i+2}] \to [\hat{t}_i, \hat{t}_{i+2}]$ satisfying $\psi_i(t_{i+j}) = \hat{t}_{i+j}$ (for $j = 0, 1$) forms a genuine re-parameterization in case of ε-uniform samplings (with $\varepsilon > 0$). The above theorem also formulates sufficient conditions for admissible samplings (1) (including the case $\varepsilon = 0$) guaranteeing ψ_i to render a re-parameterization.

We experimentally verify here *the sharpness* of the asymptotics established in [4] and [5]. Various numerical tests conducted in this paper confirm, at least for the examined curves in $\mathbb{R}^2$ or $\mathbb{R}^3$ and samplings (12), (13) or (14) the sharpness of asymptotics claimed by both Th. 4 (see (9) and (10)) and Th. 5 (see (11)). Though all discussed herein results refer to the dense reduced data Q_m, high convergence orders yield in practice satisfactory approximation on sparse data.

Thus as $\alpha(\lambda) = 3$ for either $\lambda = 1$ or $\varepsilon \geq 1$ and $\lambda \in [0, 1]$, we should expect for sufficiently sparse Q_m (but not too dense) a good performance in $\gamma \approx \hat{\gamma}_2$.

A possible extension of this work is to study other smooth interpolation schemes [9] combined with reduced data Q_m and exponential parameterization (6) - see [11]. Certain clues may be given in [15], where complete C^2 splines are dealt with for $\lambda = 1$, to obtain the fourth orders of convergence in length estimation. The analysis of C^1 interpolation for reduced data with cumulative chords (i.e. again with $\lambda = 1$) can additionally be found in [6] or [16]. More discussion on *applications* (including *real data* examples - see [1]) and *theory* of *non-parametric interpolation* can be found e.g. in [6], [10], [11] or [12].

References

1. Janik, M., Kozera, R., Kozioł, P.: Reduced data for curve modeling - applications in graphics, computer vision and physics. Advances in Science and Technology 7(18), 28–35 (2013)
2. Noakes, L., Kozera, R., Klette, R.: Length estimation for curves with different samplings. In: Bertrand, G., Imiya, A., Klette, R. (eds.) Digital and Image Geometry. LNCS, vol. 2243, pp. 339–351. Springer, Heidelberg (2002)
3. Noakes, L., Kozera, R.: Cumulative chords piecewise-quadratics and piecewise-cubics. In: Klette, R., Kozera, R., Noakes, L., Weickert, J. (eds.) Geometric Properties of Incomplete Data. Computational Imaging and Vision, vol. 31, pp. 59–75. Kluver Academic Publishers, The Netherlands (2006)
4. Kozera, R., Noakes, L.: Piecewise-quadratics and exponential parameterization for reduced data. Applied Mathematics and Computation 221, 620–639 (2013)
5. Kozera, R., Noakes, L.: Exponential parameterization and ϵ-uniformly sampled reduced data (submitted)
6. Kozera, R.: Curve modeling via interpolation based on multidimensional reduced data. Studia Informatica 25(4B-61), 1–140 (2004)
7. Noakes, L., Kozera, R.: More-or-less uniform samplings and lengths of curves. Quarterly of Applied Mathematics 61(3), 475–484 (2003)
8. Ralston, A.: A First Course in Numerical Analysis. Mc-Graw Hill (1965)
9. de Boor, C.: A Practical Guide to Splines. Springer, Heidelberg (2001)
10. Kocić, L.M., Simoncelli, A.C., Della Vecchia, B.: Blending parameterization of polynomial and spline interpolants. Facta Universitatis (NIŠ), Series Mathematics and Informatics 5, 95–107 (1990)
11. Kvasov, B.I.: Methods of Shape-Preserving Spline Approximation. World Scientific Publishing Company, Singapore (2000)
12. Piegl, L., Tiller, W.: The NURBS Book. Springer, Heidelberg (1997)
13. Lee, E.T.Y.: Choosing nodes in parametric curve interpolation. Computer-Aided Design 21(6), 363–370 (1987)
14. Wolfram Mathematica 9, Documentation Center, `reference.wolfram.com/mathematica/guide/Mathematica.html`
15. Floater, M.S.: Chordal cubic spline interpolation is fourth order accurate. IMA Journal of Numerical Analysis 26, 25–33 (2006)
16. Kozera, R., Noakes, L.: C^1 interpolation with cumulative chord cubics. Fundamenta Informaticae 61(3-4), 285–301 (2004)

Searching in the Structured Space of the Braille Music

Wladyslaw Homenda[1] and Mariusz Rybnik[2]

[1] Faculty of Mathematics and Information Science, Warsaw University of Technology
ul. Koszykowa 75, 00-662 Warsaw, Poland
[2] Faculty of Mathematics and Computer Science, University of Bialystok
ul. Sosnowa 64, 15-887 Bialystok, Poland
homenda@mini.pw.edu.pl, mariuszrybnik@gmail.com

Abstract. This paper presents a method for searching the Braille music information based on the semantic description. The searching method is based on comparison of node's metadata that aggregates subtree's structure of information. Methods for the metadata generation is specific to searching orientation and grammar. We focus on methods for notes' durations allocation as well as key and time distribution that are required in searching and rhythm analysis. We propose a syntax analysis method based on semantic analysis using attribute grammar.

Keywords: syntactic structuring, semantic mapping, data understanding, music representation.

1 Introduction

Exploring knowledge encapsulated in the information space needs structuring and agile searching methods. In this paper we remind structuring issues connected with the music information space. Then we propose methods for semantic analysis during parsing process which use attribute grammars. The further part is devoted to a method for semantic searching which uses information incorporated in derivation tree.

Any intelligent processing of natural communication languages requires uncovering structures of raw data. There are different ways leading to structuring. Our interest is focused on employing syntactic and semantic structuring of music information with an emphasis on music notations. It is obvious that raw data without any structuring is of no use in intelligent communication. Otherwise the processing covers some characters from alphabet without any meaning. The aim is to create a generic method to integrate syntactic and semantic structuring of music information. This structuring allows for optimized processing of music information described in different languages including Braille music notation and printed music notation. In this study we bind Braille music notation to printed music notation.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 442–452, 2013.

1.1 Syntax

Syntactic structuring of music information is the first stage of the analysis process. We utilize context-free grammars for syntactic structuring of Braille music notation and printed music notation. We continue the discussion of syntactic structuring outlined in [2–5].

Let us recall that we use formal grammars, which are systems $G = (V, T, P, S)$ where: V is a finite set of nonterminal symbols (nonterminals), T is a finite set of terminal symbols (terminals), P is a finite set of productions and S is the initial symbol of grammar, $S \in V$. In general productions can be seen as a finite binary relation $P \subset (V \cup T)^{+} \times (V \cup T)^{*}$. A grammar G is context-free one (CFG) $\iff (\forall p)(p \in P \Rightarrow p \in V \times (V \cup T)^{*})$.

Attribute grammar is a context free grammar allowing to tie semantics to syntax during syntax analysis. Attribute grammar attaches so called attributes to each of the nonterminals of the grammar. Such elements are divided into two disjoint sets: synthesized and inherited attributes.

Below a subset of productions for Braille music is presented ([6]). Because of the clarity attributes definitions are omitted. The whole set is specified in [4]

$$\begin{array}{ll}
<\!S\!> & \rightarrow\ <\!music\_desc\!>\ new\_line\ <\!music\!> \\
<\!music\!> & \rightarrow\ <\!part\_desc\!>\ new\_line\ <\!part\!>\ new\_line\ <\!music\!>\ | \\
 & \rightarrow\ <\!part\_desc\!>\ new\_line\ <\!part\!> \\
<\!part\_desc\!> & \rightarrow\ <\!instruments\_info\!>\ new\_line\ <\!key\_sig\!><\!time\_sig\!> \\
<\!part\!> & \rightarrow\ <\!staff\!>\ new\_line\ <\!part\!>\ |\ <\!staff\!>\ new\_line \\
<\!staff\_with\_h\!> & \rightarrow\ <\!measure\!><\!barline\!><\!staff\_with\_h\!>\ |\ <\!measure\!><\!barline\!> \\
<\!measure\!> & \rightarrow\ <\!key\_sig\!><\!key\_sig\!><\!time\_sig\!>\ space\ <\!measure\_fm\!>\ | \\
 & \rightarrow\ <\!time\_sig\!>\ space\ <\!measure\_fm\!>\ | \\
 & \rightarrow\ <\!key\_sig\!><\!key\_sig\!>\ space\ <\!measure\_fm\!>\ |\ <\!measure\_fm\!> \\
<\!measure\_fm\!> & \rightarrow\ <\!voice\!>\ voice\_div\ <\!measure\_fm\!>\ |\ <\!voice\!> \\
<\!voice\!> & \rightarrow\ <\!time\_element\!><\!voice\!>\ |\ <\!time\_element\!> \\
<\!time\_element\!> & \rightarrow\ <\!note\!><\!time\_element\!>\ |\ <\!note\!> \\
<\!note\_gf\!> & \rightarrow\ <\!octave\!><\!note\_gfo\!>\ |\ <\!note\_gfo\!> \\
<\!note\_gfo\!> & \rightarrow\ <\!pitch\!>\ |\ <\!interval\!>\ |\ <\!pitch\!><\!prolongation\!> \\
<\!pitch\!> & \rightarrow C1\,|\,C2\,|\,C4\,|\,C8\,|\,D1\,|\,D2\,|\,D4\,|\,D8\,|\,E1\,|\,E2\,|\,E4\,|\,E8\,| \\
 & \rightarrow F1\,|\,F2\,|\,F4\,|\,F8\,|\,G1\,|\,G2\,|\,G4\,|\,G8\,|\,A1\,|\,A2\,|\,A4\,|\,A8\,| \\
 & \rightarrow H1\,|\,H2\,|\,H4\,|\,H8\,|\,rest1\,|\,rest2\,|\,rest4\,|\,rest8
\end{array}$$

Since there is no evidence that Braille music notation is a context-free language, we do not attempt to construct a context-free grammar generating the language of Braille music notation. Instead we use context-free grammars covering the language of Braille music notation. Such grammars will generate all constructions of Braille music notation and some others, which are not valid Braille music constructions. This approach cannot be used to generate Braille music scores or parts neither to check their correctness. However, since we employ context-free grammars for processing scores, which are assumed to be correct, this approach is proven. A discussion on construction of context-free grammars covering printed and Braille music notations is outlined in [5].

The notion of *lexicon* is a fundamental concept of syntax analysis. The concept of lexicon for music notation was introduced in [2] and then was developed in [4, 5]. *Lexicon* is the space of language constructions, each of them supplemented with possible *derivation trees*, also known as *parsing trees*. *Lexicon* includes relations between items of this space. Such a tree satisfies the following rules: a) it is a subtree of the derivation tree of the whole score, b) it is the minimal tree generating the given language construction, c) the minimal tree can be extended by a part of the path from the root of this tree toward the root of the score derivation tree. Due to the last condition, usually there are many trees for a given language construction.

1.2 Semantics

Languages allow to describe a real world of things, sensations, thoughts, ideas etc. Braille music notation describes the space of hearing sensations, which can be outlined as the space $B \times D \times P$ of triples (b, d, p). Each triple defines the performed sound, where b is the beginning time, d is the duration and p is the pitch of the sound. In general, objects of the real world may be outlined with much richer set of features, but this simple triple is sufficient for our discussion, c.f. [5].

Above-mentioned approach is very generic, it refers to physical essence of a sound and has not any links to a particular type of music description. This structure can be used for any music notation, especially for Braille music notation. The definition of the hearing sensation space is also very useful in case of other structural operations, e.g. *conversion* between different types of music description, c.f. [5].

The purpose of using the world of real objects is to tie meaning to syntactic structures of music descriptions. The idea of collaborating syntactic and semantic has found the practical application to processing of music information. It has been involved in a real processing of Braille music accomplished in the Braille Score project, c.f. [7].

As mentioned in the previous section, descriptions of music notation expressed in different languages and representation of music notation in different formats are cast on the world of hearing sensations. Such casts are called semantics of descriptions and representations of music information. Formally, let B is the lexicon of Braille music notation and H is the space of hearing sensation. Semantics S is a relation:

$$S \subset B \times H$$

The space of language constructions is immersed in the space of sounds. The immersion gives values of real world to language constructions. The immersion defines meaning of language constructions, defines semantics.

2 Music Data Parsing

Parsing, also known as syntactic analysis, is aimed at structuring raw music data. The result of this process is parsing (derivation) tree compatible with specified grammar. Moreover, the attribute grammar produces semantic data attached to the tree's nodes.

2.1 Key and Time Signature Passage

All of the nonterminals in Braille music grammar are equipped with attributes connected with *key* and time signature. The knowledge of the *key* and time (as synthesized attributes) is discovered in productions and passed to the top of the tree, to the initial nonterminal:

$<fifths>\rightarrow \sharp \,|\, \flat \,|\, \natural \,|\, \sharp\sharp \,|\, \flat\flat \,|\, \natural\natural \,|\, \sharp\sharp\sharp \,|\, \flat\flat\flat \,|\, \natural\natural\natural \,|\, etc.$
$<fifths>\ .fifth = id$

$<time\_sig>\rightarrow num\_ind\ <beat>\ <beat\_type>$
$<time\_sig>\ .beat = (<beat>\ .beatH, <beat\_type>\ .beatL)$

$<beat>\rightarrow higher\_1 \,|\, higher\_2 \,|\, etc.$
$<beat>\ .beatH = id$

$<beat\_type>\rightarrow lower\_1 \,|\, lower\_2 \,|\, etc.$
$<beat\_type>\ .beatL = id$

Key and *time signature* attributes are passed from the root of the tree to the other structures of the notation as inherited attributes. Each of the productions has a rule similar to this:

$<part_1>\rightarrow<staff>\ new\_line\ <part_2>\ \mid\ <staff>\ new\_line$
$<staff>\ .beat =<part_1>\ .beat$
$<part_2>\ .beat =<part_1>\ .beat$

This method allows to distribute *key* and *time signature* info in structured space of music. *Key* and *time* knowledge flows from *key/time signature* branch to the root of the tree and then it is propagated to the lower parts of the tree.

2.2 Notes' Duration Identification and Passage

In case of Braille music notation, because of limited number of available characters, notes of the duration x and $\frac{x}{16}$ share the same symbol, e.g. *whole note C* and *sixteenth note C* are represented by the same Braille cell. This ambiguity demands contextual analysis to investigate the correct duration of the note.

This issue is a nondeterministic problem and it involves voices, because voices are fully filled in time dimension.

Simulation Tree Method. One of the practical solutions of this nondeterministic issue is simulation tree. We assume, that the sound of each note lasts from the beginning of the note to its end, i.e. no articulation, ornamentation or dynamic has influence on the duration of the sound. This tree is defined as follows (lets be $N = (n_0, n_1, ..., n_k)$ - notes in voice and $S = (s_0, s_1, ..., s_{k+1})$ - time slots between neighboring notes, i.e. s_0 is the time when first note starts, s_1 – the time when first note ends and the second starts, etc.:

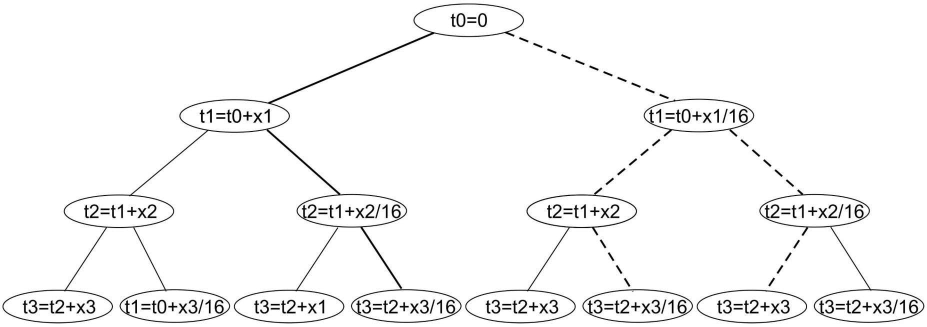


Fig. 1. Simulation tree draft for calculating duration

- nodes of the tree are labeled with elements from S
- root of the tree is labeled with s_0
- each node labeled with s_x has two children labeled with s_{x+1}, that represent two variants of the note duration n_x

The draft of simulation tree is given in the Fig. 1.

Such definition describes all possible time allocations. Each of the leaf of the simulation tree represents unique time allocation in the voice, because leaf in the tree is labeled by s_{k+1} - the space after last note in the voice. Incorrect time fillings are easily detectable. Braille music definition demands unambiguous situations in case of notes duration, i.e. there is only one path from the root to the leaf in the simulation tree that is filled in correct way. Fig. 1 presents simulation tree for 3 notes. Let assume that notes may represent the same durations, e.g. C, D, and E whole/sixteenth notes (whole/sixteenth are described by the same Braille cell), and time signature has a proper value. This is a case that fits the scheme ,,one left and two rights", which leads to three concretizations: left-right-right, right-left-right and right-right-left (marked in the Fig. 1 as bold solid and dotted lines). Each of above mentioned filling schemes fills voice in correct manner but only one of them is desired and it is left-right-right (marked as bold, solid line), because in other case there should appear special character after s_1(in the right part of the tree, where starts two dotted, bold lines; Braille specific). That character is to mark explicitly the duration of the note. Appearance of such Braille signs excludes some durations of the note and causes the tree to loose some paths. All paths are present in the tree in the Fig. 1, so there is no duration mark in the notation.

Attribute Flow Method. We propose method that bases on attribute grammar. Attribute grammars for music notation was introduced in [3]. In this paper we use concepts form this paper to resolve the problem of time/duration ambiguity of Braille music notation. In fact this method creates implicit simulation tree during syntax analysis.

Attribute flow method requires two stages (two attributes flows) during processing. Also we assume that all nonterminals located below ,,voice" element in

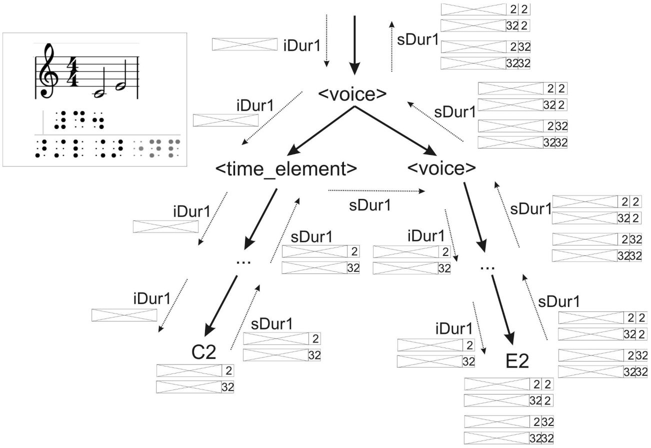


Fig. 2. Attribute flow during the first stage for production
$<\!voice\!> \rightarrow <\!time\_element\!> <\!voice\!>$

derivation tree are supplemented with four attributes: two synthesized (*sDur1* and *sDur2*) and two inherited (*iDur1* and *iDur2*).

First stage. During first stage the attribute *iDur1* is passed down to the leaf, change in *sDur1* is attempt according to semantic rule, *sDur1* is passed toward the root to the ,,time_element" nonterminal, it changes *iDur1*, then *iDur1* is passed down to the leaf, ... This procedure lasts till the end of the notes in current voice. The example of attribute flow is presented in the Fig. 2

It allows to collect all possible variants of voice's time allocation. After that phase, when *sDur1* is synthesized in ,,voice" nonterminal, the correct variant is chosen. According to remark in 2.2 paragraph, thanks to Braille music definition, there exists one unique time allocation. This one particular solution is assign to *iDur2* attribute of ,,voice" nonterminal.

$$\begin{aligned}
&<\!note\_gfo\!> \rightarrow <\!high_1\!> \mid <\!interval\!> \mid <\!high_2\!> <\!prolongation\!>\\
&<\!high_1\!>.iDur1 = <\!note\_gfo\!>.iDur1\\
&<\!note\_gfo\!>.sDur1 = <\!high_1\!>.sDur1\\
&<\!high_2\!>.iDur1 = <\!note\_gfo\!>.iDur1\\
&<\!prolongation\!>.iDur1 = <\!high_2\!>.sDur1\\
&<\!note\_gfo\!>.sDur1 = <\!prolongation\!>.sDur1
\end{aligned}$$

$$\begin{aligned}
&<\!high\!> \rightarrow Nx\text{ , where } Nx = (N,x) \in \{C,D,E,F,G,A,H,rest\} \times \{1,2,4,8\}\\
&<\!high\!>.sDur1 = (<\!high\!>.iDur1 \cup \{x\}, <\!high\!>.iDur1 \cup \{16x\})
\end{aligned}$$

$$\begin{aligned}
&<\!prolongation_1\!> \rightarrow * <\!prolongation_2\!>\\
&<\!prolongation_2\!>.iDur1 = <\!prolongation_1\!>.iDur1 \cup \{\tfrac{1}{2}\}\\
&<\!prolongation_1\!>.sDur1 = <\!prolongation_2\!>.sDur1
\end{aligned}$$

$<\!prolongation\!> \rightarrow *$
$<\!prolongation\!>.sDur1 = <\!prolongation\!>.iDur1 \cup \{\frac{1}{2}\}$

$<\!measure\_fm\!> \rightarrow <\!voice\!>\ voice\_div\ <\!measure\_fm\!>\ |\ <\!voice\!>$
$<\!voice\!>.iDur1 = ()$
$<\!voice\!>.iDur2 = \text{chooseCorrectTimeAllocation}(<\!voice\!>.sDur1, <\!voice\!>.beat)$

$<\!time\_element_1\!> \rightarrow <\!note_1\!>\ <\!time\_element_2\!>\ |\ <\!note_2\!>$
$<\!note_2\!>.iDur1 = <\!time\_element_1\!>.iDur1$
$<\!time\_element_1\!>.sDur1 = <\!note_2\!>.sDur1$
$<\!time\_element_2\!>.iDur1 = <\!time\_element_1\!>.iDur1$

$<\!voice_1\!> \rightarrow <\!time\_element_1\!>\ <\!voice_2\!>\ |\ <\!time\_element_2\!>$
$<\!time\_element_1\!>.iDur1 = <\!voice_1\!>.iDur1$
$<\!voice_2\!>.iDur1 = <\!time\_element_1\!>.sDur1$
$<\!voice_1\!>.sDur1 = <\!voice_2\!>.sDur1$
$<\!time\_element_2\!>.iDur1 = <\!voice_1\!>.iDur1$
$<\!voice_1\!>.sDur1 = <\!time\_element_2\!>.sDur1$

Second stage After the last note, the correct time allocation is chosen and is propagated in the tree. It uses *iDur2* and sDur2 attributes in the same way the *iDur1* and *sDur1* were used in the first stage. In this phase each note cuts one proper time duration and assigns to itself. After the second stage, the notes have allocated correct durations.

General Remarks. Both methods are practical solutions of the nondeterministic problem of the note's time allocation. Simulation tree and attribute flow method are equivalent. In fact, the attribute flow method is implicit simulation tree method. It creates simulation tree during the process of generation of the all possible time allocations. Simulation tree is ,,orthogonal" to derivation tree, as it is shown in the Fig. 3.

There is a possibility to optimize proposed method of attribute flow by saving memory space. Optimization requires change in semantic rule for $<\!high\!>$ production. The method will collect only one sign for each note during the first attribute flow (instead of duplicating the previous formulas):

$<\!high\!> \rightarrow Nx$, where $Nx = (N, x) \in \{C, D, E, F, G, A, H, rest\} \times \{1, 2, 4, 8\}$
$<\!high\!>.sDur1 = <\!high\!>.iDur1 \cup \{x\}$

After applying the changes above the simulation tree is created in function ,,choose" and still allows to investigate the one, possible solution. It allows however to save memory during attribute flow, as original variant has space complexity $O(|N| \cdot 2^{|N|})$ and optimized method has space complexity $O(|N|)$, where $|N|$ is the number of notes in voice.

The mentioned methods are applicable even if there is a need to check $2^{|N|}$ possible time allocations because of the fact that one voice does not contain huge numbers of notes. The average number of the notes in the one voice is about 10.

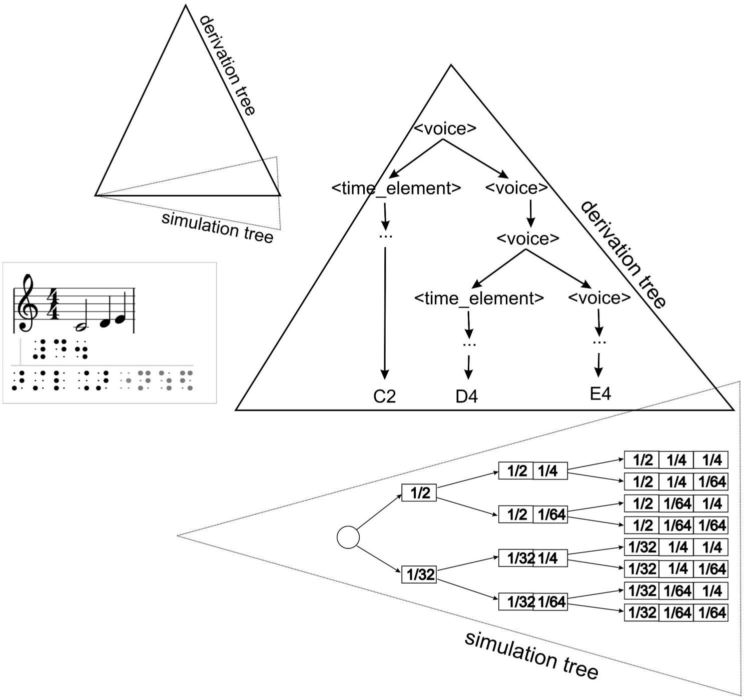


Fig. 3. Simulation and derivation tree in the attribute flow method

3 Selection and Semantic Searching Operation Performed on the Music Notation

This paragraph contains a description of selection and searching operations. We discuss here a different types of grammars and their influence on selection performed in the structured space of music. Selection is a prelude to searching operation.

In this case we focused on rhythmic analysis preceded by notes' durations allocation. In previous paragraph we showed method for duration allocation as well as *key* and time signature distribution, which are crucial for rhythmic analysis in Braille music notation. Other types of searching information are easy to perform in similar way – as attributes flow during syntax analysis or in the space of sounds.

3.1 Different Types of Grammars and Selection Operation

Music information is a data that can be presented in many different formats, e.g. printed music, Braille music, MIDI, etc. Each type is described by specified grammar, in this case they are approximated by context-free grammars. Moreover, in frames of one format there exist many grammars. Each of these grammars defines the same language (data format), but exposes different property of the language. The examples of such grammars are given in [3], where authors introduced two grammars for printed music: ,,time-prior-to-pitch" and ,,pitch-prior-to-time". [3] illustrates selection performed in the structured space of printed music notation.

Selection made in the notation space leads to selecting some nodes in the derivation tree. The structure of selected nodes in the tree differs depending on the grammar type. Some grammars make nodes to be consistent, i.e. there is some node that is an ancestor for all selected nodes and that node does not have unselected children as descendant. In other words, there exists one node, that derived all and only selected notation.

Some language has natural orientation, e.g. Braille music is voice oriented, i.e. selection of the notes in one voice is going to generate consistent selection in derivation tree. Other orientations for Braille music are also possible. In case of the printed music, in the paper [3] were given grammars: pitch and time oriented.

3.2 Semantic Searching

Semantic search allows to search for all occurrences of notation described in semantic way, e.g. all fifth accords. Syntax searching, where algorithm looks for the exact occurrence of the searching text, is a special case of semantic search.

During searching operation the lexicon is used.

Let us assume that grammar type and selection operation generate consistent selection of nodes in the derivation tree. For each lexicon's element notation is described as data put in the root of the subtree. That description should be connected with the type of grammar and searching operation. Method for accord description, in case of interval analysis, were given by Allan Forte in [1]. Descriptions are applicable during syntax analysis thanks to attribute flow.

To search specified notation described by some semantic, all that should be done is searching nodes in the whole derivation tree. Lexicon contains many elements for one piece of notation. The difference between these elements is in subtrees – each subtree has different height. The bigger height, the more precise description it represents. Searching for a *note* which description is aggregated in nonterminal ,,<*note*>" means that any of such *notes* in the whole measure meets the condition. If searching for a *note* involves nonterminal ,,<*measure*>" it means that all occurrences of specified *note* in that particular time slot in all *measures* of the score are desired. Finally, nonterminal ,,<S>" defines exactly one occurrence of the searching *note*.

Semantic search requires semantic which is delivered during syntax analysis as floating attributes or as the world of hearing sensation (thanks to semantic mappings).

3.3 Semantic Searching Example

Fig. 4 presents an example of semantic searching in the structured space of Braille music. The task is to find all occurrences of the accords that have the same intervals and duration. Please note that Braille music allows to denote accords in more than one manner (c.f. [6]), so our task is not an usual (syntax) search. Grammar presented in [4] allows for such operation. We tie with all ,,$<$ *time_element* $>$" nonterminals a vector that specifies desired information about derived structures. That activity can be done after allocation of notes' duration and *key/time signatures* passage in the derivation tree. The vector is labeled as $[d; x_1 - x_2 - ... - x_n]$ where d is the note (accord) duration, $n+1$ is the number of noteheads in the accord, x_k is the interval between $(k-1)$-th and k-th noteheads.

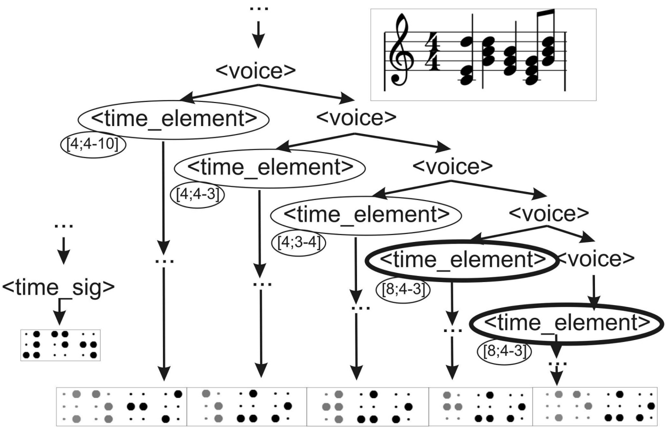


Fig. 4. Semantic Searching Example

Let us assume that we want to find all *eighths* accords built from intervals 4 and 3, so the vector is $[8; 4-3]$. To fulfill this task all that should be done is vector (,,<*time_element*>" nonterminal metadata) comparison. No syntax of the score is touched. The method points two occurrences, marked in the Fig. 4 with bold ellipses. The result of the operation is a piece of notation that is derived from the pointed nonterminals marked with bold ellipses.

4 Conclusions

This paper presents approach for searching in the structured information space of Braille music. Searching is understood as localization of specified node in the derivation tree, which corresponds to node of the subtree paired with the searched notation in the lexicon space. Our approach is rhythmic oriented, i.e. we investigate mutual time dependencies between notes. We needed to allocate duration for the notes to cope with that.

The direct expansion can be aimed at providing another methods for aggregating data at the subtree's root. The important issue is to connect with intelligent human-computer interface developing because conditions imposement is a bottleneck for the searching in the structured space of music data.

Acknowledgment. The research is supported by the National Science Center, grant No 2011/01/B/ST6/06478, decision no DEC-2011/01/B/ST6/06478.

References

1. Forte, A.: The Structure of Atonal Music. Yale University Press, London (1973) ISBN 0-300-01610-7 (cloth) ISBN 0-300-02120-8 (pbk)
2. Homenda, W., Rybnik, M.: Querying in Spaces of Music Information. In: Tang, Y., Huynh, V.-N., Lawry, J. (eds.) IUKM 2011. LNCS (LNAI), vol. 7027, pp. 243–255. Springer, Heidelberg (2011)
3. Homenda, W., Rybnik, M.: Knowledge-Driven Syntactic Structuring: The Case of Multidimensional Space of Music Information. In: Liddle, S.W., Schewe, K.-D., Tjoa, A.M., Zhou, X. (eds.) DEXA 2012, Part I. LNCS (LNAI), vol. 7446, pp. 438–452. Springer, Heidelberg (2012)
4. Homenda, W., Sitarek, T.: Performing Operations on Structured Information Space of Braille Music. In: König, A., Dengel, A., Hinkelmann, K., Kise, K., Howlett, R.J., Jain, L.C. (eds.) KES 2011, Part IV. LNCS (LNAI), vol. 6884, pp. 232–241. Springer, Heidelberg (2011)
5. Homenda, W., Sitarek, T.: Notes on automatic music conversions. In: Kryszkiewicz, M., Rybinski, H., Skowron, A., Raś, Z.W. (eds.) ISMIS 2011. LNCS (LNAI), vol. 6804, pp. 533–542. Springer, Heidelberg (2011)
6. Krolick, B.: How to Read Braille Music, 2nd edn. Opus Technologies (1998)
7. Grant no N R02 0019 06/2009, Breaking accessibility barriers in information society. Braille Score - a computer music processing for blind people, Institute for System Research, Polish Academy of Sciences, report, Warsaw (2011)

Solving Steel Alloying Using Differential Evolution and SOMA

Michal Holiš, Lenka Skanderová, Martin Plaček, Jiří Dvorský, and Ivan Zelinka

Department of Computer Science, VSB – Technical University of Ostrava,
17. listopadu 15, 708 33 Ostrava – Poruba, Czech Republic
{michal.holis,lenka.skanderova,martin.placek,
jiri.dvorsky,ivan.zelinka}@vsb.cz

Abstract. This paper proposes method for solving steel alloying problem using evolution algorithms SOMA and differential evolution. Both algorithms belong to the family of the evolution algorithms but the main ideas of these algorithms are different. In differential evolution new offspring is created during the evolution, the individuals are crossed and mutated, while in SOMA the individuals move in the space of the possible solutions and the mutation is replaced by perturbation. The main goal of this paper is to discover how much these algorithms are usable and suitable to solve the problem of steel alloying.

Keywords: steel alloying, evolutionary algorithms, Differential evolution, SOMA.

1 Introduction

1.1 Steel Alloying

It takes many steps to manufacture steel and every factory's step configuration may vary from the others. Commonly the process follows path similar to this: iron ore is smelted in iron smelters and distributed as a hot metal with temperature about 1400°C to converter or electric arc furnace to improve its quality for following treatments. Hot metal is mixed with scrap and slag formers, the heat is heated up approximately to 1650°C and first ferro-alloys are charged into the heat during tapping. This point is the first appliance of alloying process described below. Ladle with the heat is going to secondary metallurgy process, typically ladle furnace or vacuum treatment. This treatment part is exceptionally skipped, otherwise the steel quality is improved with accurate amount of alloys to achieve requested steel composition defined by steel grade and the temperature is modified for the last step. The final step of this process is casting on continuous casting machines to obtain final steel product like slabs, blooms or billets. More on this process can be found for example in book by Ahindra Ghosh and Amit Chatterjee [12].

Target of the whole process is primarily to manufacture steel of requested quality (this is commonly referred to as target or final steel grade), secondly to reduce cost of the materials and energy used in the process.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 453–464, 2013.

To manufacture steel of certain grade, with specific chemical, electrical and mechanical properties, it is crucial to be sure that the steel is composed of correct elements in correct ratio's. Composition of the steel is modified in each step throughout the whole process with technique known as alloying.

Alloying is process of continually charging the steel with certain amount's of alloying materials (compounds of chemical elements) to gradually obtain the requested steel quality. Each of the materials has certain given ratio of all elements it is composed of. Also price of the material is known. Question is how to optimally choose amount of charged materials in each step when there are many charging materials composed of many chemical elements, with one element being commonly present in many materials. We also have to consider price of the final solution, since some materials are very cheap and some can be very expensive.

To solve the problem we are always presented with these input data:

- weight of the input steel,
- input steel's composition vector (vector of ratios representing each element's presence in steel),
- list of alloying materials, their chemical solution and their unit prices,
- target steel grade represented with three vectors – one being optimal ratio of all chemical elements in final steel, second represents minimal ratio of any given element in final steel and last one represents maximal ratio of any given element in final steel.

Commonly in practice this problem is solved with Dantzig's simplex algorithm, which is a linear programming solution that with given input parameters finds cheapest solution in given final steel's grade range. This however does not allow us to fine-tune optimal balance between precision of final solution and it's price. This can be solved by iteratively running the algorithm with different steel grade ranges and trying to found solution that we consider to be optimal both in precision and cost. However since the ranges are vectors there are so many combinations of minimal and maximal values that it is very inefficient and unpractical to proceed in this way. More on this matter can be found in book by Dantzig, George B. [13].

In this paper, we present how to overcome problems of the common solution with usage of evolution algorithms and fitness function suitable for solution of this problem is presented.

1.2 Evolution Algorithms

The evolution algorithms have been successfully used to solve many practical and theoretical problems, see [9,15,16]. In [4] the differential evolution is used as classifier for the features in the data set, in [1] the algorithm SOMA is used to machining allocation of clutch assembly. Evolutionary techniques are applied in connection with, e.g. [2,5,6], too. These algorithms are still improved, see [7,8]. They are based on two essential principles – crossing and mutation. These principles proceed from Darwin's evolution theory and Mendel's principle of crossing. They work with the population of individuals, which are developing (improving) during the evolution. Better individuals survive while worse ones die. In this paper differential evolution and SOMA have been chosen.

Differential Evolution. Differential evolution (DE) is a population – based stochastic algorithm for global optimization. It was introduced by Ken Price and Rainer Storn and in this paper the original version is used for experiment design [11]. Although this algorithm is very simple and efficient, in [17] authors proved that it can not ensure the global convergence and they proposed two hybrids algorithms named QAISDE and GDISE to improve DE.

At the beginning of the algorithm the first population is generated. Each individual in population has its own set of parameters. The number of parameters is called dimension, we denote it D. In addition each individual has its fitness value. This value is very important because it says how good is this individual in current population. Fitness is the value of the cost function. Each parameter of the individual has low bound and high bound. It is not able to cross over these bounds [18].In this article DE/best/1/exp has been chosen.

When the first population is generated, for each individual in population following steps are done:

1. Three other neighbors are chosen randomly (these individuals may not be the same).
2. The noise vector is created, see Eq. (1).
3. New individual is created. The random number from the interval $[0, 1]$ is generated. If the number is smaller than crossing threshold CR, the parameter from the noise vector is chosen as a parameter of the new individual. Otherwise the parameter from the actual parent is taken.
4. If the new individual has better fitness than its parent, it will replace the parent, otherwise the new individual is forgotten and the parent stays in the population.

The noise vector equation of DE/best/1/exp:

$$v_j = x_{r3,j}^G + F\left(x_{r1,j}^G - x_{r2,j}^G\right). \tag{1}$$

where v_j is the noise vector, $x_{r3,j}^G$, $x_{r1,j}^G$ and $x_{r2,j}^G$ are three randomly chosen individuals and F denotes the mutation constant.

Self Organizing Migration Algorithm (SOMA). Unlike differential evolution, SOMA works on the principle of one population, which is changing in time. Application of this algorithm can be found for example in [19,20]. The beginning is the same like in DE. The first population is generated. But in SOMA no offspring is created. The principle of SOMA is the individuals are moving in the space of possible solutions and in the each generation (we say migration) they have a new positions. In [21] the novel multiobjective SOMA has been introduced.In this paper the strategy All to one has been chosen. At the begin the new population is generated. As well as in DE, each individual has its parameters and fitness, low and high bound. When we have the population the best individual is chosen (individual with the best fitness) – we call it leader. The following steps are done next for each individual in population:

1. The perturbation vector $\boldsymbol{\alpha}$ is generated according to the parameter PRT, which is usually set to 0.11. The $\boldsymbol{\alpha}$ is composed of 0 and 1 and it is important for the direction of the individual moving. (Individual can move to the leader in many directions.)
2. Individual will move to the leader. It moves by steps. For each step the next position and the new fitness is computed. Moving is finished when the parameter *Step* spreads the value of the parameter $PathLength$.
3. After migration of the individual its best position, which has been reached during the migration, is chosen.
4. If the best reached position of the individual is better than the original one, the original one is replaced by the new position.

The crossing is replaced by philosophy that each individual moves in the space and each individual remembers the coordinates of the position, where the resolution has been the best in scope of its way to the leader. Moving of the individual is directed by the equation below [10].

$$x_{i,j}^{ML+1} = x_{i,j,start}^{ML} + \left(x_{L,j}^{ML} - x_{i,j,start}^{ML}\right) t\boldsymbol{\alpha}_j \tag{2}$$

where $x_{i,j}^{ML+1}$ is a new position of the individual, $x_{i,j,start}^{ML}$ denotes the individual's start position, $x_{i,j,start}^{ML}$ is the actual leader's position, *Step* is marked as t and $\boldsymbol{\alpha}$ denotes the value in j^{th} position of $\boldsymbol{\alpha}$.

2 Experiment Design

Our method is based on SOMA and DE using in steel alloying process. There were two main problems in experiment design - how to represent an individual in evolutionary algorithms and how to define its fitness function. From the view of steel alloying process, there were 14 compounds, which are added to the original steel composition. These compouds are consist of the elements. In result we needed the elements amount. In evolution algorithms the individuals consist of parameters. In this paper the parameters of individuals are represented by the compounds, which were added to the original steel composition. There were 14 compounds, so the individual's dimension (number of parameters) D has been 14. Each compound has been represented by one parameter in individual. In the end of the evolution the amount of the elements have been recomputed from the amount of compounds.

Fitness function presented in this paper is designed in the following manner:

– parameters of each member in population represent weight of each alloying material that is to be charged to the steel (every member has as many parameters as there are alloying materials),

- first the function computes weight of each element in input steel using it's weight and it's composition vector,
- using the value of each parameter of current member and composition vector of material it represents we can compute the weight of each chemical element that would be added to the steel, if materials the member represents would be charged to the steel,
- using weights of elements in original steel and computed weights of elements in charged materials the final presence of each material in final steel is computed,
- value of the fitness function is then equal to absolute distance of the optimal solution vector and this solution's vector, if this vector is out of range represented by minimal and maximal vector of solution, then penalization is applied to the fitness value,
- additionally price of all used materials is computed and added to the fitness value, to do this constant defining importance of price of final solution over it's accuracy is applied, with this constant we can control and fine-tune the solution to our specific needs.

2.1 Fitness Function Equation

If $Cmin_j < \left| \frac{M_i \cdot Mat_{ij} + S_j \cdot W}{\sum_{k=0}^{n_m} M_k + W} - C_j \right| < Cmax_j$ then fitness function is defined as follows (this is the case when given member represents valid solution that is in required composition range).

$$\sum_{i=0}^{n_m} \left[\left(\sum_{j=0}^{n_e} \left| \frac{M_i \cdot Mat_{ij} + S_j \cdot W}{\sum_{k=0}^{n_m} M_k + W} - C_j \right| \right) + P_i \cdot M_i \cdot PW \right] \tag{3}$$

Otherwise we apply additional penalization constant to equation to discard the member from solution.

$$\left\{ \sum_{i=0}^{n_m} \left[\left(\sum_{j=0}^{n_e} \left| \frac{M_i \cdot Mat_{ij} + S_j \cdot W}{\sum_{k=0}^{n_m} M_k + W} - C_j \right| \right) + P_i \cdot M_i \cdot PW \right] \right\} \cdot T \tag{4}$$

Notation used in these equation is summarized in Table 1.

For both evolution algorithms, DE and SOMA, the same dimension has been set, $D = 14$. In DE's experiments we were changing the parameters: the number of individuals in the population NP, the number of generations G, mutation constant F, and crossing threshold CR in order to find better evaluation of the cost function. In SOMA experiments these parameters were also modified: number of the migration cycles $Migrations$, the stopping position of the migrating individual $PathLength$, and the step size of the migrating individual $Step$. All tested combinations of input parameters are given in Table 2.

From chemical point of view fourteen, $D = 14$, chemical compounds have been used: FeMn (with high carbon level), FeMn (with medium carbon level), FeMn

Table 1. The variables description from the Eqs. (3) and (4)

Variable	Description
n_m	Total number of available alloying materials (compounds) used.
n_E	Total number of chemical elements that materials consists of.
M	Individual being tested for it's fitness value. Bottom index specifies index of parameter whose value we want to obtain. Each parameter of member represents weight in kilograms (kg) of one material that would be charged to steel. Number of parameters of one member equals to number of all available alloying materials.
Mat	Material composition. First index specifies material, which's composition we are interested in. Second denotes index of element. Result is number in range $[0, 1]$ (1 means 100% of material is composed with that element, 0 that element is not contained in compound) that represents element's representation in given material (compound).
S	Original steel composition vector. Index denotes element whose representation we are interested in. Resulting number is again in range $[0, 1]$.
W	Original steel's weight in kg.
C	Desired composition vector. Index denotes element whose representation in steel we want to obtain. Resulting number is again in range $[0, 1]$.
$Cmin$	Desired minimal composition vector. Index denotes element whose representation in steel we want to obtain. Resulting number is again in range $[0, 1]$.
$Cmax$	Desired maximal composition vector. Index denotes element whose representation in steel we want to obtain. Resulting number is again in range $[0, 1]$.
P	Unit price of material for each kg.
PW	Constant defining importance of final solution's cost over it's precision.
T	Penalization constant. Very high number that is used to multiply the final fitness function's value in case the solution is outside the $[Cmin, Cmax]$ range.

(with low carbon level), FeSiMn, FeCr (with high carbon level), FeCr (with low carbon level), FeNi, FeMo, FeW, FeNb, FeV, NV, FeTi and Cu. Composition of these compounds can be found in Table 4. For each element the minimum value, maximum value and optimal value in final steel alloy have been set, see Table 3.

Table 2. Experiments setting for algorithms DE and SOMA

(a) Experiments setting for DE.

D	*NP*	*G*	*F*	*CR*
14	1000	1500	0.8	0.6
14	1000	1500	0.8	0.5
14	1000	1500	0.7	0.6
14	1000	1500	0.7	0.5
14	1000	1500	0.5	0.7
14	1000	1500	0.5	0.6
14	1000	1500	0.5	0.5
14	500	1000	0.8	0.6
14	500	1000	0.8	0.5
14	500	1000	0.7	0.6
14	500	1000	0.7	0.5
14	500	1000	0.5	0.7
14	500	1000	0.5	0.6
14	500	1000	0.5	0.5

(b) Experiments setting for DE.

D	*NP*	*G*	*F*	*CR*
14	50	2000	0.8	0.6
14	50	2000	0.8	0.5
14	50	2000	0.7	0.6
14	50	2000	0.7	0.5
14	50	2000	0.5	0.7
14	50	2000	0.5	0.6
14	50	2000	0.5	0.5
14	50	1500	0.8	0.6
14	50	1500	0.8	0.5
14	50	1500	0.7	0.6
14	50	1500	0.7	0.5
14	50	1500	0.5	0.7
14	50	1500	0.5	0.6
14	50	1500	0.5	0.5

(c) Experiments settings for SOMA ALLToOne.

D	*NP*	*Migrations*	*PathLength*	*Step*	*PRT*
14	1000	200	3	0.11	0.1
14	1000	200	3	0.3	0.1
14	1000	200	3	0.1	0.1

3 Experimental Results

Our experimental results are summarized in Table 5. Highlighted rows represent runs that were unable to reach satisfying solution. First column of the table contains description of given settings along with it's parameters. Second column contains best achieved fitness value of all runs and next columns contain information about final computed solution of the steel. Progress of the best fitness values is show in Figures 1 and 2. One hundred test runs using given setting were performed and each test run is displayed as one line in these charts. Only selection of plots of runs that were able to reach satisfying solution are presented in this paper.

Table 3. The table of the chemical elements whose amounts we need to know. The amounts are stated in % and are set for 1,000 kilograms of the steel.

Element	Values [%]		
	Minimal	Optimal	Maximal
C	0.003	0.035	0.040
Si	0.065	0.075	0.085
Mn	0.952	0.962	0.972
P	0.000	0.001	0.006
S	0.000	0.001	0.002
Cr	1.268	1.288	1.308
Ni	0.940	0.960	0.980
Mo	0.000	0.000	0.010
Cu	0.000	0.000	0.005
Nb	0.000	0.000	0.002
V	0.000	0.000	0.002
Al	0.000	0.003	0.013
Ti	0.000	0.000	0.005
N	0.000	0.000	0.005
W	0.000	0.000	0.005
Fe	96.671	96.676	96.681

Table 4. Materials and their solution used in our experiments. Column *Material Code* contains code of material and in other columns presence of each element in percents is shown.

Material Code	Element Values [%]														
	C	Si	Mn	P	S	Cr	Ni	Mo	Cu	Nb	V	Al	Ti	N	W	Fe
FeMnHC	7	6	78	0.05	0.02	0	0	0	0	0	0	0	0	0	0	8.93
FeMnMC	2	3	88	0.1	0.02	0	0	0	0	0	0	0	0	0	0	6.88
FeMnLC	0.5	1.8	90	0.05	0.02	0	0	0	0	0	0	0	0	0	0	7.63
FeSiMn	0.5	25	65	0.05	0.02	0	0	0	0	0	0	0	0	0	0	9.43
FeCrHC	7	1.5	0	0.03	0.06	65	0	0	0	0	0	0	0	0	0	26.41
FeCrLC	0.05	1.5	0	0.03	0.02	67	0	0	0	0	0	0	0	0	0	31.4
FeNi	0.02	0	0	0	0	0	99.8	0	0	0	0	0	0	0	0	0.18
FeMo	0.05	0.8	0	0.05	0.1	0	0	60	0	0	0	0	0	0	0	39
FeW	0.1	0.8	0.2	0.03	0.02	0	0	6	0.1	0	0	4	0	0	80	8.75
FeNb	0.1	2	0	0.08	0.08	0	0	0	0	65	1.5	0.8	0	0	0	30.44
FeV	0.2	1.5	5	0.05	0.05	0	0	0	0	0	60	0	0	0	0	33.2
NV	5	0.2	0.5	0.05	0.5	0.2	0	0	0.15	0	75	0.3	0	16	0	2.1
FeTi	0.2	0.5	0.1	0.05	0.05	0	0	0	0	0	0	5	70	0	0	24.1
Cu	0	0	0	0	0	0	0	0	99	0	0	0	0	0	0	1
Al pigs	0	0.15	0	0	0	0	0	0	0.5	0	0	99.2	0.04	0	0	0.11
Al pigs	0	1	0	0	0	0	0	0	0	0	0	97	0	0	0	2
C	98	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2

From the results measured in our experiments we can see, that by using higher values then 0.5 for F in DE the evolution requires higher number of generations to successfully reach valid solution. With $F = 0.5$ we were able to reach satisfying solution even with only 50 members in whole population. Although some of the runs were unsuccessful and were unable to present valid solution, the runs that completed successfully yielded correct result reliably and we have not recorded single case when they would fail.

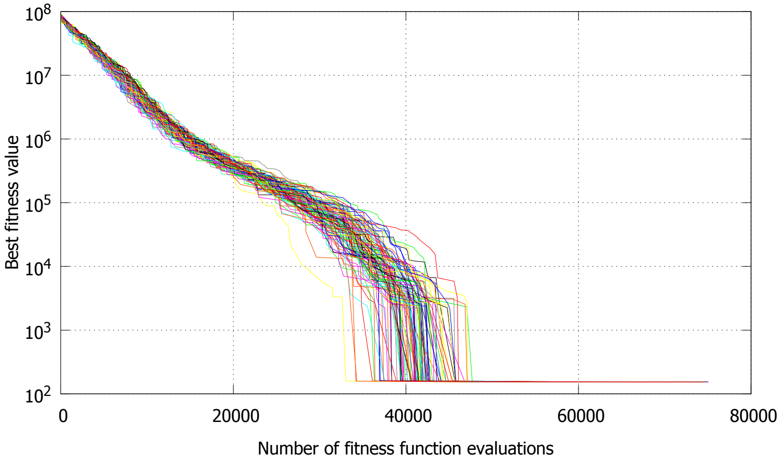


Fig. 1. Differential Evolution (NP: 50, G: 1500, F: 0.5, CR: 0.7)

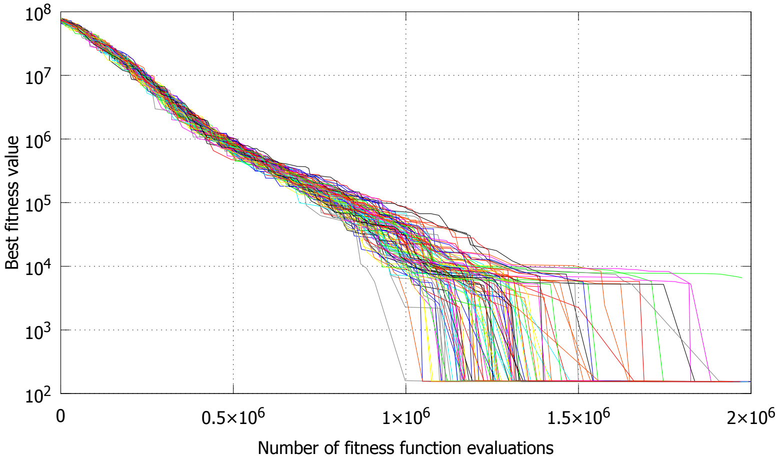


Fig. 2. SOMA (NP: 1000, $Migrations$: 200, PRT: 0.1, $Step$: 0.3)

Table 5. Experimental results

Settings	Fitness	Alloying elements [%]															
		C	Si	Mn	P	S	Cr	Ni	Mo	Cu	Nb	V	Al	Ti	N	W	Fe
DE (1,000, 1,500, 0.5, 0.5)	153.518	0.040	0.083	0.969	0.001	0.001	1.270	0.943	0.000	0.001	0.000	0.001	0.011	0.000	0.000	0.000	96.680
DE (1,000, 1,500, 0.5, 0.6)	153.578	0.039	0.083	0.972	0.001	0.001	1.269	0.942	0.000	0.003	0.000	0.001	0.008	0.002	0.000	0.000	96.680
DE (1,000, 1,500, 0.5, 0.7)	153.689	0.040	0.085	0.957	0.001	0.001	1.273	0.949	0.000	0.003	0.000	0.002	0.010	0.000	0.000	0.000	96.680
DE (1,000, 1,500, 0.7, 0.5)	15,789.470	0.042	0.084	0.958	0.001	0.001	1.273	0.940	0.000	0.000	0.007	0.003	0.012	0.002	0.000	0.004	96.674
DE (1,000, 1,500, 0.7, 0.6)	100,347.300	0.055	0.083	0.918	0.001	0.001	1.274	0.948	0.003	0.014	0.002	0.003	0.014	0.007	0.000	0.004	96.674
DE (1,000, 1,500, 0.8, 0.5)	146,710.100	0.050	0.091	0.953	0.001	0.001	1.287	0.933	0.009	0.007	0.002	0.008	0.007	0.004	0.001	0.022	96.624
DE (1,000, 1,500, 0.8, 0.6)	334,186.400	0.075	0.105	0.947	0.001	0.001	1.195	0.916	0.007	0.015	0.003	0.026	0.028	0.013	0.005	0.009	96.656
DE (500, 1,000, 0.7, 0.5)	139,756.300	0.060	0.084	0.895	0.001	0.001	1.305	0.955	0.009	0.001	0.013	0.007	0.008	0.006	0.001	0.002	96.654
DE (500, 1,000, 0.7, 0.6)	325,398.300	0.065	0.078	0.803	0.001	0.001	1.278	0.960	0.028	0.012	0.016	0.032	0.021	0.016	0.000	0.015	96.675
DE (500, 1,000, 0.8, 0.5)	352,591.900	0.085	0.121	0.883	0.001	0.001	1.205	0.969	0.006	0.014	0.003	0.015	0.033	0.002	0.000	0.010	96.652
DE (500, 1,000, 0.8, 0.6)	926,096.500	0.209	0.169	0.895	0.001	0.001	1.144	0.706	0.035	0.042	0.009	0.037	0.022	0.024	0.006	0.021	96.678
DE (50, 1,500, 0.5, 0.5)	153.164	0.040	0.083	0.972	0.001	0.001	1.271	0.941	0.000	0.000	0.000	0.001	0.010	0.001	0.000	0.000	96.680
DE (50, 1,500, 0.5, 0.6)	153.021	0.040	0.085	0.972	0.001	0.001	1.269	0.941	0.000	0.002	0.000	0.001	0.009	0.000	0.000	0.000	96.679
DE (50, 1,500, 0.5, 0.7)	152.896	0.040	0.085	0.970	0.001	0.001	1.270	0.940	0.000	0.000	0.000	0.001	0.010	0.000	0.000	0.000	96.681
DE (50, 2,000, 0.5, 0.5)	152.956	0.040	0.085	0.971	0.001	0.001	1.268	0.942	0.000	0.001	0.000	0.001	0.009	0.001	0.000	0.000	96.681
DE (50, 2,000, 0.5, 0.6)	152.858	0.040	0.085	0.971	0.001	0.001	1.270	0.943	0.000	0.000	0.000	0.000	0.009	0.000	0.000	0.000	96.681
DE (50, 2,000, 0.5, 0.7)	152.666	0.040	0.085	0.971	0.001	0.001	1.269	0.940	0.000	0.001	0.000	0.000	0.011	0.000	0.000	0.000	96.681
SOMA (1,000, 200, 0.1, 0.1)	152.582	0.040	0.085	0.972	0.001	0.001	1.268	0.940	0.000	0.000	0.000	0.000	0.012	0.000	0.000	0.000	96.681
SOMA (1,000, 200, 0.1, 0.11)	152.717	0.040	0.085	0.972	0.001	0.001	1.268	0.942	0.000	0.000	0.000	0.000	0.011	0.000	0.000	0.000	96.680
SOMA (1,000, 200, 0.1, 0.3)	152.935	0.040	0.085	0.971	0.001	0.001	1.268	0.940	0.000	0.004	0.000	0.000	0.008	0.000	0.000	0.000	96.681
		Desired composition vectors															
Minimal composition vector		0.030	0.065	0.952	0	0	1.268	0.940	0	0	0	0	0	0	0	0	96.671
Ideal composition vector		0.035	0.075	0.962	0.001	0.001	1.288	0.960	0	0	0	0	0.003	0	0	0	96.676
Maximal composition vector		0.040	0.085	0.972	0.006	0.002	1.308	0.980	0.010	0.005	0.002	0.002	0.013	0.005	0.005	0.005	96.681

4 Conclusion

We have proposed alternative way to compute steel alloying recipe. Method using evolution algorithms has several advantages when compared to most commonly used Simplex method. It allows to incorporate final cost of used alloying materials into equation. Result of Simplex method is always cheapest solution in given range. Using our fitness function described in Section 2 we can control the importance of precise solution of final steel over it's price.

All runs presented in our experiments that were able to achieve less then approximately 155 fitness value provide satisfactory results and are fully prepared to be used in real life environment on real life cases.

We would like to extend our work and try out more evolution algorithms to compare their results. Mainly we would like to design larger experiment to compare performance and precision of larger set of evolution algorithms on this problem.

Acknowledgement. This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and by the Development of human resources in research and development of latest soft computing methods and their application in practice project, reg. no. CZ.1.07/2.3.00/20.0072 funded by Operational Programme Education for Competitiveness, co-financed by ESF and state budget of the Czech Republic.

References

1. dos Santos Coelho, L.: Self-organizing migration algorithm applied to machining allocation of clutch assembly. Mathematics and Computers in Simulation 80(2), 427–435 (2009)
2. Senkerik, R., Zelinka, I., Oplatkova, Z.: Evolutionary Techniques for Deterministic Chaos Control. Technological Developments in Education and Atomation, 391–396 (2010), 10.1007/978-90-481-3656-8_71
3. De Falco, I.: Differential Evolution for automatic rule extraction from medical databases. Applied Soft. Computing, 1265–1283 (2013), 10.1016/j.asoc.2012.10.022
4. Koloseni, D., Lampinen, J., Luukka, P.: Differential Evolution Classifier with Optimized Distance Measures for the Features in the Data Sets. In: Snasel, V., Abraham, A., Corchado, E.S. (eds.) SOCO Models in Industrial & Environmental Appl. AISC, vol. 188, pp. 103–111. Springer, Heidelberg (2013)
5. Zelinka, I., Chadli, et al.: An investigation on evolutionary reconstruction of continuous chaotic systems. Mathematical and Computer Modelling 57, 2–15 (2013)
6. Senkerik, R., et al.: Performance comparison of differential evolution and SOMA on chaos control optimization problems. International Journal of Bifurcation and Chaos 22 (2012), doi:10.1142/S021812741230025X
7. Sun, Y., et al.: A hybrid co-evolutionary cultural algorithm based on particle swarm optimization for solving global optimization problems. Neurocomputing 98, 76–89 (2012)

8. Zhou, Y., Li, X., Gao, L.: A differential evolution algorithm with intersect mutation operator. Applied Soft Computing 13, 390–401 (2013), doi:10.1016/j.asoc.2012.08.014
9. Alguliev, R., Aligulizev, R., Isazade, N.: DESAMC+DocSum: Differential evolution with self-adaptive mutation and crossover parameters for multi-document summarization. Knowledge - Based Systems, 21–38 (2012), doi:10.1016/j.knosys.2012.05.017
10. Onwubolu, G.C., Babu, B.V.: New Optimization Techniques in Engineering. STUDFUZZ, vol. 141. Springer, Heidelberg (2004)
11. Chakraborty, U.K.: Advances in Differential Evolution. Springer, Heidelberg (2008) ISBN 978 - 3 - 540 - 68827 - 3
12. Ghosh, A., Chatterjee, A.: Ironmaking and steelmaking: theory and practice. Prentice-Hall of India, New Delhi (2008) ISBN 978-812-0332-898
13. Dantzig, G.B., Thapa, M.N.: Linear programming, vol. 2. Springer, New York (c1997-2003) ISBN 03879861382-
14. Zhang, Y., Gong, D.W., Zhang, J.H.: Robot path planning in uncertain environment using multi-objective particle swarm optimization. Neurocomputing, 172–185 (2013), doi:10.1016/j.neucom.2012.09.019
15. Chakaravarthy, G.V., Marimuthu, S., Sait, A.: Performance evaluation of proposed Differential Evolution and Particle Swarm Optimization algorithms for scheduling m-machine flow shops with lot streaming. Journal of Intelligent Manufacturing 24, 175–191 (2013), 10.1007/s10845-011-0552-2
16. Subramanian, P., et al.: PRISM: PRIority based SiMulated annealing for a closed loop supply chain network design problem. Applied Soft Computing 13, 1121–1135 (2013), doi:10.1016/j.asoc.2012.10.004
17. Chengfo, S.: Improved differential evolution algorithms, Computer Science and Automation Engineering (CSAE). In: 2012 IEEE International Conference, May 25-27, pp. 142–145 (2012)
18. Price, K.V., Storn, R.M., Lampinen, J.A.: Differential Evolution A Practical Approach to Global Optimization. Springer (1997)
19. Zelinka, I., Skanderova, L.: Investigation on Evolutionary Control and Optimization of Chemical Reactor. In: Snasel, V., Abraham, A., Corchado, E.S. (eds.) SOCO Models in Industrial & Environmental Appl. AISC, vol. 188, pp. 469–474. Springer, Heidelberg (2013)
20. Pavlech, M.: Self-organizing Migration Algorithm on GPU with CUDA. In: Snasel, V., Abraham, A., Corchado, E.S. (eds.) SOCO Models in Industrial & Environmental Appl. AISC, vol. 188, pp. 173–182. Springer, Heidelberg (2013)
21. Kadlec, P., Raida, Z.: A Novel Multi-Objective Self-Organizing Migrating Algorith. Radioengineering 20, 804–816 (2011)

The Cost Estimation of Production Orders

Tomasz Chlebus

Wroclaw University of Technology, The Faculty of Computer Science and Management,
Institute of Organization and Management
tomasz.chlebus@pwr.wroc.pl

Abstract. Time and cost oriented management of production orders requires methods and models of business processes in the enterprise as well as a structural analysis and a reference model of a production process based on a model of a production order. Estimating costs of the execution of a production order may contribute to initial and quite accurate analysis of the costs of the order. The developed method allows for estimation of the interdependence of costs in comparison with different clients. An example shows an analysis of costs of the manufacture of a frame for usual clients, who have no specific requirements, and for clients with additional requests regarding the products. The analysis of the costs of execution of such orders may help estimate the costs of diverse products in the future as well as predict cost variations for different orders. Additional operations may be taken into account and added to the final assessment of costs, which will allow for more accurate cost calculating and indicating orders with worse profitability or orders which cannot be executed due to excessive production costs. The method allows for assessment of production costs both in a sequential model and in distributed manufacturing.

Keywords: Cost analyze simulation production system order estimation.

1 Introduction

Requirements of market economy and increasing competition is forcing companies to change the management style in the direction of rationality and economic efficiency, which is based on the correct use of all outstanding offer companies the means of production. This is possible only if the decision-making processes are based on full information about the state of the company. Adequate information is required for economic decision making at all levels of management. Special role in the information system of economic information companies represent a high cost because the costs are the economic parameter, which shows the level of effort and resources involved in the production of economic activity. The focus is on cost management processes. Processes are identified, which raises the value of the so-called. value chain to be added. The management costs are related to the rationalization of their new methods of managing such a radical reconstruction processes. By contrast, directly aimed at cost management are the new cost accounting methods as developed in the U.S. cost accounting activities (Activity Based Costing - ABC)[1] and a native of Japan, target costing (Target Costing)[2], and the concept of product life management

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 465–475, 2013.

costs. Thus improving the balance sheet of a company is possible, inter alia, by carefully analyzing the cost of production and final demand. The best way to reduce costs is their foresight that can be made from the very beginning of its life cycle, so in the course of its conceptual design. Early diagnosis and determine the cost of the design, manufacture and distribution of finished products will allow the company to better manage their own capital and relationships with network providers. Basic data processing phase of construction-technological and organizational device manufacturing process is also called the phases of product development - design (Fig.1). What is essential in costs of manufacturing process, especially during the development phase, is the possibility of influence of the designer and planner for the development costs implementation phases of the manufacturing process. As can be seen from the diagram as much as 70% of the cost is generated in the development phase of the product and its manufacturing processes.[3]

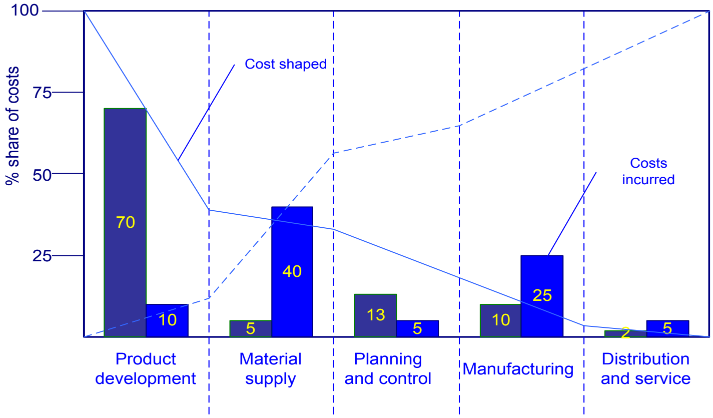


Fig. 1. Phase of product development and cost [3]

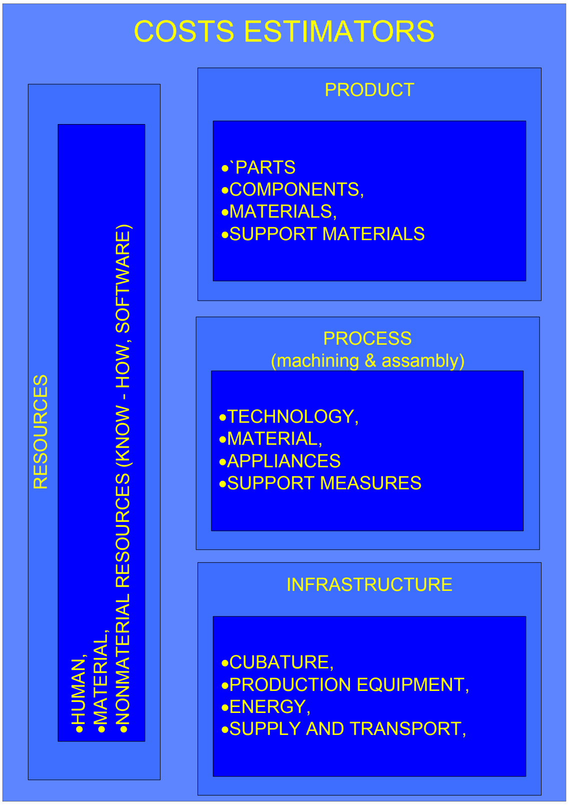
Items to be taken into account in this analysis are as follows:

- the purchase of materials,
- costs of materials,
- means of internal transport
- costs associated with machines, tools and resources needed to manufacture,
- costs of servicing and ancillary staff (not production),
- storage of materials for the production in progress.

When examining the impact of individual elements on the final cost of the final product can be assumed that the cost of buying the materials will have the greatest impact on production. It is not always true. We are seeking suppliers of materials to choose from a number acceptable, which means that you can negotiate a purchase price of raw materials and components so that the costs incurred, did not influence significantly the cost of producing the product. On the other hand, need to pay special attention to the quality of the raw materials and components, whether they meet the requirements and whether the supply is consistent with the contract and the date.

Costs incurred during the production mainly generated by the many ingredients in the workplace and have a significant impact on the general costs of production, Fig.2.

Places cost order at this stage of creating the finished product can be found primarily on the machinery, equipment and tools intended for machining and assembly components (in addition to the costs of course material, which is usually the price does not change dynamically). In order to solve problems cost please consider what factors influence the development of costs in the production process.



COSTS ESTIMATORS

RESOURCES

•HUMAN,
•MATERIAL,
•NONMATERIAL RESOURCES (KNOW - HOW, SOFTWARE)

PRODUCT

•`PARTS
•COMPONENTS,
•MATERIALS,
•SUPPORT MATERIALS

PROCESS
(machining & assambly)

•TECHNOLOGY,
•MATERIAL,
•APPLIANCES
•SUPPORT MEASURES

INFRASTRUCTURE

•CUBATURE,
•PRODUCTION EQUIPMENT,
•ENERGY,
•SUPPLY AND TRANSPORT,

Fig. 2. Estimators costs in the production process [prep.]

Expenditure incurred in the almost three pillars (Fig.3):

1. Product development,
2. Process development and production planning,
3. Machining, production control and logistic and assembly process management.

The most important step and also the point where the costs are planned and decisions are taken, often unknowingly, to the cost of production is where you prepare the documentation and product design, develop manufacturing technologies, identify the necessary resources and organization recalculate pre-project costs.

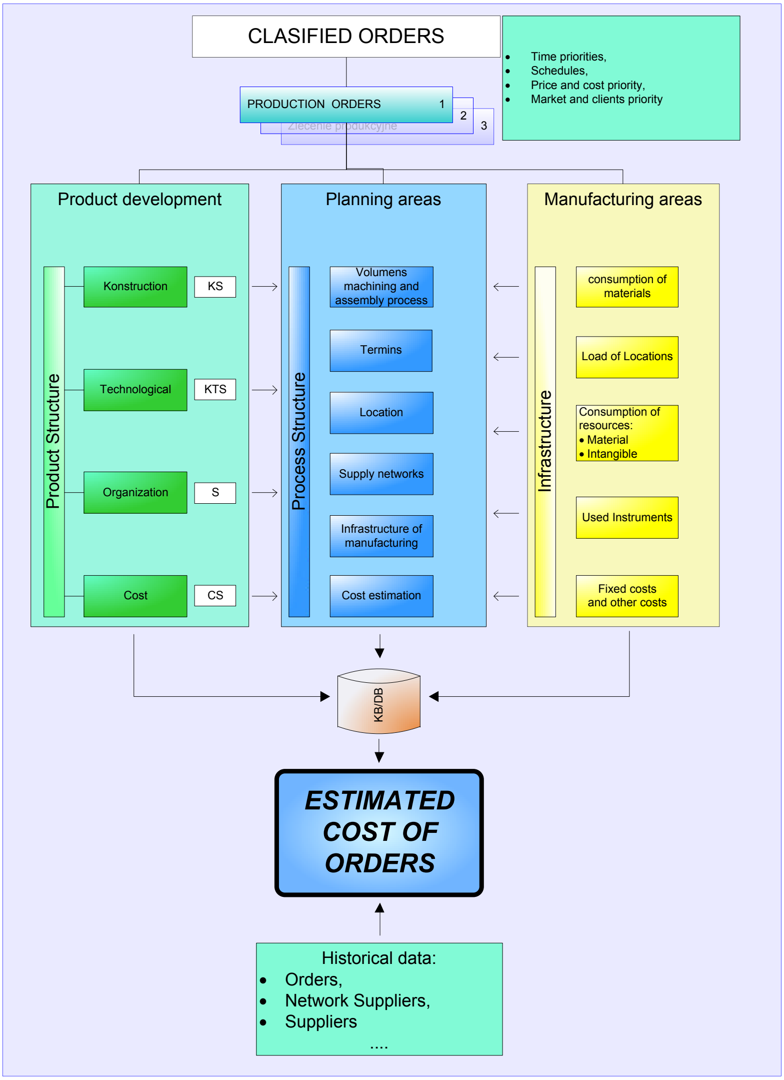


Fig. 3. Factors affecting the cost of production orders (prep.)

2 Cost Data Model for Forecast

For the preparation of product documentation will be needed design, construction, technology, information on customer requirements. The costs of construction are not one of the most spending in the first phase of the product life, and therefore in the design. This happens because when the design is extremely difficult to assess whether the costs identified, incurred during this first phase will be in effect as they are scheduled.

In the preparation phase of production should also take into account the cost of implementing the technology and technological burden of production. The structure should be taken into account the technological process of manufacturing the elements taken from the structure of construction and the identity of all aspects of production on some activities such as:

- The order of delivery of materials to production lines,
- Transport equipment responsible for the internal transportation of raw materials,

Cost based on the structure of the product it is necessary to determine when and what the costs and time are applied to the product. Analysis of the structure of the product, the tree cost and the time of manufacture is shown on Fig.4 (Ishikawa diagram [4]).

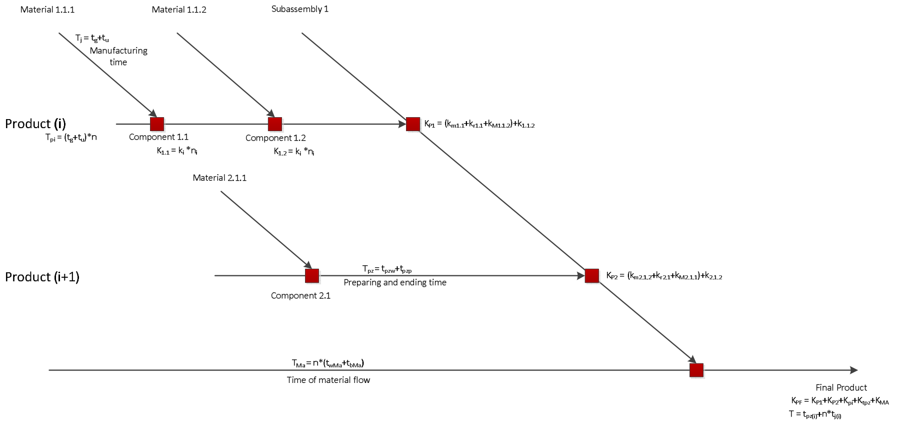

Material 1.1.1
Material 1.1.2
Subassembly 1
$T_j = t_g+t_u$
Manufacturing time
Product (i)
$T_{pi} = (t_g+t_u)*n$
Component 1.1
$K_{1.1} = k_i *n_i$
Component 1.2
$K_{1.2} = k_i *n_i$
$K_{P1} = (k_{m1.1}+k_{r1.1}+k_{M1.1.2})+k_{1.1.2}$
Material 2.1.1
Product (i+1)
Component 2.1
$T_{pz} = t_{pzw}+t_{pzp}$
Preparing and ending time
$K_{P2} = (k_{m2.1.2}+k_{r2.1}+k_{M2.1.1})+k_{2.1.2}$
$T_{Ma} = n*(t_{wMa}+t_{bMa})$
Time of material flow
Final Product
$K_{PF} = K_{P1}+K_{P2}+K_{pi}+K_{tpz}+K_{MA}$
$T = t_{pz(i)}+n*t_{j(i)}$

Fig. 4. Model of balancing time and cost components in the node of the article (prep.)

T –	Production time
T_{Ma} –	Time of material flow
T_{wMa} –	Main time of material flow
T_{bMa} –	Brake time of material flow
T_{pz} –	Preparing and ending time
T_{pzw} –	Main preparing and ending time
T_{pzp} –	Subsidiary preparing and ending time
K_{PF} –	Costs of final product
K_{P1} – Costs of sub product	
K_{m1} –	Cost of material flow
K_{r1} –	Cost of produce of subassembly
K_{M1} –	Cost of material

3 The Algorithm of Costs Estimating Process

Taking into account these elements can be to build an algorithm of determining the cost of labour and production orders, which was presented at Fig.5 and Fig.6. By proceeding according to the individual steps away from the top of a full scenario in which there is to generate models: product, process and manufacturing system and manufacturing industry. These models are intended to identify the individual modes of production and providing the necessary data to apply methods, models and simulation tools.

The all-important issue in planning a new production facility is to determine the output level and pair it with an adequate production process. The output level is determined based on available expert and theoretical knowledge [9].Very important information needed in this algorithm should be stored and accessed in knowledge bases and data bases. Data that are needed in order planning should include: specifications, procedures for KM and IT, and the other data that may be used in future. A significant role in this algorithm is modelling and simulation. Results of those to steps are easy to receive not incurring additional costs and charge on the production line. Of course there are needs that the enterprise that take or calculate order has to prepare a few variants of the order realization, but with sure they are less cost-intensive. There might to be more than one model in result, but all models should be stored in DB/KB, and be stored and analyzed in different productions orders. Continuous improvement of manufacturing system is important point of this method because the environment of production is changing dynamic.

Another important step is to generate simulation models, which will estimate the value sought. On this basis, a specific model of the aggregate costs and labour. The next step is to introduce data to the simulation programs such as Arena, ProModel and Igrafx and track the accuracy of the model built. With formula, you cannot tell right away whether it is best and meets the expectations. Therefore, on the basis of the model is base generates other variants that may arise during the actual production. The next step should be to compare the results with data from databases to determine the validity of the analysis and to be able to verify the developed models. Of course the model must be stable. Both supply (shipment) and production system should have to stable input. Ship stability system is rather complicated [5], but in this model material delivery is realized by cells called kits. Following this analysis and find the best variant of the model, change the base model and deploy the generated model to an existing company. The company is a living, functioning organism, therefore, be still and seek to improve the system both in the form of the model and the actual conditions of production.

Described company manufactures freight and passenger trains as well as complete transportation systems. Wroclaw plant of the company consists of three divisions: Locomotives, Bogies, Service. The project presented in this paper was conducted in Bogies Division, where bogies frames for locomotives and for passenger trains are produced [6]. The following Fig.7 presents an example of the deployment of machines from the ProModel simulation.

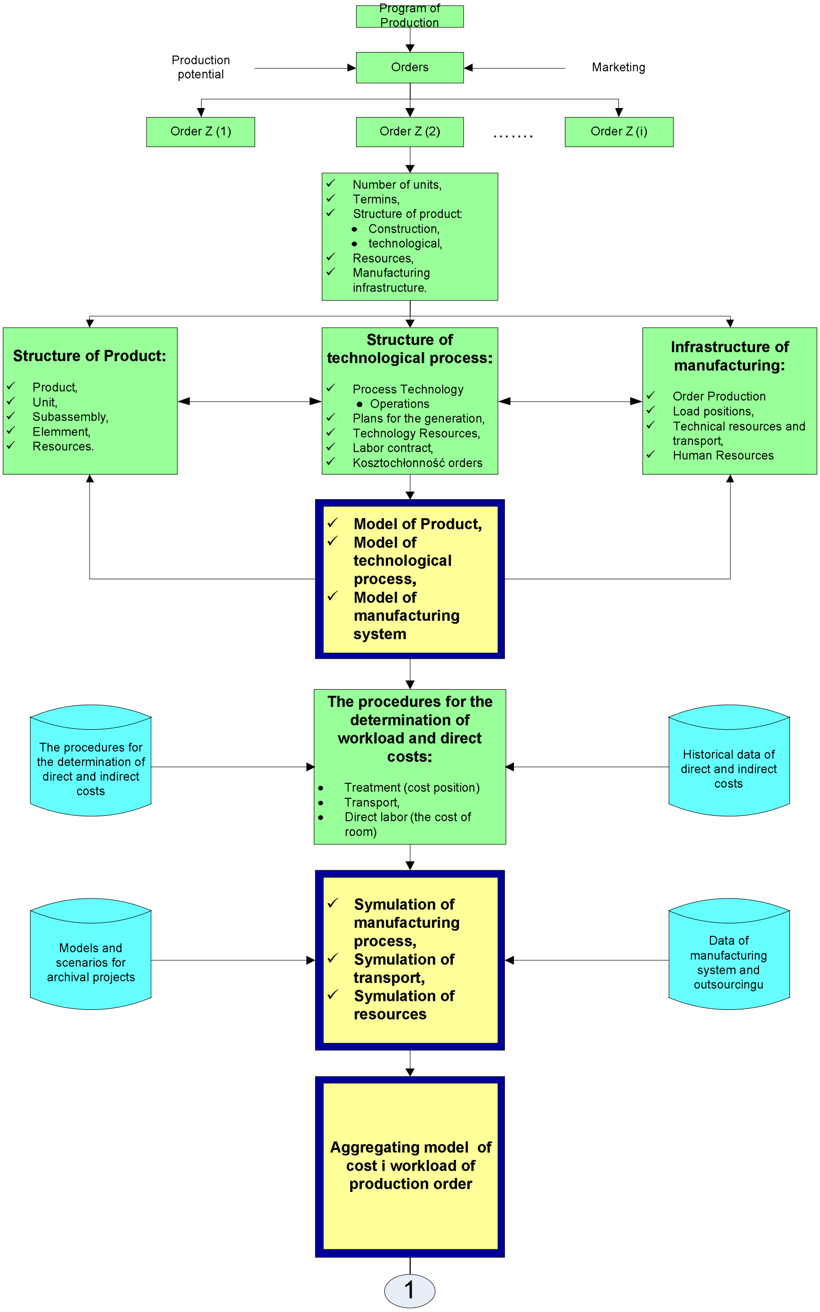

Program of Production
Production potential
Orders
Marketing
Order Z (1)
Order Z (2)
…….
Order Z (i)
✓ Number of units,
✓ Termins,
✓ Structure of product:
• Construction,
• technological,
✓ Resources,
✓ Manufacturing infrastructure.
Structure of Product:
✓ Product,
✓ Unit,
✓ Subassembly,
✓ Elemment,
✓ Resources.
Structure of technological process:
✓ Process Technology
• Operations
✓ Plans for the generation,
✓ Technology Resources,
✓ Labor contract,
✓ Kosztochłonność orders
Infrastructure of manufacturing:
✓ Order Production
✓ Load positions,
✓ Technical resources and transport,
✓ Human Resources
✓ Model of Product,
✓ Model of technological process,
✓ Model of manufacturing system
The procedures for the determination of direct and indirect costs
The procedures for the determination of workload and direct costs:
• Treatment (cost position)
• Transport,
• Direct labor (the cost of room)
Historical data of direct and indirect costs
Models and scenarios for archival projects
✓ Symulation of manufacturing process,
✓ Symulation of transport,
✓ Symulation of resources
Data of manufacturing system and outsourcingu
Aggregating model of cost i workload of production order
1

Fig. 5. The algorithm of determination of workload and cost of production order (Part 1) (prep.)

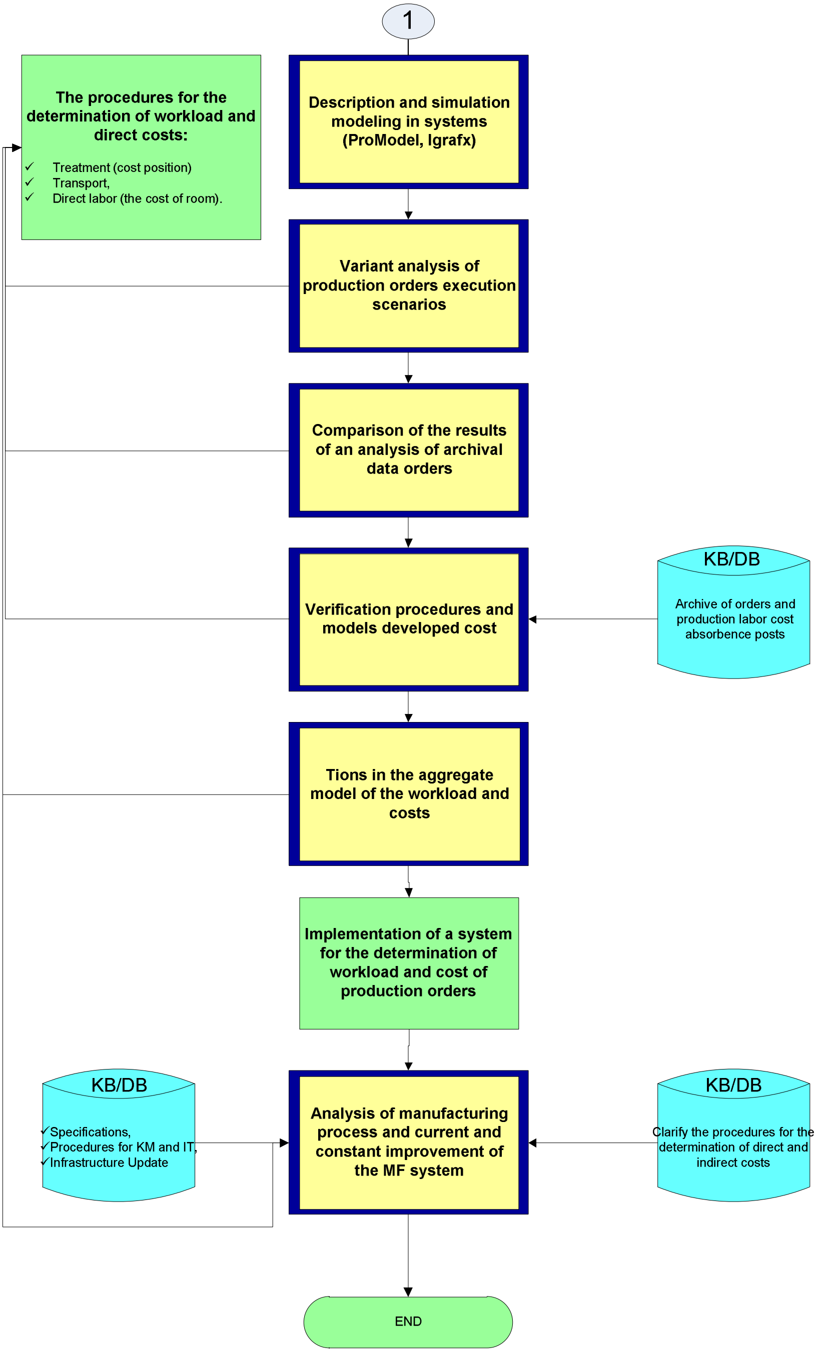

1
The procedures for the determination of workload and direct costs:
✓ Treatment (cost position)
✓ Transport,
✓ Direct labor (the cost of room).
Description and simulation modeling in systems (ProModel, Igrafx)
Variant analysis of production orders execution scenarios
Comparison of the results of an analysis of archival data orders
Verification procedures and models developed cost
KB/DB
Archive of orders and production labor cost absorbence posts
Tions in the aggregate model of the workload and costs
Implementation of a system for the determination of workload and cost of production orders
KB/DB
✓Specifications,
✓Procedures for KM and IT,
✓Infrastructure Update
Analysis of manufacturing process and current and constant improvement of the MF system
KB/DB
Clarify the procedures for the determination of direct and indirect costs
END

Fig. 6. The algorithm of determination of workload and cost of production order (Part 2) (prep.)

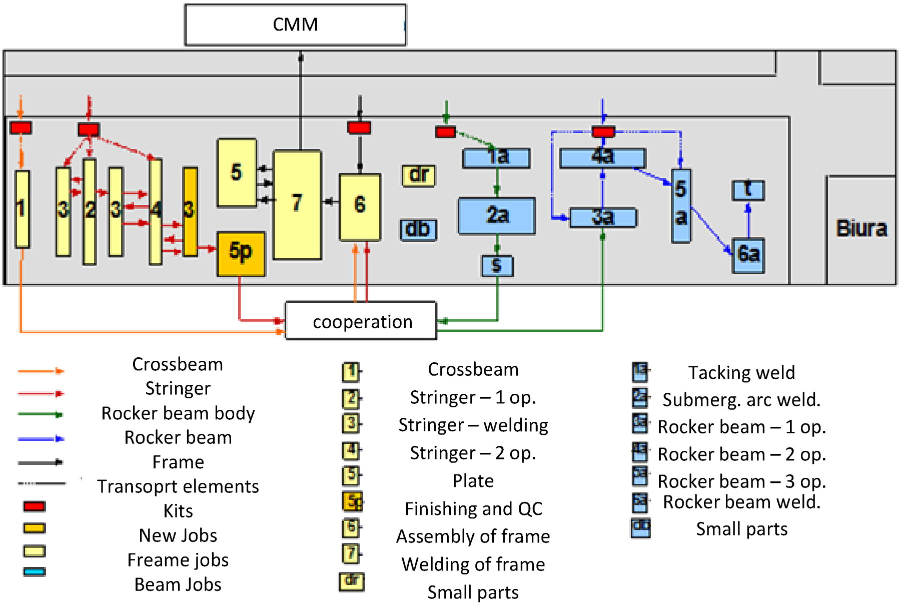


Fig. 7. The plan layout production system simulation framework for ProModel[7]

In addition, the final model was optimized in terms of:

- the optimal number of units of transport,
- shortening the length of road transport
- optimal use of productive resources,
- optimal use of the production line.

After inputting new workstations in layout and applying some productions changes there are some interesting results which can see on (Fig.8.) presented improved the productivity of a plot line for the new version.

From the analyzes and the developed method can draw insights indicating that the full identification of the product model in the form of its hierarchical structure and complete information describing parameterized processes and production together with the available human and technical resources allows for accurate and rapid identification of the effort and cost of production job. It may be carried out in a simplified way, using tools such as MS Excel or in a multiple and multivariate analysis, with the help of simulation systems such as iGrafix or ProModel. Taking these elements set out in previous chapters, you can use them build an algorithm determining workload and cost of production job was, which is shown in Fig.5 and Fig.6. Items that are shown in the boxes, there are new procedures that have been introduced by the author to calculate the effort and cost of production order. Following each step from the top of the full scenario arises as a result of which there is to generate models of the product and the manufacturing process technology and manufacturing system. These models are intended to describe the various phases of production and providing the necessary data for the application of simulation tools.

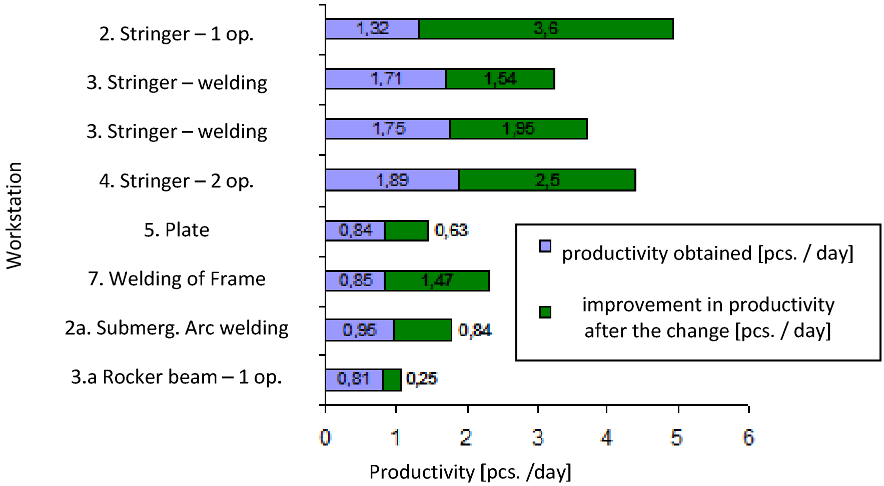


Fig. 8. Summary of productivity, obtained before and after the changes made for selected workstations

4 Summary

The algorithm for estimating the costs can be particularly useful for SMEs (small and medium-sized enterprises)[8], which for reasons of economic systems cannot implement ERP / Bus, and are devoid of advanced calculation tools which significantly reduces their competitiveness. CAD Tools / TPP / CAP / TPM today are already widely used in SMEs and can provide a basis to build custom implementations of storage product knowledge, technology and markets, which can significantly improve the competitiveness of not only business but also can ensure the stability of market and innovation through the use of accumulated knowledge on immunization and personnel turnover.

References

1. Cooper, R., Kaplan, R.S.: Profit Priorities from Activity-Based Costing. Harvard Business Review (May-June 1991)
2. Filomena, T.P., Neto, F.J.K., Duffey, M.R.: Targetcosting operationalization during product development: Model and application. International Journal of Production Economics 118(2)
3. Bauer, C.-O.: Produkthaftung-Ansprüche an die Konstruktion haben einen Anteil von70%. Maschinenmarkt, nr 68 (1984)
4. Konieczka, P., Namieśnik, J.: Chem. Anal. 53, 785 (2008)
5. Kobyliński, L.: System and risk approach to ship safety, with special emphasis of stability. Archives of Civil and Mechanical Engineering 7
6. Burduk, A., Chlebus, E.: Methods of risk evaluation in manufacturing systems. Archives of Civil and Mechanical Engineering 9

7. Chlebus, E., Burduk, A., Chrobot, J., Kowalski, A., Wierzchowski, L.: Variant simulation and optimisation of production system in Bombardier Transportation Polska Company, Zarządzanie Przedsiębiorstwem, vol. 7(2), pp. s.15–s.20 (2004)
8. Wilde, S.: Small and Medium-Sized Enterprises. Customer Knowledge Management. Springer (2011)
9. Kowalski, A., Marut, T.: Hybrid Methods Aiding Organisational and Technological Production Preparation Using Simulation Models of Nonlinear Production Systems. In: Corchado, E., Snášel, V., Abraham, A., Woźniak, M., Graña, M., Cho, S.-B. (eds.) HAIS 2012, Part II. LNCS, vol. 7209, pp. 259–266. Springer, Heidelberg (2012)

Achieving Desired Cycle Times by Modelling Production Systems

Marcin Juszczyński and Arkadiusz Kowalski

Institute of Production Engineering and Automation, Wroclaw University of Technology,
5/7 Lukasiewicza St., 50-370 Wroclaw
130989@student.pwr.wroc.pl, arkadiusz.kowalski@pwr.wroc.pl

Abstract. Modelling and simulating discrete production processes issues were discussed in this paper. Electric scooter production line at the start-up stage was analysed. Requirements towards its efficiency were change before the line going on-line. Simulation model used production plans, drafted prior to starting production. It considered among other adding machinery, increase production capacity, process of production and transport, transport routes, layout, material flow, human resources and means of transport.

The manufacturing process was designed to use KANBAN. Hence simulation models including various alternative solutions of Technical and Organisational Production Set-up were used to verify whether 12 minute cycle time was achieved. Modification to the production system included among other interchangeability of workstations and operators. Consequently bottle necks in the production process were eliminated.

The project discussed in this paper proves feasibility and validity of combining simulation of production systems with Lean Manufacturing tools.

Keywords: simulation of discrete processes, production system, cycle time.

1 Introduction

One of many definitions of simulation is: "Simulation is the art and science of creating representations of a process or system in order to conduct experiments and assessments". [6], [11] Simulation methodologies are most often classified by time. [5], [13] In DESS – Differential Equation System Specification both the states and attributes of the model are being update continuously. In DEVS – Discrete Event System Specification on the other hand, system status is approximated for fixed time periods. Those models may be referred to as time-dependent. System status is recognised using the predetermined time periods, hence attributes and system status may be registered. In case of discrete events, time is registered once system status is changed i.e. at least one attribute of an object changes in the modelled system. Moreover, the DEVS&DESS method combines the two above mentioned methods.

Simulation has become a commonly used tool to engineer production systems. [3], [9] Modelling by simulation is one of the most important supporting techniques allowing to cut production time of a new product and its market roll-out. Naturally, it is

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 476–486, 2013.

possible to investigate real production systems, however, it is not economically viable. Hence additional steps currently used in production system engineering are building best possible simulation model, its validation and computer simulation, review of obtained results testing whether or not theoretical solutions are in fact effective and if necessary making some adjustments.

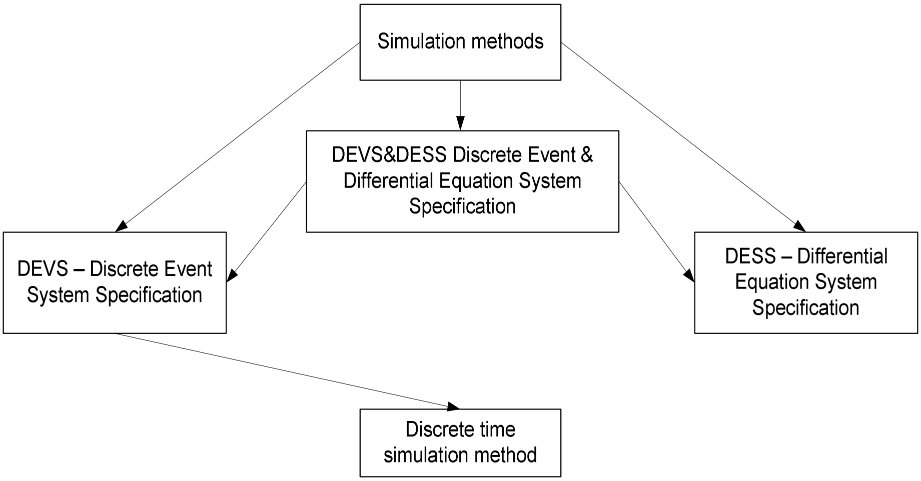

Simulation methods
DEVS&DESS Discrete Event & Differential Equation System Specification
DEVS – Discrete Event System Specification
DESS – Differential Equation System Specification
Discrete time simulation method

Fig. 1. Classification of simulation methods [13]

Combining modelling with simulation in investigating production processes allows among other to:

- streamline processes and eliminate waste of time and resources,
- improve efficiency and cut production costs,
- analyse and simulate different but critical cases and develop adequate procedures to action,
- create excellent environment for training and understanding processes taking place in an organisation. [1], [2], [7]

Computer simulation is also one of the most important tools for planning production. It allows for addressing market demand in a flexible manner and cut costs of production. [10], [8], [12] Currently available simulation suites are versatile and user-friendly. The most popular systems are FACTOR/AIM, ProModel, Taylor II, Arena and WITNESS. The most useful features include model building environment using predefined graphical elements, considerably improving the process of building a simulation model.

Presented below example projects regarding Technical and Organisational Production Set-up prove it is possible to combine production system simulation techniques with Lean Management tools, here with the Kanban scheduling system.

2 Objective of the Project

The primary objective of the project was to analyse and streamline material flow for purposes of starting-up electric scooter production line.

Simulation models of investigated production system were based on production plans, drafted prior to starting production. It considered among other adding machinery, increase production capacity, process of production and transport, transport routes, layout, material flow, human resources and means of transport. The main criterion for designing the layout was minimum transporting distance for products and materials.

The project took two stages to complete, which focused on expanding production area, adding machinery and increasing manufacturing capacity. The first stage included modelling current layout of the facilities in order to determine current manufacturing capacity and pin-point bottle necks in production lines. The second stage comprised consecutive model modifications until objective defined for particular production cycle was achieved.

3 Description of Technological Process

The main production process if the process of assembly. The scooter, assembled on the main assembly line, is fixed to a carrier platform (Figure 2) which is moved by operators between consecutive workstations.

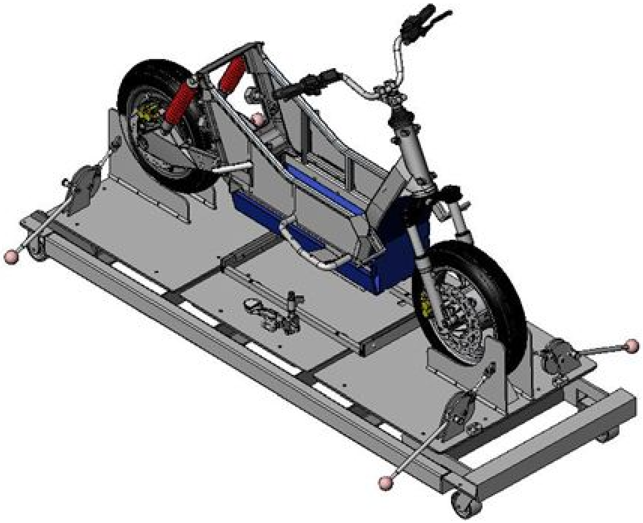


Fig. 2. Platform for assembling the scooter

The platform for assembling the scooter consists of three independently operating surfaces which are lifted depending on currently assembled element. Material flow along the assembly line was designed to allow preliminary component assembly in side assembly cells (both left and right to the main assembly line) for them later to be directly moved to the point where they would be assembled on the main assembly line. Below schematically represented in the scooter production line (Figure 3).

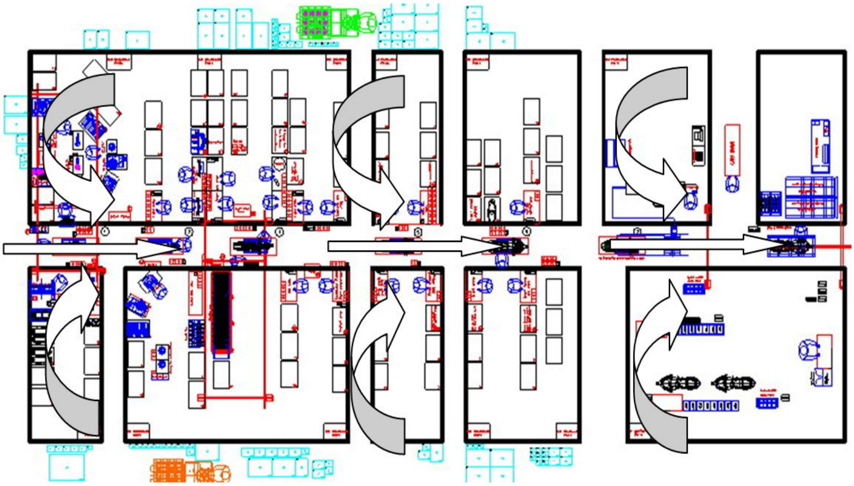


Fig. 3. Assembly line diagram

Each assembly cell uses small pneumatic presses, fixings, assembly tables, power drill-drivers and ancillary equipment. The production line has twenty nine production cells. Seven to the left of the line, eight to the right and fourteen on the main line. Those twenty nine cells consist of sixty six workstations.

Traffic management workstation is also on the shop floor, which stores all tools necessary for the event of production line failure.

4 Development and Implementation of Production System Model

ProModel software was used to build the simulation model. Its functionalities enable to effectively simulate production processes by using predefined objects representing workstations, processed goods and production resources [4]. For purposes of discussed project, the graphics library was built with particular attention to details to assure good quality visualisations, whereas the shop floor layout was imported from AutoCad (Figure 4).

Advantages of that solution are among other that the scale of investigated workstation layout is retained, thus allowing later, in simulation software, to factor in real distance travelled by production resources during transport along transport routes.

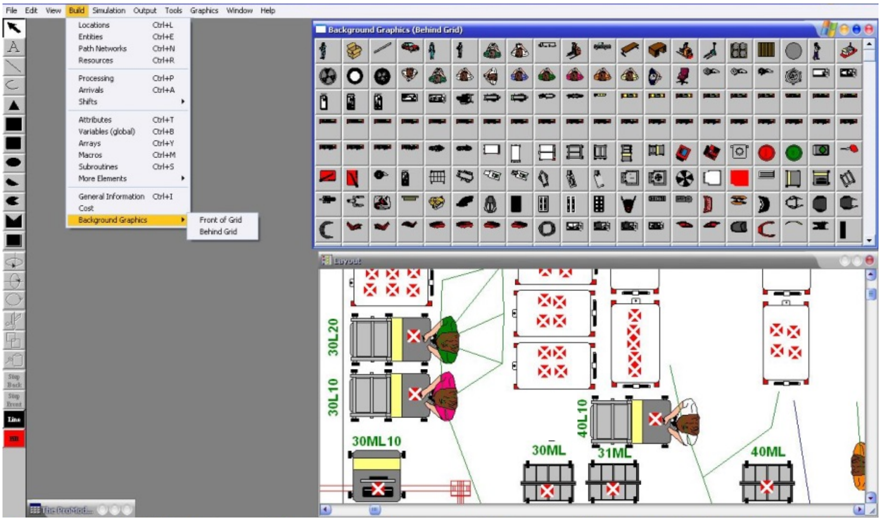


Fig. 4. Simulation model of investigated manufacturing system - workstations

Production processes carried out by individual production cells received the same amount of attention as well as activities completed by individual employees did – Figure 5.

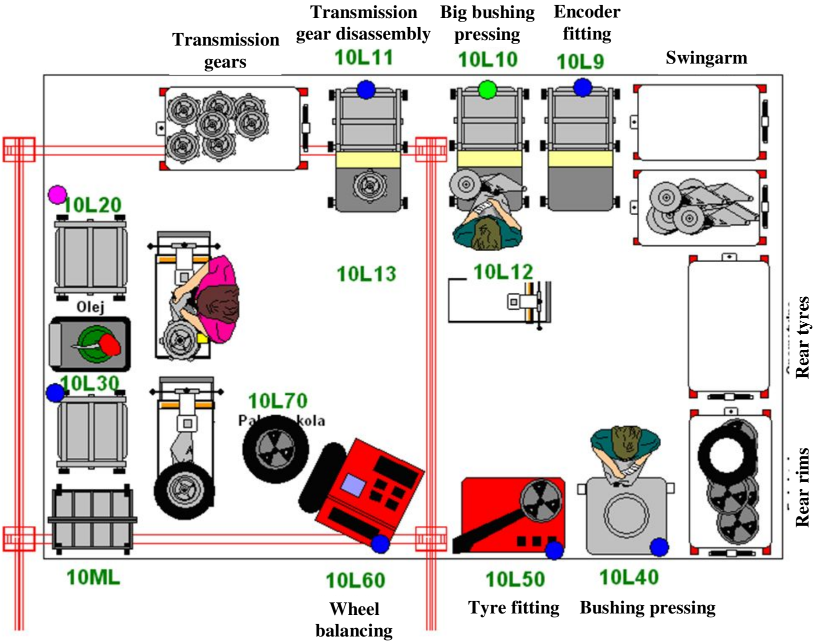


Fig. 5. Details factored into modelling production cells

For instance, modelling cell 10L included the following activities:

- Worker no 1 – preliminary assembly and encoder fitting, pressing bushing onto swingarm and transmission gear disassembly,
- Worker no 2 – fitting transmission gear onto the swingarm, screwing the breather on, pouring oil and fixing wheel to the swingarm. That worker also transports the swingarm to the main line and fits it to the frame on the platform,
- Worker no 3 – pressing small bushings into rear rim, fitting tyres and wheel balancing on the balancing machine.

4.1 Stage I – as-Is State

When building simulation models, it is critical to provide correct data about the modelled production process, e.g. about times of technological operations. Here, time taken by each worker to complete operations at individual stations was measured. Time per each operation was measured three times to calculate an average. Total time of all activities was 354.1 min. Below chart (Figure 6) presents time to completion for operations at individual production cells which were then factored into simulation model.

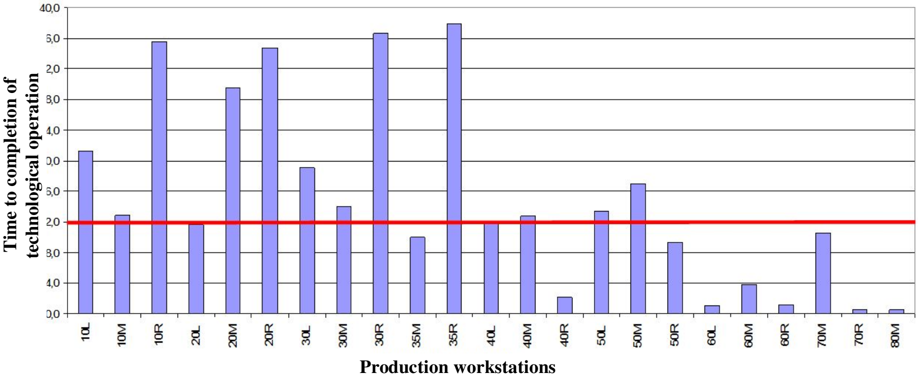


Fig. 6. Time to completion for operations at individual cells

Red line marks the 12 minute target time, which ought to be maintained to assure required cycle of production line. Twelve cells took more time and they were first reorganised. Based on ex-post review of statistics simulation, it was found that workstation utilisation was very diverse. Some workstations are utilised in almost 75%, whereas other in 5%. Worker utilisation, similarly to workstation utilisation, ranges greatly. Obtained data was validated by running it against results achieved by the real production line. Times to completion of e.g. batches of scooters proved negligibly small.

4.2 Stage II – Verification of Proposed Solutions

Impact of modifications and improvements was tested during consecutive simulations of the production system model in order to determine their effect and achieve the objective of 12 minute cycle time. Types of proposed improvements and their performance is presented in Table 1.

Table 1. Types of proposed improvements and their effect

No.	Improvement	Outcome		
		Cycle Time [min]	No of scooters [units/week]	No of workers
1	Two workers were added to the 10R cell, where the frame is assembled	44	48	31
2	Two workers were added for assembly of ICM (Input Control Module) and wheel fitting at the main line	43	48	33
3	Bottlenecks at brake fitting station were eliminated by adding two more workers	27	71	35
4	One worker was added to the 30L station	27	71	36
5	Two workers were added to the 30R stations and one to the 30M station	26	71	38
6	Workers were added to the 30R30 station, where the battery is fitted	25	71	41
7	15M station was added to the main line. Activities at the 20M station were partially relocated to the 15M station	17	106	41
8	Technological operations completed at 50M station were redistributed	15	114	41
9	One worker was added to the 10R station where Motor Controller is fitted.	15	114	42
10	5M stations were added to decrease utilisation of the 10M station.	15	114	42
11	25M station was added to decrease utilisation of the 20M and 30M workstation	15	114	42
12	45M station was added to the main line to decrease utilisation of the 40M station.	11.4	140	42

5 To-Be State Final Model

The final model factors in all 12 proposed organisational improvements. It was then subject to testing in order to verify viability of proposed changes. The final outcome was that time to completion of technological operations at each individual workstation was lower than assumed cycle time.

Chart in figure 7 presents time to completion per operation at individual workstations. Notice, times of consecutive operations at each workstation were lower than the initial target time of 12 minutes – the cross-over point is marked by red line. Some workstations take several minutes, hence one worker can attend to several workstations.

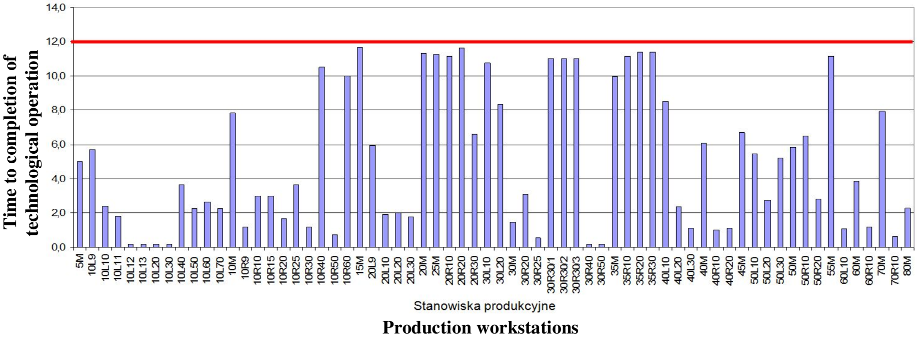


Fig. 7. Time to completion for operations at individual workstations

Figure 8 illustrates, similarly to figure 7, times to completion of operations. The difference is that figure 7 presents times of individual operations, but at production cells. Said chart shows cells with highest utilisation in terms of technological operations. It is evident, those cells are 10R, 20R, 30R and 35R.

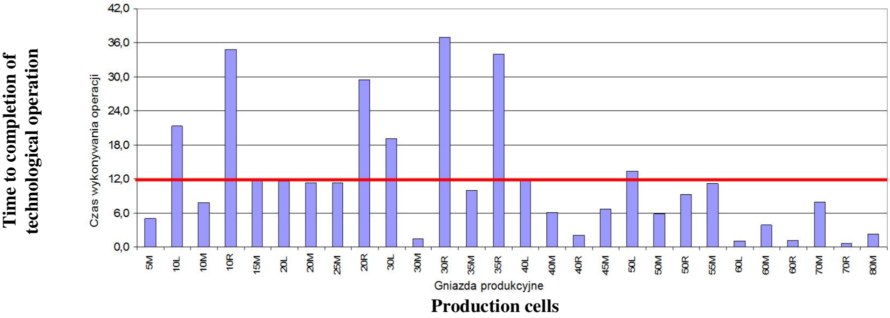


Fig. 8. Time to completion for operations at individual cells

Chart presented in Figure 9 shows the number of scooter and cycle time relative to introduced improvements. Changes made to the production process decreased cycle time, which determines the number of produced scooters.

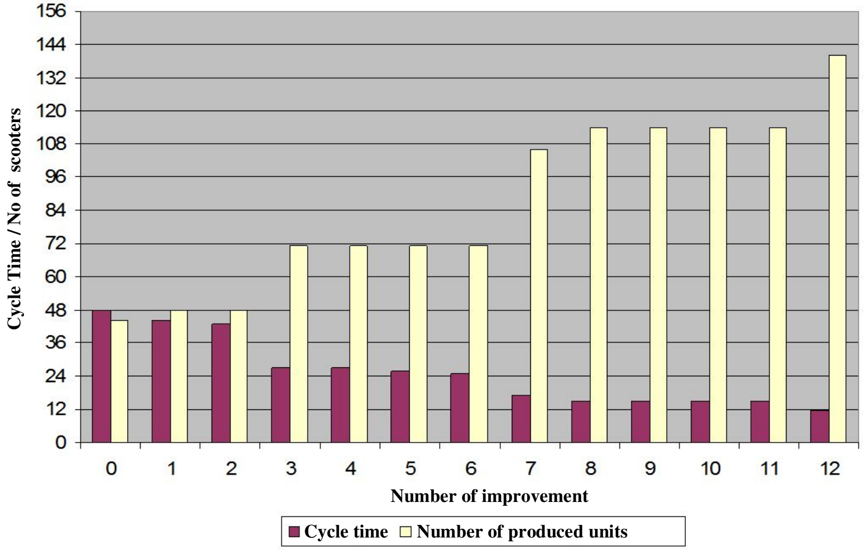


Fig. 9. The number of scooter and cycle time as a function of introduced improvements

The last step of streamlining the production line was optimisation of worker utilisation. Chart showing workstation utilisation before and after changes is presented in Figure 10. The claret colour marks worker utilisation before optimisation, whereas blue colour marks worker utilisation after all the improvements – given 12 minute cycle time.

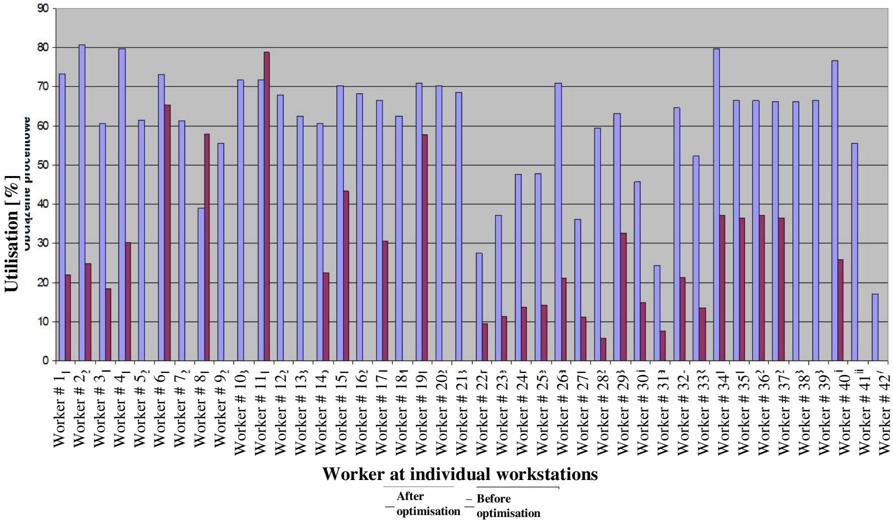


Fig. 10. Worker utilisation before and after optimisation

For comparing the two simulation models, a function calculating median proved useful. Median for worker utilisation before changes was 24%, whereas after optimisation increase up to 66% – approx. threefold. The number of workers in the factory grew from 30 to 42.

6 Summary

The project objective was to analyse and streamline material flow for purposes of scooter production line. The main criterion was to achieve a 12 minute cycle time. Reaching target time was possible by making changes to the machinery, increase manufacturing capacity, transport routes and increasing human resources.

The graphics received particular attention. Image quality and resemblance with real objects were prioritised. All workstations, processed goods and lengths of transport routes were modelled to be identical in terms of dimensions and shapes with real objects. The underlying idea was to represent technological process comparable to the real system, thus enabling to obtain detailed information about material flow.

The project was completed in two stages. The first stage included building the as-is state model in order to pin-point bottlenecks and validate obtained results. At the second stage proposed solutions were verified by assessing their efficiency.

An additional outcome of the project – apart from achieving target cycle time – was achieving more balanced worker utilisation. Figure 11 presents worker utilisation before and after modifying the production line.

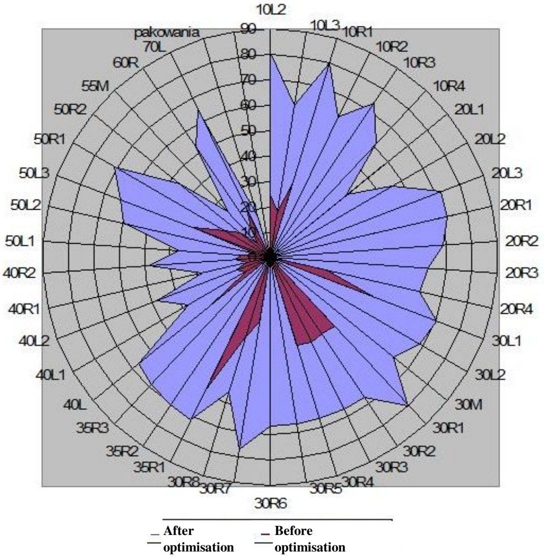


Fig. 11. Worker utilisation before and after streamlining individual production cells

Pie chart was used for graphical representation of the problem. It shows very distinctly utilisation of individual workers. Worker utilisation before optimisation is marked by the claret colour. Blue colour shows worker utilisation after all the changes and improvements. Here, the blue zone is more balanced. The numbers on the chart mean numbers of individual production cells – the first number, and worker number – the second number.

Simulation models and data they generated proved very useful for production system engineering. Initial workstation and resources layout allowed for production of 44 scooters per week, producing cycle time of 48 minutes. After reconfiguration, that time was shortened down to 12 minutes, and the number of scooters rolling off production line jumped to 140 units per week, thus increasing utilisation of resources owned by the company.

References

1. Biniek, Z.: Elements of the systems, modelling and simulation theoretics. INFOPLAN (2002)
2. Chlebus, E., Burduk, A., Kowalski, A.: Concept of a Data Exchange Agent System for Automatic Construction of Simulation Models of Manufacturing Processes. In: Corchado, E., Kurzyński, M., Woźniak, M. (eds.) HAIS 2011, Part II. LNCS, vol. 6679, pp. 381–388. Springer, Heidelberg (2011)
3. Chlebus, T., Stefaniak, P.: The Concept of Intelligent System for Horizontal Transport in a Copper Ore Mine. In: Corchado, E., Snášel, V., Abraham, A., Woźniak, M., Graña, M., Cho, S.-B. (eds.) HAIS 2012, Part II. LNCS, vol. 7209, pp. 267–273. Springer, Heidelberg (2012)
4. Harrell, C., Biman, K., Bowden, R.: Simulation Using ProModel, Utah (2004)
5. Helal, M.: A Hybrid System Dynamics-discrete Event Simulation Approach to Simulating the Manufacturing Enterprise. ProQuest (2008)
6. Hopp, W., Spearman, M.: Factory Physics, Times Mirror Company, Irwin (1996)
7. Ulrich, K., Eppinger, S.: Product Design and Development, 2nd edn. McGraw-Hill, New York (2000)
8. Krenczyk, D., Kalinowski, K., Grabowik, C.: Integration Production Planning and Scheduling Systems for Determination of Transitional Phases in Repetitive Production. In: Corchado, E., Snášel, V., Abraham, A., Woźniak, M., Graña, M., Cho, S.-B. (eds.) HAIS 2012, Part II. LNCS, vol. 7209, pp. 274–283. Springer, Heidelberg (2012)
9. Law, A., Kelton, D.: Simulation Modeling and Analysis, 3rd edn. McGraw-Hill, Singapore (2000)
10. Tisza, M.: Recent achievements in computer aided process planning and numerical modelling of sheet metal forming processes. Journal of Achievements in Materials and Manufacturing Engineering of Achievements in Materials and Manufacturing Engineering 24(1) (2007)
11. Ören, T.: The Many Facets of Simulation through a Collection of about 100 Definitions. SCS M&S Magazine (2011)
12. Westkämper, E., Bischoff, J., Briel, R., Dürr, M.: Fabrikdigitalisierung – Ein angepasster Ansatz für die digitale Fabrikplanungin bestehenden Fabriken und Gebäuden, Werkstatttechnik (2001)
13. Zeigler, B., Praehofer, H., Gon Kim, T.: Theory of Modeling and Simulation: integrating discrete event and continuous complex dynamic systems. Academic Press, San Diego (2000)

Artificial Neural Networks as Tools for Controlling Production Systems and Ensuring Their Stability

Anna Burduk

Wrocław University of Technology, 27 Wybrzeże Wyspiańskiego St.,
50-370 Wrocław, Poland
anna.burduk@pwr.wroc.pl

Abstract. Models of artificial neural networks can be used to control a production system, and thus to ensure its stability. Such models are very useful tools, because they can be built quickly and easily. The only issue is a large amount of data needed in the neural network training process. However, in the era of common availability of IT systems, the parameterization and standardization of production processes is not a problem anymore. Contemporary production systems are mostly automated and metered. This paper presents a method for building a model of an artificial neural network for controlling a wire harness production system and determining its stability.

Keywords: production system, production process stability, neural networks.

1 Introduction

A contemporary customer requires that a product should be not only of good quality and sold at a low price, but also diversified, i.e. available in different versions and variants. In order to meet these requirements, manufacturers are forced to manufacture products in small batches and quickly deliver them to the market. However, a problem with ensuring the stability of production appears here. Production in small batches with highly diversified products and low inventory levels is characterised by a much lower stability as compared with the large-batch manufacturing. Although these disturbances are usually temporary, they may lead to a loss of the functioning or manufacturing stability in a company, which in turn translates into financial losses as well as a loss of customers [5].

In order to ensure smooth functioning of a production system, the stability of its processes must be ensured and, on the other hand, quick decisions, which would be encumbered with the lowest possible risk and uncertainty, should be made. The concept of stability is derived from systems theory and means the ability of a system to return to the steady state after the disturbances have ceased. The assessment of the system stability is a typical task of the qualitative analysis. Stability in a very broad sense means permanency (invariability over time) of a system feature or a certain series of states of the system [4, 6].

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 487–498, 2013.

The steady state is a system state that meets certain conditions. Each system is in a certain state, and because production systems are of dynamic character, a continuous transformation, which causes transition from one state to another, takes place in them. In other words, parameters of a production process may have a different value at any moment [4, 7]. Taking into account any time interval (provisionally adopted unit), it can be said that there is a sequence of output or input states.

When transposing the definition of stability to production systems it can be said that a production system may be considered stable if values of the parameters defining it are within the ranges defined in the planning function and registered in a standard (usually a production plan). Steady state of a production system is presented in Fig. 1.

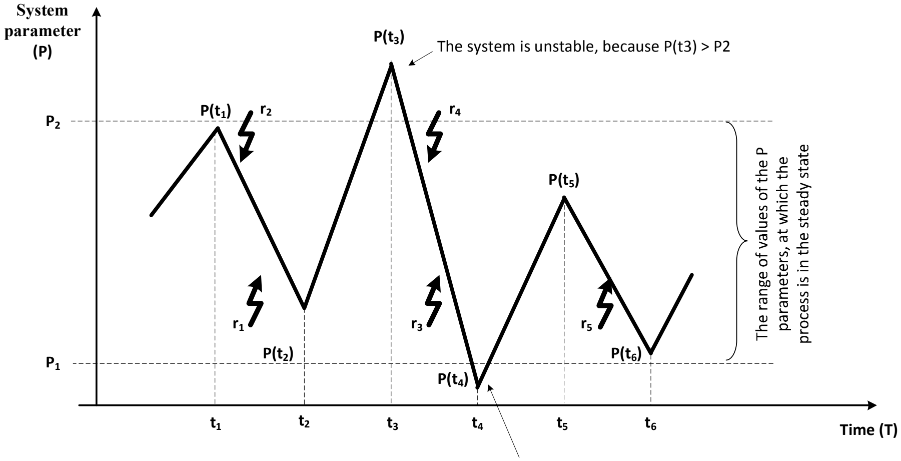


Fig. 1. Steady state of a system

If the value of the parameter P_i in the instant t_i is within a defined range, that is $P_1 \leq P_i \leq P_2$, it proves that the course of the process is correct. Otherwise, corrective measures should be taken. Corrective measures usually consist in changing the values of control variables (inputs to the system X) in such a way, so that the values of the parameters characterizing the controlled variables (outputs from the system Y) return to the process course standards established in the planning function. A correct decision will cause that the system will return to the steady state [3, 8].

In connection with the above, the stability of a production system at the assumed margin of variability will be understood as maintaining the steady state by the system for a certain assumed period. A production system is in the steady state, if values of the parameters characterizing the system are within the ranges defined in the planning function and registered in a standard (usually a production plan).

Production systems are not only of technical nature, but also of economic character, and one of their purposes is to generate profit through a continuous increase of the market share. If the development and an increase in the market share are included in company's plans, the stability of a production system should be then understood as the ability to maintain or increase the values assumed in the parameters plans.

2 Artificial Neural Networks in Production System Control

Decisions are based on objectives with different planning perspectives, while possible variants are verified on various types of models of products, processes and even entire systems. A model is a simplified representation of the reality. It constitutes the main source of knowledge about the object being modelled and the impact of planned solution on the object. A model describes and explains the method of operation and functioning of an object or a system under specific conditions. Studies performed on a model do not cause disturbances in the functioning stability of a real object or system. It offers the possibility to verify the effects of the decisions planned both in relation to the current operations of an enterprise and its future [1, 2, 9].

Usability of models is influenced by their accuracy, possibility of populating them with reliable data, and a short time of building them. It is possible thanks to the progressing unification, standardization and parameterization of production processes and products. Standardization, apart from shortening the time needed to build a model, also lowers the costs of product development or production process improvement.

In the case of management or optimization of a production process, a model is used to verify the decisions planned. Thanks to the use of IT systems, a model can be populated with appropriate data from a real system. Experiments conducted on a model in a way that does not disturb the functioning of the real system allow learning about the effects and selecting the most optimal decision variant concerning the quantities and types of inputs to the system.

The entire 19th century and most of the 20th century are characterized by predominance of mathematical modelling. Such methods usually allow obtaining good results from models, however the complexity and dynamics of contemporary enterprises cause that there is a need to look for new modelling methods such as artificial neural networks (ANN, NN). They belong to the group of so-called empirical modelling techniques, and the main issue associated with them is to provide measurement data. In the computerized environment of today's companies, a lack of data is not a problem, because most of them have different types of IT systems, while production processes are generality metered and automated. It can be said that companies often have a problem not with a lack of data but with their excess.

In turn, the market requirements and the dynamics of the business environment caused that new, faster methods of production system modelling are searched for. Such methods include models of artificial neural networks, which provide solutions that take into account the process of learning on the basis of available data. They allow skipping the stage of modelling complex relationships between elements of a production system. In order to build a model of a production system, it is enough to define the problem: selection of input and output variables, and a set of training data.

Artificial neural networks are usually used to solve problems related to approximation, interpolation, prediction, classification, recognition, and control. Image recognition, which also includes classification, grouping and processing, accounts for approx. 70% of all industrial applications. In the management and operation of production systems, artificial neural network are more and more often used for [1, 9, 10, 11]:

- control of production processes, robots,
- analysis of manufacturing problems,
- diagnostics of electronic systems of machines,
- selection of personnel and input materials,
- optimization of the business activity, waste disposal, robot movements,
- planning overhauls of machines,
- forecasting.

The primary objective of modelling the dynamics of a production process is to identify the temporal variability of its physical quantities or states. To this end, a time series, i.e. an ordered sequence of values of a certain variable over time, should be determined. A time series may have a form of a vector $[y(t_1), y(t_2), \dots, y(t_N)]$. Due to the fact that process parameters may differ in separate phases of the process, the time series vector can take the form of a vector defined in N-dimensional space. Individual components of this vector will be the states of the production process stages in the past, which in turn can be regarded as points in a multi-dimensional output space. Thus the task of analysing the temporal variability of the production process can be reduced to searching N-dimensional space for a certain trajectory, on which the analysed output variable of the process "moves". Therefore, a given quantity in the form of a time series is determined in order to predict its value in future moments [2, 10].

3 A Multilayer Perceptron Network for Determining the Stability of the Wire Harness Assembly Process

3.1 Characteristics of the Wiring Harness Assembly Process

A growing company producing wiring harnesses for Electrolux dishwashers planned to increase the production plan by 30 to 40% in the coming years to meet the increasing orders. An extension of the production program will lead to an increase in the load on production resources, while the production system will be more vulnerable to disturbances and risk factors occurring at the production line.

The analysed factory manufactures approximately 700 different types of products. All the products are characterized by a high similarity of the structure and the manufacturing process. Each wiring harness consists of so-called modules, while a module - of wires ended with terminals. Both the number of modules and the number of wires may differ depending on the type of wiring harness. Some wires may be joined with the use of insulating tape. Individual modules are connected in the enclosure.

Control in the analysed production system takes place according to the rules applicable for a pull control system. The process is stimulated by assembly centres. The assembly takes place in three centres operating in parallel. Material flow in the assembly centre is presented in

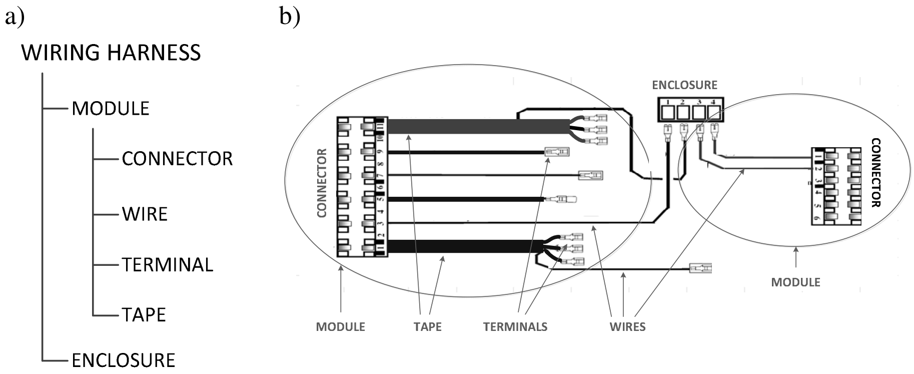


Fig. 2. a) Structure of a wiring harness, b) schematic diagram of the selected wiring harness

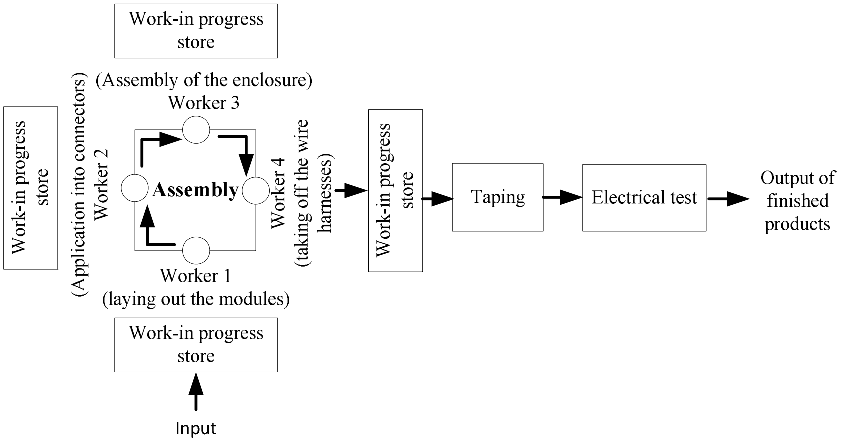


Fig. 3. Material flow in the assembly centre

Assembly operations are performed on a rotary table by three workers. Worker 1 lays out the prepared modules on the assembly table in accordance with the drawing of the product to be assembled, which is shown on the table. Worker 2 applies two additional wires into connectors, while Worker 3 inserts modules into enclosures. The wire harnesses are then taken from the table and transferred for the taping operation, after which an electrical test is carried out. If one of the modules or additional wires was not installed correctly, the product is considered defective.

All assembly operations require precision and high skills of the workers. An incorrect arrangement of modules causes a significant extension of time for application of additional wires in the subsequent operation. Both the operation of laying out the modules and taking off the wiring harnesses must be performed very carefully because the wires may slip out from the connectors or enclosure. Because final assembly centres operate on the principle of a swivel, the skills of the operators re very important as they need to work with the same pace.

3.2 The Purpose and Method of Building the Artificial Neural Network

The purpose of building the artificial neural network was to ensure the stability of a wiring harness assembly process. An assembly process can be deemed stable, if the production volume during a shift is consistent with the adopted production plan. Otherwise, corrective actions, which consist in changing the values of input parameters of the production resources used in the process, should be taken.

The assembly time of a wiring harness depends primarily on the number of the modules it consists of. This is associated both with the fact that more elements need to be assembled and taped. The boundaries of stability and the productivity of the assembly process for different numbers of modules were determined. These values are presented in Table 1.

Table 1. The productivity and the ranges of stability in the wiring harness assembly process depending on the number of modules

Number of modules in the wiring harness [pcs]	Production plan [pcs/shift]	Boundaries of stability [pcs/shift]
8 - 14	370	(360 – 380)
14 - 20	350	(340 – 360)
21 – 25	320	(310 – 330)
26 – 30	300	(290 – 310)

Basing on an observation of the process, an analysis of the documentation and interviews with the workers, it has been found that the factors affecting the assembly process include experience and skills of the workers designated in Fig. 3 as Workers 1, 2, 3 and 4. Therefore, the absences and turnover of workers are the factors that disturb the course of the assembly process the most. A new worker slows down the operation of an entire assembly centre and increases the number of defective elements found. It has been found that a new worker is able to acquire an adequate efficiency only after 1 month of work at a given assembly centre.

In order to predict the quantity of the products produced at the input parameters set, a unidirectional neural network (multilayer perceptron) was built. The quantity of assembled wiring harnesses of good quality, i.e. those which passed the electrical test successfully, was to be the dependent variable. The independent variables were selected as follows:

X_1 – the number of modules in the wire harness,
X_2 - the skills level of Worker 1,
X_3 - the skills level of Worker 2,
X_4 - the skills level of Worker 3,
X_5 - the skills level of Worker 4,
X_6 – taping time,
X_7 – the number of defective elements detected at the electrical test station.

In order to evaluate the parameter of workers' skills levels, 4 values have been introduced:

1 - a worker who works less than 1 week,
2 - a worker who works less than 2 week,
3 - a worker who works less than 4 week,
4 - an experienced worker.

The data were collected from observations and measurements of an actual process as well as from the analysis of the organizational documentation and quality control reports. In total, 378 measurements were available for each variable. This set was divided into two parts, one of which served as a training set, while the second part was used for testing the network.

The experiment was performed in the SAS Enterprise Miner 6.2 environment. The first step was to investigate the correlation between independent variables and the dependent variable. The results containing the correlation value are shown in Table 2.

Table 2. The values of the correlation between variables

Independent attribute (variable)	Correlation value
number of modules in the wiring harness	0.16583
the skills level of Worker 1	-0.16872
the skills level of Worker 2	-0.22465
the skills level of Worker 3	-0.14535
the skills level of Worker 4	0.03276
taping time	0.02104
the number of defective elements detected at the electrical test station	-0.02957

The obtained results indicate that it is pointless to use a linear regression method (absolute values of the correlation are below 0.5) for the analysed problem. Therefore it is justified to the use neural networks which build non-linear regression models.

As a part of further experiments, a model of a multilayer perceptron network was built, for which the values of the number of neurons in the hidden layer were changed. In order to confirm the results of the correlation analysis, a neural network according to the generalized linear model was also built.

For the neural network models built, a series of experiments for different numbers of independent variables was performed. Their goal was to establish the combination of independent variables, for which the neural network will provide the best prediction of the number of wiring harnesses manufactured per shift. When building the models, different numbers of independent variables were considered. Their selection was dictated by previous experiments, i.e. it depended on the absolute value of the correlation (see: Table 2). Under the experiment no. 1, all input attributes were used, while in the experiment no. 2 the 'taping time' attribute (the lowest absolute value of the correlation) was discarded. In the experiment no. 3 the 'number of defective elements found at the electrical test station' attribute (the next lowest absolute value of the correlation) was discarded in addition. The results are presented in Table 3, where the values obtained represent a network selection criterion, which is a mean square error. These results concern the analysis of the input data set, which was also used for network training process.

Table 3. The results of the experiments for different variants of the neural network built

Model of the neural network	**Mean squared error**			
	Experiment No. 1	Experiment No. 2	Experiment No. 3	Experiment No. 4
MPN – NN=3	999.05	2443.71	1056.1	427.08
MPN – NN=16	2537.86	1369.98	1437.86	1019.25
MPN – NN=32	327.08	767.69	375.39	526.14
MPN – NN=48	1219.25	754.22	**327.15**	2088.12
MPN – NN=64	2375.39	872.49	999.05	368.14
GLM	1851.50	1450.28	1851.50	2569.8

Where MPN - a multilayer perceptron network,
NN - number of neurons in the hidden layer,
GLM - generalized linear model.

The analysis of the results confirms that linear models are not suitable for solving this problem. For each experiment, the worst results (with the highest mean square error) were obtained for a neural network built according to the generalized linear model. The best results were obtained for a multilayer perceptron network with 48 neurons under the experiment no. 3, a schematic diagram of which is shown in Fig. 4.

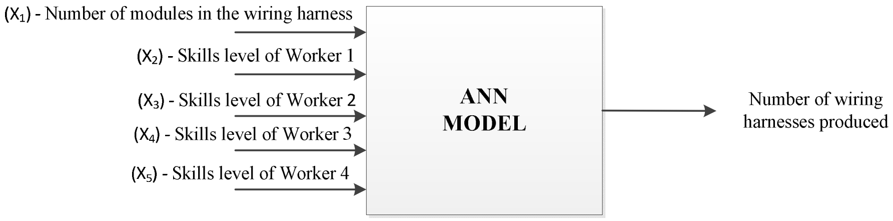

(X_1) - Number of modules in the wiring harness
(X_2) - Skills level of Worker 1
(X_3) - Skills level of Worker 2
(X_4) - Skills level of Worker 3
(X_5) - Skills level of Worker 4
ANN MODEL
Number of wiring harnesses produced

Fig. 4. Independent variables and the dependent variable used to build the artificial neural network (ANN)

The ANN model presented in Fig. 4 was used for further experiments, i.e. to assess the stability of the wiring harness assembly process for different values of independent variables.

3.3 Determination of the Stability of the Wiring Harness Assembly Process Using a Neural Network with 48 Neurons in the Hidden Layer

In order to determine the stability of the analysed process, test data were prepared and the "score" node of the SAS Enterprise Miner 6.2 environment was used. The test data contained various variants of changes in input attributes (independent variables). For such data, the selected neural network model predicts the values for the manufactured wiring harnesses, which are interpreted in the context of the stability of the assembly process. Sample test data along with the predicted number of manufactured elements are presented in **Błąd! Nie można odnaleźć źródła odwołania.**, and Table 5.

The purpose of the experiment no. 1 was to examine how the skills levels of the workers at the assembly centre affect the stability of the analysed process. For example, a wiring harness with 12 modules was selected. The production plan for wiring harnesses consisting of 12 modules is at the level of 370 pcs/shift. For the needs of the study it has been assumed that the production process is stable, if the absolute value of the difference between the quantity assumed in the production plan and the quantity produced does not exceed 20 pcs/shift, i.e. is within the range (360 - 380 pcs of wiring harnesses per shift). **Błąd! Nie można odnaleźć źródła odwołania.** shows that the production volume predicted by the ANN model depends on the skills level of Worker 3, assuming that the level of skills of other workers is high.

Table 4. The predicted production volume for different skills levels of Worker 3 and a fixed number of modules to be assembled

Network inputs					**Network outputs**
Number of modules [pcs]	Skills level of Worker 1	Skills level of Worker 2	Skills level of Worker 3	Skills level of Worker 3	Predicted production volume
12	4	4	4	3	370
12	4	4	3	3	362
12	4	4	2	3	341
12	4	4	1	3	289

The data included in **Błąd! Nie można odnaleźć źródła odwołania.** are presented additionally in the context of the process stability in Fig. 5.

As it results from **Błąd! Nie można odnaleźć źródła odwołania.** and Fig. 5, the process loses the steady state, if Worker 3 works for a period shorter than two weeks. This result confirms the observations made when collecting the data and analysing the process. It also confirms the opinions of the workers and process managers that only after a month of performing assembly operations, a new employee is able to work in accordance with the pace adopted for the assembly centre, and the number of defective products returns to the assumed level.

The purpose of the next experiment was to check whether the stability of the process would be similar for a larger number of modules in a wiring harness and at the same skills levels of the workers. A wiring harness consisting of 28 modules was used as an example. The production plan for this product was at the level of 300 pcs per shift, while the boundaries of the process stability were set at 290 - 310 pcs/shift.

Table 6**Błąd! Nie można odnaleźć źródła odwołania.** presents the production volume predicted by the ANN model for these assumptions.

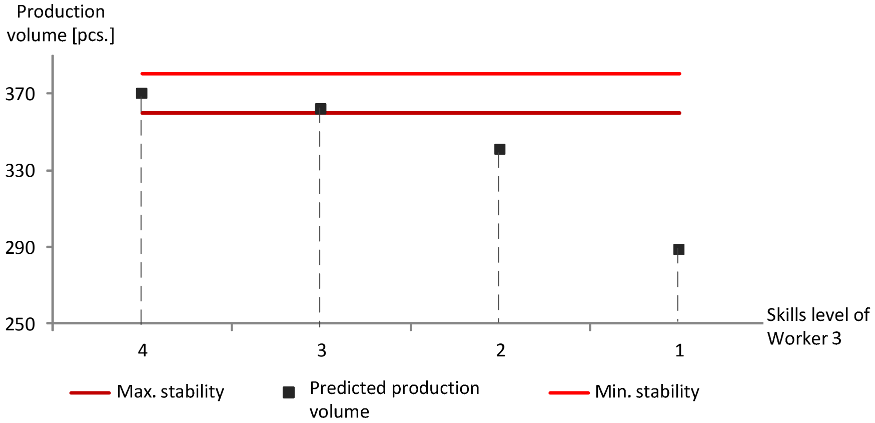


Fig. 5. The predicted production volume of wiring harnesses for a fixed number of modules and different skills levels of Worker 3

Table 5. The predicted production volume of wiring harnesses with 28 modules for different skills levels of Worker 3

Network inputs					**Network outputs**
Number of modules [pcs]	Skills level of Worker 1	Skills level of Worker 2	Skills level of Worker 3	Skills level of Worker 3	Predicted production volume
28	4	4	4	3	308
28	4	4	3	3	288
28	4	4	2	3	225
28	4	4	1	3	162

The data included in **Błąd! Nie można odnaleźć źródła odwołania.** are presented additionally in the context of the process stability in Fig. 6.

As it appears from **Błąd! Nie można odnaleźć źródła odwołania.** and Fig. 6 at the same skills levels of the workers but for an increased number of modules in a wiring harness, the process is stable only if all workers at the assembly centre are experienced, i.e. work at this centre for more than 4 weeks. If Worker 3 works at this centre for a period shorter than 4 weeks, the process is no longer stable. The results of the experiment confirm the observations and previous assumptions that only experienced workers (i.e. those who work at a given centre for more than 4 weeks) are able to work in accordance with the adopted pace, and the number of defective elements manufactured by them decreases to a minimum.

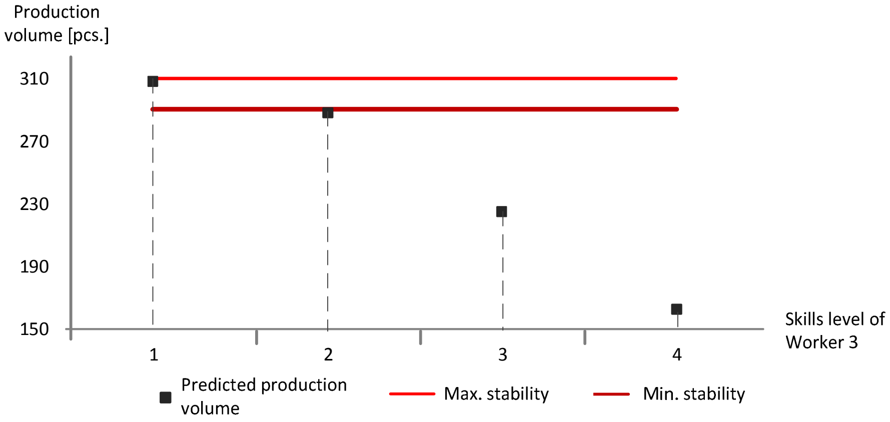


Fig. 5. The predicted production volume for wiring harnesses with 28 modules for different skills levels of Worker 3

4 Summary

Models of artificial neural networks can be used to control a production system, and thus to ensure its stability. Such models are very useful tools, because they can be built quickly and easily. The only issue is a large amount of data needed in the neural network training process. However, in the era of common availability of IT systems, the parameterization and standardization of production processes is not a problem anymore.

The development of IT systems led to an increase in the amount of information collected in enterprises. In turn, the market requirements and the dynamics of the business environment caused that new, faster methods of production system modelling are searched for. Such methods include models of artificial neural networks, which provide solutions that take into account the process of training on the basis of available data. They allow skipping the stage of modelling complex relationships between elements of a production system. In order to build a model of a production system, it is enough to define the problem, select input and output variables, and prepare a training data set.

Artificial neural networks (ANN) constitute a modern method for mathematical modelling of phenomena and processes, which in the last dozen or so years found many applications in different areas of human activity. The reason why ANN are used in the science and engineering is the possibility to treat dynamic, complex variables or imprecisely defined processes as a black box. In other words, ANN allow finding a relationship between many input variables and the output variable in the process without a need to build complex mathematical equations.

References

1. Bilski, J.: The UD RLS algorithm for training feedforward neural networks. International Journal of Applied Mathematics and Computer Science 15(1), 115–123 (2005)
2. Bilski, J., Rutkowski, L.: A fast training algorithm for neural networks. IEEE Trans. Circuits Syst. II 45(6), 749–753 (1998)
3. Burduk, A., Chlebus, E.: Methods of risk evaluation in manufacturing systems. Archives of Civil and Mechanical Engineering 9(3), 17–30 (2009)
4. Frumusanu, G., Epureanu, A., Ionut, C.: Cutting process stability evaluation by process parameters monitoring. In: NOLASC 2009 Proceedings of the 8th WSEAS International Conference on Non-linear Analysis, Non-linear Systems and Chaos, vol. 1, pp. 345–350 (2009)
5. Chlebus, E., Burduk, A., Kowalski, A.: Modelling and computer simulation as tools for reorganization of production processes. In: HAIS 2011 Proceedings of the 6th International Conference Hybrid Artificial Intelligent Systems, Wroclaw, Poland (2011)
6. Chlebus, T., Stefaniak, P.: The concept of intelligent system for horizontal transport in a copper ore mine. In: Corchado, E., Snášel, V., Abraham, A., Woźniak, M., Graña, M., Cho, S.-B. (eds.) HAIS 2012, Part II. LNCS, vol. 7209, pp. 267–273. Springer, Heidelberg (2012)
7. Mahmood, K., Zidouri, A., Zerguine, A.: Performance analysis of a RLS-based MLP-DFE in time-invariant and time-varying channels. Digital Signal Processing 18, 307–320 (2008)
8. Sankar, N., Prabhu, B.: Modified approach for prioritization of failures in a system failure mode and effects analysis. International Journal of Quality & Reliability Management 18(3), 324–336 (2001)
9. Rutkowski, L.: Methods and Techniques of Arificial Inteligence. PWN, Warszawa (2009)
10. Wieczorek, T.: Neural models of technological processes, Monograph. Publishing House of the Silesian University of Technology, Gliwice (2008)
11. Tung-Hsu, H., Wang-Lin, L., Li, L.: Intelligent remote monitoring and diagnosis of manufacturing processes using an integrated approach of neural networks and rough sets. Journal of Intelligent Manufacturing 18(2), 239–253 (2003)

Generalized Predictive Control for a Flexible Single-Link Manipulator

Rahma Boucetta

Control and **E**nergy **M**anagement **Lab**oratory (CEM Lab)
University of Sfax, Sfax Engineering School, BP W, 3038, Sfax, Tunisia
rboucetta@yahoo.fr

Abstract. This paper presents the development of a generalized predictive controller applied to a flexible single-link manipulator robot to compare to a fuzzy supervisory controller in input tracking and end-point vibration suppression. A dynamic model of the flexible manipulator is derived using finite elements method and Lagrange's equations to determine dynamics behavior. A generalized predictive controller is then developed and introduced in the system closed-loop to minimize end-point residual vibrations. A fuzzy supervisory controller is also synthesized to compare simulation results between the two methods of control in terms of input tracking and disturbance rejection.

Keywords: Flexible manipulator robot, dynamic model, generalized predictive control, fuzzy supervisory control.

1 Introduction

Robotic manipulators are generally built using heavy material to maximize stiffness, in an attempt to minimize system vibration and achieve good positional accuracy. As a consequence, such robots are usually heavy with respect to the operating payload. The operation speed of the robot manipulation is limited, so the actuators size is increased boosting energy consumption and increasing the overall cost. Moreover, the robot has a low payload to robot weight ratio. In order to solve these problems, robotic manipulators are designed to be lightweight.

Conversely, flexible manipulators exhibit many advantages over their rigid counterparts: they require less material, are lighter in weight, have higher manipulation speed, lower power consumption, require smaller actuators, are more maneuverable and transportable, are safer to operate due to reduced inertia, have enhanced back-drive ability due to elimination of gearing, have less overall cost and higher payload to robot weight ratio.

However, the control of flexible robotic manipulators to maintain accurate positioning is an extremely challenging problem. Due to the flexible nature and distributed characteristics of the system, the dynamics are highly non-linear and complex. Problems arise due to precise positioning requirement, vibration due to system

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 499–510, 2013.

flexibility, difficulty in obtaining an accurate model and non-minimum phase characteristics of the system. Therefore, flexible manipulators have not favored in production industries, due to un-attained end-point positional accuracy requirements in response to input commands. Thus, the design of control algorithms for flexible systems possessing nonlinear time-varying and ill-modeled dynamics presents great challenges for all conventional methodologies.

M.A. Arteaga and B. Siciliano collected a number of recent results on modeling, nonlinear control and observer for flexible-link manipulators [8]. S.S. Ge proposed energy-based robust control strategies for the control of flexible link robots without using the dynamics of the systems explicitly [8]. O. Al Jarrah, Y.F. Zheng and K.-Y. Yi presented three approximation methods of the optimal trajectories and a compliant control scheme [8]. R.N. Banavar and P. Dominic applied an LQG/H_∞ controller for a flexible manipulator [11]. S.B. Choi and J.W. Cheon proposed a vibration control of a single-link flexible arm subjected to disturbances [12]. A neural network control is developed by C.-F.J. Kuo and C.-J. Lee for a rotating elastic manipulator [13]. L. Tian and C. Collins proposed firstly a dynamic recurrent neural network-based controller for a rigid-flexible manipulator system [14] and secondly an adaptive neuro-fuzzy control for a flexible manipulator [15]. A self-organizing fuzzy logic controller is used by G.L.C.M. de Abreu and J.F. Ribeiro for the active control of flexible structures using piezoelectric actuators [9]. Soft computing methods are applied to the control of a flexible robot manipulator by B. Subudhi and A.S. Morris [10]. It is in the last area that our contribution can be located.

The main purpose of this work is to determine the adequate controller to minimize human intervention and to increase performance responses. So, this paper is organized as follows: Section 2 describes the flexible manipulator system and how to derive its dynamic model using Lagrange's equations and finite elements method. The open-loop analysis is given in section 3. Section 4 presents the GPC controller applied to the flexible single-link manipulator. Section 5 developed the supervisory fuzzy controller given to the system. Finally, a comparative assessment of performances between the different strategies in terms of vibration suppression and input tracking is presented and discussed.

2 The Flexible Manipulator System

A schematic representation of the single-link flexible manipulator system is shown in figure 1, where a control torque $\tau(t)$ is applied at the hub of a motor with E, I, ρ, L and I_H represent Young's modulus, second moment of area, mass density per unit volume, length, and hub inertia moment respectively [1].

The angular displacement of the link in the X_0OY_0 coordinates is denoted by $\theta(t)$. w(x,t) represents the elastic deflection of the manipulator at a distance x from the hub, measured along the OX axis. X_0OY_0 and XOY represent the stationary and moving frames respectively.

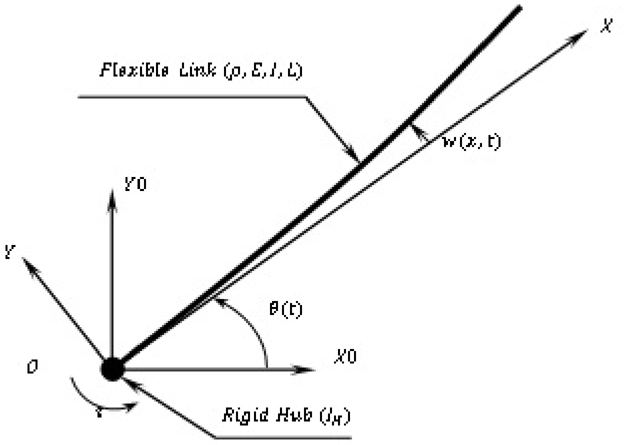


Fig. 1. Flexible manipulator scheme

The height (width) of the link is assumed to be much greater than its depth, thus allowing the manipulator to vibrate dominantly in the horizontal direction (X_0OY_0 plane). To avoid difficulties arising from time varying lengths, the length of the manipulator is assumed to be constant. Moreover, the shear deformation, the rotary inertia and the effect of axial force are ignored. For an angular displacement θ and an elastic deflection w, the total displacement y(x,t) of a point along the manipulator at a distance x from the hub can be described as a function of both the rigid body motion θ(t) and the elastic deflection w(x,t), i.e. [3, 7]

$$y(x,t) = x.\theta(t) + w(x,t) \tag{1}$$

Thus, by allowing the manipulator to be dominantly flexible in the horizontal direction, the elastic deflection of the manipulator can be assumed to be confined to the horizontal plane only.

Kinetic energy of the flexible manipulator, depending of hub rotation, modes rotation in the X0OY0 and XOY frames, has the following expression

$$T = \frac{1}{2} I_H \dot{\theta}^2 + \frac{1}{2}(\dot{q} + L\dot{\theta})^T M(\dot{q} + L\dot{\theta}) \tag{2}$$

Potential energy just depending of link flexibility has the form

$$V = \frac{1}{2} q^T K q \tag{3}$$

After applying Lagrange's equations, the dynamic model can be written as

$$(I_H + L^T M L)\ddot{\theta} + L^T M \ddot{q} = \tau \tag{4}$$

$$M\ddot{q} + L^T M \ddot{\theta} + Kq = 0 \tag{5}$$

where M, K and L are the mass matrix, the stiffness matrix and the length array respectively, and q is the elastic modes vector.

3 Dynamic Behavior

The dynamic equations can be presented in a state-space form as

$$\dot{v} = Av + Bu$$

$$y = Cv + Du$$

where the state-space matrices are

$$A = \begin{pmatrix} 0 & I \\ -M^{-1}K & 0 \end{pmatrix}, B = \begin{pmatrix} 0 \\ M^{-1} \end{pmatrix}, C = (I \quad 0), D = (0)$$

The state and control vectors are given by

$$v^T = (\theta \quad q_1 \quad q_2 \quad ... \quad \dot{\theta} \quad \dot{q}_1 \quad \dot{q}_2 \quad ...)$$

$$u^T = (\tau \quad 0 \quad ...)$$

In order to simulate the flexible manipulator system, an aluminum link of dimensions L=0.61m and S=3×10^{-5}m², with E=200×10^{9}N/m², I=2.5×10-^{12}m^{4}, I$_{H}$=4.3×10^{-3}Kg.m² and ρ=7.8×103Kg/m^{3} is considered [16]. The link is discretized into two elements.

Solving the state-space matrices gives the vector of states v, that is, the hub angle, the elastic modes and their velocities. The derived dynamic model is a nonminimum phase system, not strictly proper, and unstable. Also, the model has zeros very close to the imaginary axis; this deteriorates the time domain performance of the closed-loop system.

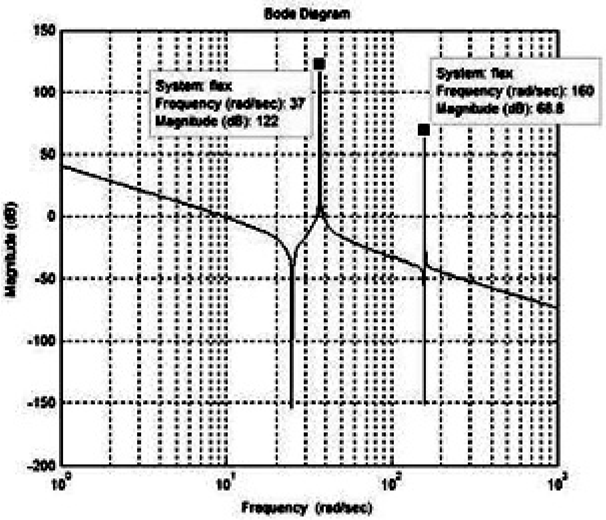


Fig. 2. Open-loop frequency response

Generally, linear models of flexible structures used in design of controllers are derived under restrictive assumptions which are often not valid for large motions that occur during slewing maneuvers. Hence, considerable uncertainty in the linear model exists. Another feature characteristic of lightly damped systems is the occurrence of poles (±159.67j, ±37j, 0, 0) and zeros (±158j, ±25j) very close to the imaginary axis that gives rise to ill-conditioned systems. The state-space matrices arising out of such systems have largely separated singular values, posing considerable computational difficulty in controller design. In the spectral density given by figure 2, the vibration frequencies of the system are obtained as 37rad/s and 160rad/s, i.e. 5.9Hz and 25.46Hz, and the magnitude of frequency response for the two resonance modes are 122dB and 68.8dB.

4 Generalized Predictive Control

4.1 ARIMAX Model

The Generalized Predictive Controller was introduced by Clarke and al. in 1987 [22]. The computation of the output predictions supposes the knowledge of a model of the system that is in the ARIMAX form:

$$A(q^{-1})y(t) = B(q^{-1})u(t - T_e) + \frac{C(q^{-1})}{\Delta(q^{-1})}\xi(t) \tag{6}$$

where:

- q^{-1} is the backward operator
- $y(t)$ is the output signal
- $u(t)$ is the input signal
- $\xi(t)$ is the disturbance (white noise) process with $E\big(\xi(t)\big) = 0$
- A and C are $p \times p$ monic polynomial matrices where C can be used to model a colored noise. To simplify the problem, we consider that $C = c(q^{-1})I_{p\times p}$ where c is a monic polynomial
- B is a $p \times m$ polynomial matrix
- The operator Δ is defined as $\Delta = 1 - q^{-1}$ used to make noise be non-stationary, which suitable to model any perturbation in a control loop.

The polynomial matrices A, B, C are respectively of order n_a, n_b, and n_c:

$$A(q^{-1}) = I_{p\times p} + A_1 q^{-1} + \cdots + A_{na} q^{-na}$$

$$B(q^{-1}) = B_0 q^{-d} + B_1 q^{-d-1} + \cdots + A_{nb} q^{-nb-d}$$

$$C(q^{-1}) = I_{p\times p} + C_1 q^{-1} + \cdots + C_{nc} q^{-nc}$$

where d is a positive integer representing the delay of the system. The operator $\Delta(q^{-1})$ allows the rejection of constant perturbations and is equivalent to the introduction of an integral action in the controller.

4.2 Cost Function

The GPC is based on the minimization of a cost function J over a finite receding horizon [18, 19, 21]:

$$J = E\left\{\sum_{j=N_1}^{N_2} \|y(t+jT_e) - r(t+jT_e)\|^2 + \lambda \sum_{j=0}^{N_u-1} \|\Delta u(t+jT_e)\|^2\right\} \tag{7}$$

with $N_u < N_2$ and $\Delta u(t+jT_e) = u(t+jT_e) - u(t+(j-1)T_e) = 0\ \forall j \geq N_u$ where λ is a positive scalar and N_1, N_2 and N_u are positive integers defined as follows:

- N_1 is the minimum costing horizon,
- N_2 is the maximum costing horizon,
- N_u is the length of the control cost horizon,
- λ weights the relative importance of the control energy,
- T_e is the sampling time
- $r(t)$ is the reference trajectory

The aim is to compute the N_u future control increments, so as to drive the actual system outputs towards the theoretical ones, or, equivalently, to compensate for the measurement disturbances. This controller is predictive because it takes into account the future references. Indeed, the minimization of (6) requires the computation of N_2 predictions of the output using the future reference signals. The arguments of the minimization are the N_u future steps of the control input.

4.3 Control Computing

The minimization of the cost function J on a finite horizon allows us to compute the predictor on j optimal number of sampling periods to predict real output in $(k-j)$ sample periods. The vector $\underline{Y_c}$ is setting as

$$\underline{Y_c} = [y_c(k+HI), \dots, y_c(k+HP)]^T \tag{8}$$

The prediction depending of past and present measures $\underline{E\hat{Y}_a}$ is defined as

$$\underline{E\hat{Y}_a} = \underline{\hat{Y}_a} - \underline{Y_c} = R^*\underline{\Delta U}^* + G^*Y^* - \underline{Y_c} \tag{9}$$

Similarly, the prediction depending of the sequence of future control increments $\underline{E\hat{Y}_p}$ minimizing the cost function is given by

$$\underline{E\hat{Y}_p} = Q^*\underline{\Delta U_p} \tag{10}$$

The prediction margin ranged from HI to HP, and the system delay is known a priori, so, the $(d-1)$ first lines of matrices G^*, R^* and Q^* are not taken into account, and eq. 9 and eq. 10 are rewritten as

$$\underline{E\hat{Y}_a} = R^{**}\underline{\Delta U}^* + G^{**}\underline{Y}^* - \underline{Y_c}$$

$$\underline{E\hat{Y}_p} = Q^{**}\underline{\Delta U_p}$$

The aim of control is to compute optimal control sequence $\underline{\Delta U_p}$ minimizing the cost function. Length of this vector is reduced to $(HC, 1)$, and $(HP - HC)$ last columns of Q^{**} matrix are not taken into account. Consequently, a new matrix Q_* and a new expression of $\underline{E\hat{Y}_p}$

$$\underline{E\hat{Y}_p} = Q_*\underline{\Delta U_p}$$

$$\underline{\hat{Y}} = \underline{\hat{Y}_p} + \underline{\hat{Y}_a} = Q_*\underline{\Delta U_p} + R^{**}\underline{\Delta U^*} + G^{**}\underline{Y^*}$$

The cost function has the following matrix form

$$J = [Q_*\underline{\Delta U_p} + \underline{E\hat{Y}_a}]^T\left[Q_*\underline{\Delta U_p} + \underline{E\hat{Y}_a}\right] + \underline{\Delta U_p^T}\lambda I_{HC}\underline{\Delta U_p}$$

The minimal value of the cost function is

$$\frac{\partial J}{\partial \underline{\Delta U_p}} = 2Q_*^T\left[Q_*\underline{\Delta U_p} + \underline{E\hat{Y}_a}\right] + 2\lambda\underline{\Delta U_p} = 0$$

The optimal control for the minimal cost function has the following form

$$\underline{\Delta U_p} = -[Q_*^T Q_* + \lambda I_{HC}]^{-1} Q_*^T [R^{**}\underline{\Delta U^*} + G^{**}\underline{Y^*} - \underline{Y_c}$$

$$= -M[R^{**}\underline{\Delta U^*} + G^{**}\underline{Y^*} - \underline{Y_c}]$$

with

$$M = [Q_*^T Q_* + \lambda I_{HC}]^{-1} Q_*^T = \begin{bmatrix} m_1^T \\ m_2^T \\ \vdots \\ m_{HC}^T \end{bmatrix}$$

Finally, the resulted control signal is the first element of the control vector $\underline{\Delta U_p}$, given by the following expression

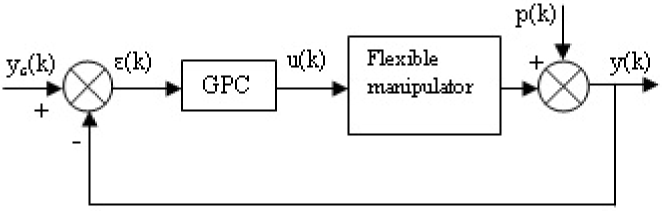


Fig. 3. GPC control scheme

$$u(k) = \Delta u(k) + u(k-1)$$

with

$$\Delta u(k) = -m_1^T[R^{**}\underline{\Delta U}^* + G^{**}\underline{Y}^* - \underline{Y_c}]$$

5 Fuzzy Supervisory Control

A highest level supervisor uses any available data from the control system to characterize the system's current behavior so that it knows how to change the controller and ultimately achieve the desired specifications. In addition, the supervisor can be used to integrate other information into the control decision-making process. It can incorporate certain user inputs, or inputs from other subsystems [2]. Conceptually, the design of the supervisory controller can then proceed in the same manner as it did for direct fuzzy controllers [18]: either via the gathering of heuristic control knowledge or via training data that we gather from an experiment. The type of heuristic knowledge that is used in a supervisor may take one of the following two forms

- Information from a human control system operator who observes the behavior of an existing control system and knows how this controller should be tuned under various operating conditions.
- Information gathered by a control engineer who knows that under different operating conditions controller parameters should be tuned according to certain rules.

A higher level of control can be achieved for monitoring and adjusting the direct fuzzy controller. The expert controller expands or compresses the universes of discourse by simply changing the scaling gains. When the universe is expanded, a coarse control given by the table 1 is achieved, and when it is compressed, a fine control given by the table 2 is achieved. The supervisor would be a fuzzy system that can gradually rather than abruptly switch between the two conditions using a Sugeno fuzzy system based on the following rule-base:

- If error is negative (positive) Then control action is coarse
- If error is zero Then control action is fine

Table 1. Rule-base for the coarse control

u		ε'						
		-3	-2	-1	0	+1	+2	+3
ε	-3	-4	-3	-3	-2	-2	-1	0
	-2	-3	-3	-2	-2	-1	0	+1
	-1	-3	-2	-2	-1	0	+1	+2
	0	-2	-2	-1	0	+1	+2	+2
	+1	-2	-1	0	+1	+2	+2	+3
	+2	-1	0	+1	+2	+2	+3	+3
	+3	0	+1	+2	+2	+3	+3	+4

Table 2. Rule-base for the fine control

u		ε'						
		-3	-2	-1	0	+1	+2	+3
ε	-3	-4	-4	-3	-3	-2	-1	0
	-2	-4	-3	-3	-2	-1	0	+1
	-1	-3	-3	-2	-1	0	+1	+2
	0	-2	-1	0	0	0	+1	+2
	+1	-2	-1	0	+1	+2	+3	+3
	+2	-1	0	+1	+2	+3	+3	+4
	+3	0	+1	+2	+3	+3	+4	+4

6 Simulation Results

To study the GPC and supervisory control performances and to compare simulation results in terms of input tracking, vibration suppression and disturbance rejection, the GPC controller is introduced in the closed-loop position control of the flexible single-link manipulator. Sampling time is chosen 0.01s.

Figure 5 depicts the time evolutions of both consign and hub angle responses. A transient appears at t=10s, corresponding to the start of the step consign chosen $pi/4$.

The control signal can compensate the fast change of consign at t=10s. So, the hub angle output varies before the consign signal and has very fast rise explaining the appearance of an overshoot peak over the consign value. After that, the plant output oscillates around the desired value with very low vibration magnitude.

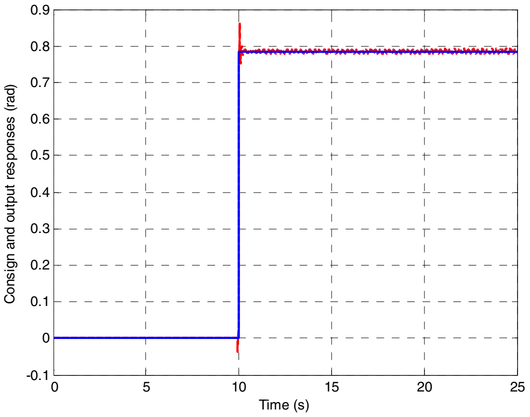


Fig. 4. Consign and output responses without disturbances

In figure 6, an impulse disturbance with amplitude of $pi/3$ is applied at t=15s, the plant output receives a small variation and continues with the same specifications.

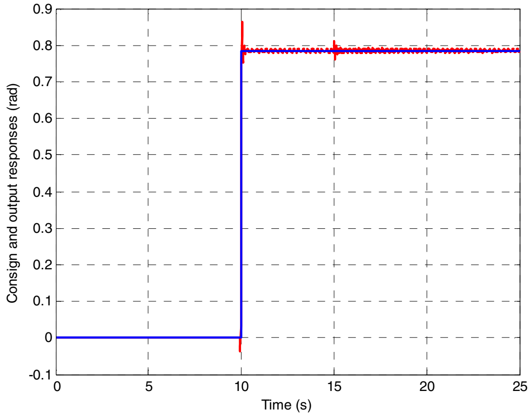


Fig. 5. Consign and output responses with disturbance at t=15s

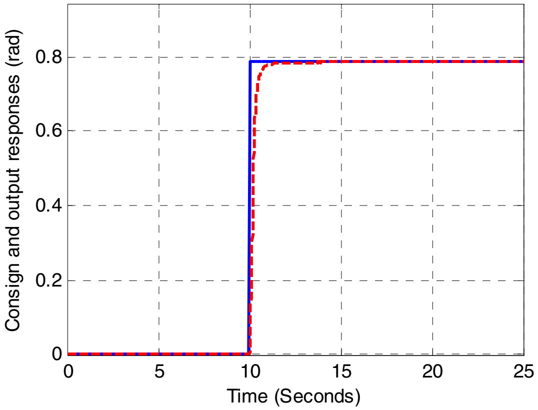


Fig. 6. Consign and output responses with supervisory controller

Figure 7 gives the hub angle response deduced from the fuzzy supervisory control, the output response has a fast transient phase to reach consign without overshoot, with rise time and settling time about 0.5s.

The introduction of a disturbance at t=15s in the fuzzy supervisor control case yields the result of figure 8, we can notice that the hub angle receives a slight peak and returns to its stable state.

For the two controllers, we can easily noticed that the GPC controller gives a fast hub angle response with a good input tracking but it has an overshoot in the rise, contrary to the fuzzy supervisory controller that has a slower rise time compared to the GPC, without vibrations or overshoot. The GPC controller has a good disturbance rejection against the fuzzy supervisor.

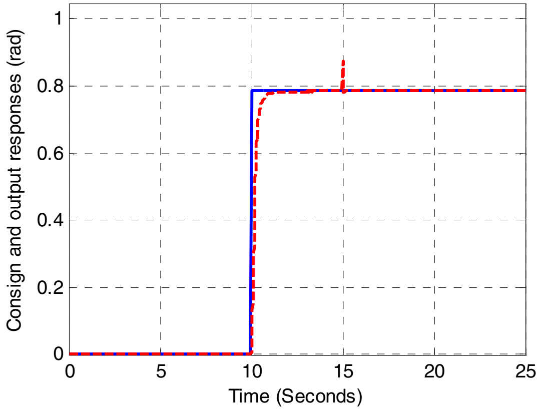


Fig. 7. Consign and output responses with supervisory controller and disturbance

7 Conclusion

A comparative assessment between the GPC and the fuzzy supervisory control strategies applied to a flexible single-link manipulator has been presented in this paper. A dynamic model of the flexible manipulator system is first derived using Lagrange's equations and finite element method. A GPC controller is then introduced in the closed-loop of the flexible system. Next, a fuzzy supervisory controller is developed to be incorporated in the system closed loop to increase level of desired performances. Simulation results are compared in terms of input tracking, disturbance rejection and vibration reduction.

References

1. Meirovitch, L.: Elements of Vibration Analysis. McGraw-Hill International Book Company (1982)
2. Passino, K.M., Yurkovich, S.: Fuzzy Control. Addison-Wesley Longman (1998)
3. Wolovich, W.A.: Automatic Control Systems, Basic Analysis and Design, International Edition (1994)
4. Md Zain, M.Z., Tokhi, M.O., Mailah, M., Mohamed, Z.: Improving Performance in Single-link Flexible Manipulator using Hybrid Learning Control. Jurnal Mekanikal, Bil. 18, 13–18 (2004)
5. Knani, J.: Contribution à la Commande Adaptative des Robots Manipulateurs Rigides et Flexibles. Habilitation Universitaire, ENIT (2003)
6. Azad, A.K.M., Tokhi, M.O., Anand, N.: Teaching of Control for Complex Systems through Simulation. In: Proceedings of the 2003 ASEE/WFEO International Colloquium. American Society for Engineering Education (2003)
7. Moberg, S.: On Modeling and Control of Flexible Manipulators, Thesis, Division of Automatic Control, Department of Electrical Engineering, Linköping Studies in Science and Technology, Sweden (2007)

8. Wang, F.-Y., Gao, Y.: Advanced Studies of Flexible Robotic Manipulators, Modeling, Design, Control and Applications. World Scientific Edition (2003)
9. de Abreu, G.L.C.M., Ribeiro, J.F.: A self-organizing fuzzy logic controller for the active control of flexible structures using piezoelectric actuators. Applied Soft. Computing, 271–283 (2002)
10. Subudhi, B., Morris, A.S.: Soft computing methods applied to the control of a flexible robot manipulator. Applied Soft. Computing, 149–158 (2009)
11. Banavar, R.N., Dominic, P.: An LQG/H Controller for a Flexible Manipulator. IEEE Transactions on Control Systems Technology 3(4) (December 1995)
12. Choi, S.B., Cheon, J.W.: Vibration control of a single-link flexible arm subjected to disturbances. Journal of Sound and Vibration 271, 1147–1156 (2004)
13. Kuo, C.-F.J., Lee, C.-J.: Neural Network Control of a Rotating Elastic Manipulator. Computers and Mathematics with Applications 42, 1009–1023 (2001)
14. Tian, L., Collins, C.: A dynamic recurrent neural network-based controller for a rigid-flexible manipulator system. Mechatronics 14, 471–490 (2004)
15. Tian, L., Collins, C.: Adaptive neuro-fuzzy control of a flexible manipulator. Mechatronics 15, 1305–1320 (2005)
16. Boucetta, R., Chabir, A., Chelly, N., Abdelkrim, M.N.: Building and Control of a Flexible One-link Manipulator. In: The 1st IAA Mediterranean Astronautical Conference, Tunis Science City (November 17-19, 2008)
17. Boucetta, R., Bel Hadj Ali, S., Abdelkrim, M.N.: On the Fuzzy Control of a Flexible Single-link Manipulator. In: SSD 2011, Sousse, Tunisia (March 2011)
18. Abdelkrim, R., Boucetta, R.: Compensation des mouvements respiratoires par la commande prédictive généralisée à modèle répétitif d'un robot médical. In: CRATT 2013 (2013)
19. Abdelkrim, R.: commande par retour visuel d'un robot médical, projet de fin d'étude, Université de Gabès, ENIG (2012)
20. Ott, L., Zanne, P., Nageotte, F., de Mathelin, M., Gangloff, J.: Physiological Motion Rejection in Flexible Endoscopy Using Visual Servoing. In: IEEE International Conference on Robotics and Automation, Pasadena, Californie USA (2008)
21. Ginhoux, R.: Compensation des mouvements physiologiques en chirurgie robotisée par commande prédictive, Thèse, Université de Louis Pasteur Strasbourg (2003)
22. Clarke, D.W., Mohtadi, C.: Properties of Generalized Predictive Control, vol. 25, pp. 859–875. Pergamon Press, International Federation of Automatic Control (1989)

A Disruption Recovery Model in a Production-Inventory System with Demand Uncertainty and Process Reliability

Sanjoy Kumar Paul, Ruhul Sarker, and Daryl Essam

School of Engineering and Information Technology
University of New South Wales, Canberra, Australia
sanjoy.paul@student.adfa.edu.au,
{r.sarker,d.essam}@adfa.edu.au

Abstract. This paper develops a risk management tool for a production-inventory system that involves an imperfect production process and faces production disruption and demand uncertainty. In this paper, the demand uncertainty is represented as fuzzy variable and the imperfectness is expressed as process reliability. To deal with the production scheduling in this environment, a non-linear constrained optimization model has been formulated with an objective of maximizing the graded mean integration value (GMIV) of the total expected profit. The model is applied to solve the production-inventory problem with single as well as multiple disruptions on a real time basis that basically revises the production quantity in each cycle in the recovery time window. We propose a genetic algorithm (GA) based heuristic to solve the model and obtain an optimal recovery plan. A numerical example is presented to explain usefulness of the developed model.

Keywords: Production inventory, series of disruptions, demand uncertainty, process reliability, genetic algorithm.

1 Introduction

Batch production is a well accepted technique in advanced manufacturing and logistics management system. Production lot size is determined to minimize the costs of the system. There are numerous industries, such as pharmaceutical, textile and food, that produce the products using the batch production systems. There are several risks factors in real life problems which should be taken into consideration when production system is analyzed. Production disruptions i.e. raw material shortage, machine breakdown, labor strike, or any other production interruptions are very common scenario in the production systems. Moreover, it is very difficult to find the production process that produces 100% non-defective products. So the production process reliability, can be less than 100%, is also an important factor because of the imperfect production process. In real life situations, product demand cannot be known with uncertainty. In this paper, the process reliability and demand uncertainty are considered with production disruption to make the research problem very close to the practical scenario.

K. Saeed et al. (Eds.): CISIM 2013, LNCS 8104, pp. 511–522, 2013.

Over the last few decades, one of the most widely studied research topics, in operations research and industrial engineering, is the production inventory system. Few examples of such studies in single stage production inventory system include a single-item inventory system with non-stationary demand process [1], determination of lot size and order level for a single-item inventory model with a deterministic time-dependent demand [2], a single-item periodic review stochastic inventory system [3] and a single-item single-stage inventory system with stochastic demand in a periodic review where the system must order either none or at least as much as a minimum order quantity [4].

The above studies, with many others, are conducted under ideal conditions. However, production disruption is a very familiar event in the production environment. Production disruption is defined as any form of interruption that may cause due to shortage of material, machine breakdown and unavailability, or any other form of disturbance. The development of an appropriate recovery policy can help to minimize the loss and maintain the goodwill of the company. Lin and Gong [5] analysed the impact of machine breakdown on EPQ model for deteriorating items in a single stage production system with fixed period of repair time. Widyadana and Wee [6] extended the model of Lin and Gong [5] for deteriorating items with random machine breakdown and stochastic repair time with uniform and exponential distribution. A disruption recovery model for single stage and single item production system is developed by Hishamuddin et al. [7] and the model was formulated for a single disruption, for recovering within a given time window, considering back order as well as lost sales option. Recently, a transportation disruption recovery model in a two-stage production and inventory system was developed by Hishamuddin et al. [8]. In the production and inventory modelling, numerous studies have been performed considering supply disruptions. Parlar and Perry [9] developed inventory models considering supplier availability with deterministic product demand under the continuous review framework. Özekici and Parlar [10] considered back orders to analyse a production-inventory model under random supply disruptions modelled as a Markov chain. Recently, other models of supply disruptions considering deterministic product demand in the inventory models have been studied [11] and [12].

There are some recent studies, where reliability of the imperfect production process has been considered. At first, process reliability is considered by Cheng [13] in a single period inventory system and formulated as unconstrained geometric programming problem. Later, it was extended in [14] by considering fuzzy random demand. Later, process reliability of the imperfect production process was incorporated to determine the optimal product reliability and production rate that achieved the biggest total integrated profit [15], to study unreliable supplier in a single-item stochastic inventory system [11] and to analyze an EPQ model with price and advertising demand pattern under the effect of inflation [16].

Many authors considered only fuzzy characteristics of the variables to tackle uncertainty. Lee and Yao [17] introduced fuzzy senses in the EPQ model considering demand and production per day as fuzzy variables. Later fuzzy product quantity as a triangular fuzzy number [18], order quantity and total demand quantity as triangular fuzzy numbers Yao et al. [19], demand as a fuzzy random variable in a single-period inventory model [20] are considered in developing production inventory modelling. Recently, Islam and Roy [21] developed a modified geometric programming program in an EPQ model under storage space constraint and reliability of the production process considering inventory related costs, storage spaces and others parameters as triangular fuzzy number.

In the previous studies of production inventory modelling, no study considered demand uncertainty and process reliability to develop a disruption recovery model. Also most of the previous studies focused on developing recovery plan only from single disruption. In this paper, a real time disruption recovery model is developed where the production process faces single or multiple disruptions. Other risk factors, process reliability and demand uncertainty, are also incorporated to make the model realistic. Finally, a genetic algorithm based heuristic is proposed to solve the model with single or multiple disruptions on a real time basis.

2 Problem Definitions

In the real life production system, disruptions are very common scenario and it can happen at any time at any point. It needs to develop an optimal plan to recover from those production disruptions. Revision in the production quantities and use of the idle timeslot of the systems are the significant ways to obtain the recovery plan [7]. After each disruption, production quantities in each cycle during the recovery period are revised. We develop a solution approach to obtain the recovery plan that deals with single or series of disruptions on a real time basis.

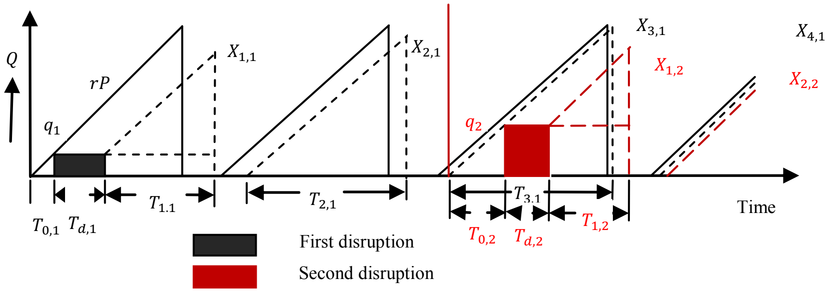


Fig. 1. Disruption recovery plan

The recovery plan after a production disruption is presented in Figure 1. The recovery plan is a new schedule which includes the revised production quantities in each cycle to maximize the total profit in the recovery period. Number of cycles allocated to return to the original production schedule from the disrupted cycle is known as recovery period. The first disrupted production cycle is considered as first cycle (l=1). Now an optimal recovery plan is proposed to revise the production quantities $X_{i,1}$ $(i = 1,2, \ldots, M)$ to recover from that disruption which is shown as black dashed line in the Figure 1. Again after the second disruption within the recovery period of previous disruption, production quantities in each cycle $X_{i,2}$ $(i = 1,2, \ldots, M)$ are revised by considering the effect of both disruptions which is shown as red dashed line in Figure 1. It will be continued same way if there is any other disruption. The model is generalized by formulating for the n^{th} disruption.

In this study, we have made a number of assumptions as follows.

i. Production rate is greater than GMIV of the demand rate.
ii. Single item is produced in the system.
iii. All products are inspected and defective products are rejected.
iv. Total cost of interest and depreciation per production cycle $F(A,r)$ is inversely related to set-up cost (A) and is directly related to the process reliability (r) according to following general power function [13]:

$$F(A,r) = aA^{-b}r^{c}$$

Where a, b and c are positive constants chosen to provide the best fit of the estimated cost function.

2.1 Notations Used in the Study

The following notations have been used in this study.

S_t	Set-up time for a production cycle
δ	Idle time of a production cycle
$\widetilde{D}$	Fuzzy demand per year
H	Holding cost per unit per year ($ per unit per year)
r	Reliability of the production process – which is known from the historical data of the production system
Q	Production lot size per normal cycle with reliability r
A	Set up cost per cycle ($ per set-up)
P	Production rate (units per year) in a 100% reliable system
u	Production downtime for a cycle (set-up time + idle time) $= S_t + \delta$
M	Number of cycles to recovery after the n^{th} disruption – given from the management
l	New disrupted cycle number from previous disruption
$T_{d,n}$	Disruption period in the n^{th} disruption
q_n	Pre-disruption production quantity in the n^{th} disruption
$T_{0,n}$	Production time for $q_n = \frac{q_n}{rP}$
$X_{i,0}$	Production quantity for a normal cycle i
$X_{i,n}$	Production quantity for cycle i the recovery period after n^{th} disruption– which is the decision variable; $i = 1,2,\ldots,M$
$T_{i,0}$	Productions up time for cycle a normal cycle $i = \frac{X_{i,0}}{rP}$
$T_{i,n}$	Productions up time for cycle i in the recovery period after n^{th} disruption
B	Unit back order cost per unit time ($ per unit per unit time)
L	Unit lost sales cost ($ per unit)
C_P	Per unit production cost ($ per unit)
C_R	Rejection cost per unit ($ per unit)
C_I	Inspection cost as a percentage of production cost
m_1	Mark-up for selling price ($m_1 C_P$) – must be greater than 1

3 Model Formulation

In this section, economic production lot size (Q), equations for different costs and revenues and final objective function are derived for the single stage production inventory system that considers process reliability and demand uncertainty. Economic lot size is calculated to minimize the total annual set-up and holding cost. For a single item production system, with lot-for-lot condition under ideal situation [22] with process reliability, the economic production quantity can be formulated as:

$$Q = \sqrt{\frac{2ArP}{H}} \tag{1}$$

3.1 Costs and Revenues Formulation

Holding, set-up, back order, lost sales, production, rejection, inspection and depreciation costs are identified as the relevant costs. Holding cost is determined as unit holding cost multiplied by total inventory during the recovery period which is equivalent to the area under the curve of Figure 1. Set-up cost is calculated as cost pet set-up multiplied by number of set-up in the recovery period. Back order cost is determined as unit back order cost multiplied by back order units and it's time delay [7]. Lost sales cost is determined as unit lost sales cost multiplied by lost sales units [7]. Unit production cost multiplied by total quantity produced during the recovery period is the total production cost. Rejection cost is determined as unit rejection cost multiplied by total rejected quantities [23]. Inspection cost is considered as a certain percentage of the production cost [23]. Cost of interest and depreciation equation is considered as a general power function [13]. The model is generalized by considering the production quantity in cycle i after the n^{th} disruption as $X_{i,n}$ and the original production quantity as $X_{i,0}$.

$$\text{Holding cost } = \frac{1}{2}H\left[\frac{(q_n)^2}{rP} + 2q_n\left(T_{d,n} + S_t\right) + \frac{2X_{1,n}q_n}{rP} + \sum_{i=1}^{M}\frac{(X_{i,n})^2}{rP}\right] \tag{2}$$

$$\text{Set-up cost } = AM \tag{3}$$

$$\text{Production cost } = C_P P\left(\sum_{i=1}^{M} T_{i,n} + T_{0,n}\right) = \frac{C_P}{r}\left(\sum_{i=1}^{M} X_{i,n} + q_n\right) \tag{4}$$

$$\text{Rejection cost} = C_R(1-r)P\left(\sum_{i=1}^{M} T_{i,n} + T_{0,n}\right) = C_R\left(\frac{1}{r} - 1\right)\left(\sum_{i=1}^{M} X_{i,n} + q_n\right) \tag{5}$$

$$\text{Inspection cost} = \frac{C_I C_P}{r}\left(\sum_{i=1}^{M} X_{i,n} + q_n\right) \tag{6}$$

$$\text{Cost of interest and depreciation} = Ma\,(A_1)^{-b}(r)^c \tag{7}$$

$$\text{Back-order cost } = B\left[(X_{1,n} + q_n)\left[T_{d,n} + \frac{q_n}{rP} + \frac{X_{1,n}}{rP} - \frac{X_{l,n-1}}{rP}\right] + \sum_{i=2}^{M} X_{i,n}\cdot\left[T_{d,n} + (i-1)S_t + \frac{q_n}{rP} + \sum_{j=1}^{i}\frac{X_{j,n}}{rP} - \sum_{j=1}^{i}\frac{X_{l+j-1,n-1}}{rP} - (i-1)u\right]\right] \tag{8}$$

$$\text{Lost sales cost } = L\left(\sum_{i=1}^{M} X_{l+i-1,n-1} - \sum_{i=1}^{M} X_{i,n} - q_n\right) \tag{9}$$

Selling price of the acceptable items, which is revenue, in the recovery period is determined as unit selling price multiplied by the demand in the recovery period [23].

$$\text{Revenues} = m_1 C_P\,\widetilde{D}\,\left[\sum_{i=1}^{M}\frac{X_{i,n}}{rP} + \frac{q_n}{rP} + MS_t\right] \tag{10}$$

3.2 Final Objective Function

Total profit, the objective function, is derived by subtracting all costs from the total revenues. Considering all the equations from (2) to (10), the objective function is obtained as follows.

$$\text{Max } \widetilde{Z} = \text{Total Revenues- Total Costs} \tag{11}$$

4 Fuzzy Parameter

In this paper, we consider product demand as a triangular fuzzy number (TFN) to tackle uncertainty. A TFN $\widetilde{D}$ is specified by a triplet (d_1, d_2, d_3) and is defined by its continuous membership function $\mu_{\widetilde{D}}(x): x \rightarrow [0,1]$ as follows:

$$\mu_{\widetilde{D}}(x) = \begin{cases} L(x) = \left(\frac{x-d_1}{d_2-d_1}\right) & if\ d_1 \leq x \leq d_2 \\ R(x) = \left(\frac{d_3-x}{d_3-d_2}\right) & if\ d_2 \leq x \leq d_3 \\ 0 & otherwise \end{cases} \tag{12}$$

$L(x)$ and $R(x)$ indicates the left and right branch of the TFN $\widetilde{D}$ respectively. An α-cut of $\widetilde{D}$ can be expressed by the following interval [17]:

$$D(\alpha) = [d_1 + (d_2 - d_1)\alpha,\ d_3 - (d_3 - d_2)\alpha],\ \ \alpha \in [0,1]$$

The graded mean integration value (GMIV) of a LR-fuzzy number is introduced by Chen and Hsieh [24]. The graded mean integration representation method is based on the integral value of the graded mean α-level of the LR-fuzzy number for defuzzifing LR-fuzzy numbers. By considering $\widetilde{D}$ is a LR-fuzzy number and according to Chen and Hsieh [24], the GMIV of $\widetilde{D}$ is defined as:

$$\mathrm{G}(\widetilde{\mathrm{D}}) = \frac{\int_0^1 \left(\frac{\alpha}{2}\right)\{\mathrm{L}^{-1}(\alpha)+\mathrm{R}^{-1}(\alpha)\}d\alpha}{\int_0^1 \alpha d\alpha} = \int_0^1 \alpha\{\mathrm{L}^{-1}(\alpha) + \mathrm{R}^{-1}(\alpha)\}\, d\alpha \tag{13}$$

5 Disruption Recovery Model with Fuzzy Demand

In this section, fuzziness of demand is incorporated to the final mathematical model. The GMIV of the expected total profit function is evaluated. Relevant constraints are also developed with the GMIV of expected fuzzy demand. After simplifying the equation (11), the following equation of the total profit is obtained:

$$\widetilde{Z} = C\,\widetilde{D} + Y \tag{14}$$

Now, considering the fuzzy random demand $\widetilde{\mathrm{D}}$ with the given set of ta $(\tilde{d}_1, \tilde{p}_1), (\tilde{d}_2, \tilde{p}_2), (\tilde{d}_3, \tilde{p}_3), \ldots\ldots, (\tilde{d}_v, \tilde{p}_v)$, the profit $(\widetilde{Z})$ is also a fuzzy random variable and its expectation is a unique fuzzy number [14] which is,

$$E\widetilde{Z} = C\sum_{k=1}^{v} \tilde{d}_k \tilde{p}_k + Y$$

In this paper, demand data are considered as a triangular fuzzy number (TFN). Demand TFN and associated probabilities are taken as a triplet $\left(\underline{d_k}, d_{k,} \overline{d_k}\right)$ and

$\left(\underline{p_k}, p_k, \overline{p_k}\right)$ respectively. Where, k= *1, 2, 3....*, *v*. Then the fuzzy expected profit function will also be a TFN, $E\widetilde{Z} = \left(\underline{\mathrm{EZ}}, \mathrm{EZ}, \overline{\mathrm{EZ}}\right)$ which is determined as follows:

$$EZ = E[Z(\alpha = 1)] = C\textstyle\sum_{k=1}^{v} d_k\, p_k + Y$$
$$\underline{EZ} = E[Z_L(\alpha = 0)\,] = C\textstyle\sum_{k=1}^{v} \underline{d_k}\;\underline{p_k} + Y$$
$$\overline{EZ} = E[Z_R(\alpha = 0)] = C\textstyle\sum_{k=1}^{v} \overline{d_k}\;\overline{p_k} + Y$$

Here the α-level set of the fuzzy number $E\widetilde{Z}$ are considered as $EZ(\alpha) = E[Z(\alpha)] = \left[E\left(Z_L(\alpha)\right), E\left(Z_R(\alpha)\right)\right]$; $0 \le \alpha \le 1$ and different α -cut intervals for the fuzzy number $E\widetilde{Z}$ are obtained for different α between 0 and 1.Taking, α-cut on both sides of equation of $E\widetilde{Z}$.

$$E\tilde{Z}_\alpha = C\textstyle\sum_{k=1}^{v} \tilde{d}_{k\alpha}\tilde{p}_{k\alpha} + Y$$

The arithmetic interval of fuzzy demand and associated probabilities using an α-cut is determined as follows.

$$\tilde{d}_{k\alpha} = \left[\underline{d_k} + \alpha\left(d_k - \underline{d_k}\right),\ \overline{d_k} - \alpha\left(\overline{d_k} - d_k\right)\right]$$
$$\tilde{p}_{k\alpha} = \left[\underline{p_k} + \alpha\left(p_k - \underline{p_k}\right),\ \overline{p_k} - \alpha(\overline{p_k} - p_k)\right]$$

By using these arithmetic intervals, $E\tilde{Z}_\alpha$ is evaluated as:

$$E\tilde{Z}_\alpha = \left[\left[C\sum_{k=1}^{v}\left[\underline{d_k} + \alpha\left(d_k - \underline{d_k}\right)\right]\left[\underline{p_k} + \alpha\left(p_k - \underline{p_k}\right)\right] + Y\right],\right.$$
$$\left.\left[C\sum_{k=1}^{v}\left[\overline{d_k} - \alpha\left(\overline{d_k} - d_k\right)\right]\left[\overline{p_k} - \alpha(\overline{p_k} - p_k)\right] + Y\right]\right]$$

From the representation of graded mean integration methods based on the integral value of the graded mean α-level of the LR-fuzzy number of the total profit, $\mathrm{L}^{-1}(\alpha)$ and $\mathrm{R}^{-1}(\alpha)$ are obtained as follows.

$$\mathrm{L}^{-1}(\alpha) = C\textstyle\sum_{k=1}^{v}\left[\underline{d_k} + \alpha\left(d_k - \underline{d_k}\right)\right]\left[\underline{p_k} + \alpha\left(p_k - \underline{p_k}\right)\right] + Y$$
$$\mathrm{R}^{-1}(\alpha) = C\textstyle\sum_{k=1}^{v}\left[\overline{d_k} - \alpha\left(\overline{d_k} - d_k\right)\right]\left[\overline{p_k} - \alpha(\overline{p_k} - p_k)\right] + Y$$

The unique fuzzy number $\mathrm{G}\left(E\widetilde{Z}\,\right)$ is determined by substituting the value of $\mathrm{L}^{-1}(\alpha)$ and $\mathrm{R}^{-1}(\alpha)$ to the equation (13),

$$G\left(E\widetilde{Z}\,\right) = \int_0^1\left[\alpha\left[C\sum_{k=1}^{v}\left[\underline{d_k} + \alpha\left(d_k - \underline{d_k}\right)\right]\left[\underline{p_k} + \alpha\left(p_k - \underline{p_k}\right)\right] + Y\right]\right]d\alpha$$
$$+ \int_0^1\left[\alpha\left[C\sum_{k=1}^{v}\left[\overline{d_k} - \alpha\left(\overline{d_k} - d_k\right)\right]\left[\overline{p_k} - \alpha(\overline{p_k} - p_k)\right] + Y\right]\right]d\alpha$$

After integrating and simplifying the above equation of $G\left(E\widetilde{Z}\right)$, the GMIV of the total profit function, which is to be maximized, and obtained as:

$$\text{Max } \mathrm{G}\left(E\widetilde{Z}\right) = C(Z_1 + Z_2) + Y \tag{15}$$

Where,

$$Z_1 = \sum_{k=1}^{v} \{\frac{1}{2}\underline{d_k}\,\underline{p_k} + \frac{1}{3}\underline{d_k}\left(p_k - \underline{p_k}\right) + \frac{1}{3}\underline{p_k}\left(d_k - \underline{d_k}\right) + \frac{1}{4}\left(d_k - \underline{d_k}\right)\left(p_k - \underline{p_k}\right)\}$$

$$Z_2 = \sum_{k=1}^{v} \{\frac{1}{2}\overline{d_k}\overline{p_k} - \frac{1}{3}\overline{d_k}(\overline{p_k} - p_k) - \frac{1}{3}\overline{p_k}\left(\overline{d_k} - d_k\right) + \frac{1}{4}\left(\overline{d_k} - d_k\right)(\overline{p_k} - p_k)\}$$

GMIV of the expected fuzzy demand, $\mathrm{G}\left(E\widetilde{D}\right) = Z_1 + Z_2$

Subject to the following constraints:

$$X_{i,0} = Q \tag{16}$$

$$X_{1,n} + q_n \le X_{l,n-1} \tag{17}$$

$$X_{i,n} \le X_{l+i-1,n-1}\text{ ; } i = 2,3,4,\ldots\ldots,M \tag{18}$$

$$rP \ge G\left(E\widetilde{D}\right) \tag{19}$$

$$r \le 1 \tag{20}$$

$$\textstyle\sum_{i=1}^{M} X_{i,n} + q_n \le rP\left(\sum_{i=1}^{M}\frac{X_{l+i-1,n-1}}{G(E\widetilde{D})} - MS_t - T_{d,n}\right) \tag{21}$$

$$\textstyle\sum_{i=1}^{M} X_{i,n} + q_n \ge \left(\frac{\sum_{i=1}^{M} X_{i,n}+q_n}{rP} + MS_t\right)G(E\widetilde{D}) - \left(\sum_{i=1}^{M} X_{l+i-1,n-1} - \sum_{i=1}^{M} X_{i,n} - q_n\right) \tag{22}$$

$$\frac{X_{1,n}+q_n}{G(E\widetilde{D})} - \frac{X_{2,n}}{rP} - S_t \ge 0 \tag{23}$$

$$\frac{X_{i,n}}{G(E\widetilde{D})} - \frac{X_{i+1,n}}{rP} - S_t \ge 0;\ i\ =\ 2,3,\ldots.,M \tag{24}$$

$$T_{d,n} + \frac{q_n}{rP} + \frac{X_{1,n}}{rP} - \frac{X_{l,n-1}}{rP} \ge 0 \tag{25}$$

$$T_{d,n} + (i-1)S_t + \frac{q_n}{rP} + \textstyle\sum_{j=1}^{i}\frac{X_{j,n}}{rP} - \sum_{j=1}^{i}\frac{X_{l+j-1,n-1}}{rP} - (i-1)u \ge 0;\ i = 2,3,4,\ldots\ldots,M \tag{26}$$

6 Solution Approach

We propose a genetic algorithm based heuristic to solve the model. Genetic algorithm is very popular technique to solve complex non-linear constrained optimization problem. GAs are general purpose optimization algorithms which apply the rules of natural genetics to explore a given search space [25]. The heuristic is designed to make a recovery plan from a single or a series of production disruptions. The proposed heuristic revises the production lot size of each cycle as long as disruptions take place in the system. For a series of disruptions, the heuristic revises the lot size of each cycle by considering the effect of all previous dependent disruptions. The proposed genetic algorithm based heuristic is presented in the Figure 2. The above mentioned heuristic is coded in MATLAB R2012a with the help of its optimization toolbox. In the proposed heuristic, following GA parameters are used to solve the model.

Population size: 100; Population type: Double vector; Crossover fraction: 0.8; Maximum number of generations: 3000; Function tolerance: 1e-6; Non linear constraint tolerance: 1e-6 and other parameters are set as default of the optimization toolbox.

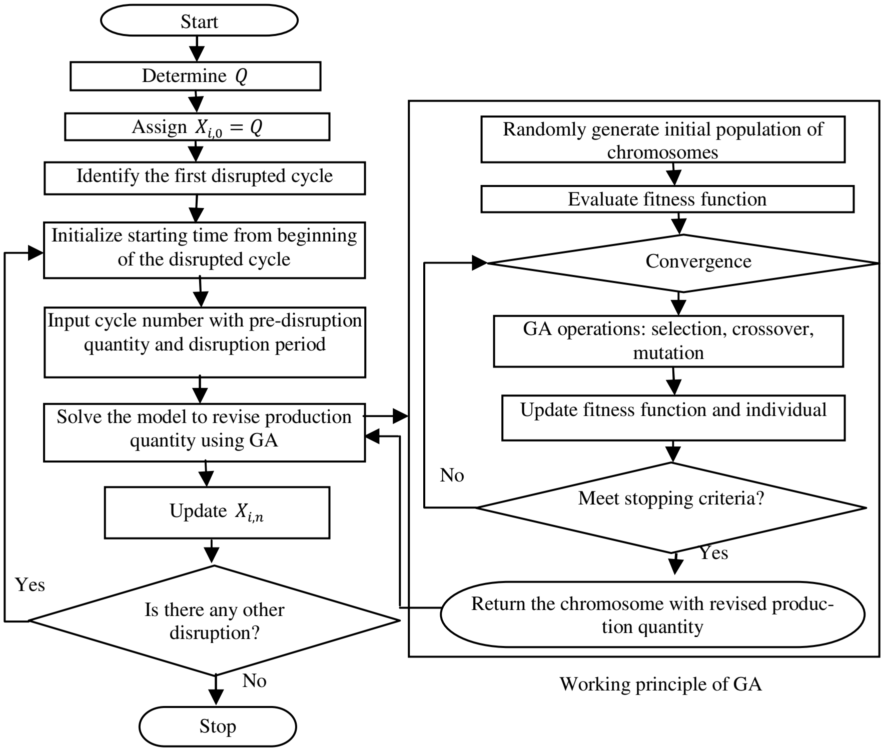

Start
Determine Q
Assign $X_{i,0} = Q$
Identify the first disrupted cycle
Initialize starting time from beginning of the disrupted cycle
Input cycle number with pre-disruption quantity and disruption period
Solve the model to revise production quantity using GA
Update $X_{i,n}$
Is there any other disruption?
Yes
No
Stop
Randomly generate initial population of chromosomes
Evaluate fitness function
Convergence
GA operations: selection, crossover, mutation
Update fitness function and individual
No
Meet stopping criteria?
Yes
Return the chromosome with revised production quantity
Working principle of GA

Fig. 2. Flowchart of proposed GA based heuristic

7 Results Analysis

Results have been analysed for both single and multiple disruptions on a real time basis. For single disruption, there is only one random disruption in the system and there is no more disruption within the recovery period. For multiple disruptions, there is a series of disruptions, one after another, on a real time basis in the system.

7.1 Results Analysis for Single Disruption

Following data are considered to analyze the results for single disruption:
$S_t = 0.000057, u = 0.000077,\ A = 60,\ H = 1.4,\ r = 0.92,\ P = 550000,\ M = 5,$ $B = 20,\ L = 25,\ T_{d1} = 0.01,\ q_1 = 500, C_P = 40,\ C_R = 15,\ C_I = 0.01, a = 1000,$ $b = 0.5, c = 0.75, m_1 = 2.5$, and demand data are considered as TFN which is shown in the Table 1.

Table 1. Demand data as TFN and associated probabilities

Demand rate	Probability
(300000, 320000, 340000)	(0.050, 0.055, 0.060)
(350000, 370000, 390000)	(0.144, 0.150, 0.156)
(400000, 420000, 440000)	(0.293, 0.300, 0.307)
(450000, 470000, 490000)	(0.194, 0.202, 0.210)
(500000, 520000, 540000)	(0.104, 0.110, 0.117)
(550000, 570000, 590000)	(0.094, 0.100, 0.106)
(600000, 620000, 640000)	(0.088, 0.093, 0.098)

The problem is solved using the proposed GA based heuristic. The results are obtained from 30 different runs. The best recovery plan after single disruption is shown in Table 2. The production system returns to original schedule from the sixth cycle after the disruption with $X_{6,1} = 6586$, $X_{7,1} = 6586$ and so on. The maximum total profit in the recovery period is obtained as 1381112.5.

Table 2. Best results obtained for the single disruption

Disruption number (n)	Revised production quantity					Total profit
	$X_{1,1}$	$X_{2,1}$	$X_{3,1}$	$X_{4,1}$	$X_{5,1}$	
1	5305	5638	6066	6469	6563	1381112.5

7.2 Results Analysis for a Series of Disruptions

In this case, a series of disruptions on a real time basis is considered, which is shown in Table 3. In this series of disruptions, seven dependent disruptions are considered and each one occurs within the recovery period of the previous disruption. Other data remain same as in section 7.1.

Table 3. Data for the series of disruptions

Disruption number (n)	Disrupted cycle number from previous disruption	Pre-disruption quantity	Disruption period
1	1	1000	0.0045
2	2	650	0.0092
3	3	500	0.0025
4	5	1500	0.0065
5	2	0	0.0110
6	4	800	0.0098
7	3	0	0.0078

The production system with multiple disruptions is also solved using the GA based heuristic on a real time basis. The results are obtained from 30 different runs. Production quantity in each cycle is revised after each disruption considering the effect of entire dependent disruptions to maximize the total profit in the recovery period. The best recovery plan obtained from the heuristic for the series of disruptions is shown in Table 4. The production system returns to original schedule from the sixth cycle after each disruption with $X_{6,n} = 6586,\ X_{7,n} = 6586$ and so on.

Table 4. Best results obtained for the series of disruptions

Disruption number (n)	Revised production quantity					Total profit
	$X_{1,n}$	$X_{2,n}$	$X_{3,n}$	$X_{4,n}$	$X_{5,n}$	
1	5585	6561	6545	6567	6563	1545915.7
2	5272	5671	6111	6553	6572	1404616.5
3	5611	6548	6570	6541	6550	1524257.8
4	4804	6539	6433	6476	6522	1506722.6
5	5271	5647	5994	6336	6381	1324912.7
6	4941	5668	5936	6271	6532	1364154.2
7	5657	5914	6049	6486	6443	1407321.9

8 Conclusions

The objective of this research was to incorporate demand uncertainty and process reliability in developing a disruption recovery model for managing risk in a production inventory system. A single or a series of disruptions on a real time basis was considered to make the model applicable in practical problems. The model was formulated as a non-linear constrained optimization problem and generalized by formulating the model for the n^{th} disruption. A genetic algorithm based heuristic was proposed to solve the model with single or multiple disruptions on a real time basis. This model can be applied in an imperfect production process where the process countenances a single or multiple production disruptions and product demand is uncertain. The model can be extended by considering multiple stages in the production system.

References

1. Graves, S.C.: A single-item inventory model for a nonstationary demand process. Manufacturing & Service Operations Management 1(1), 50–61 (1999)
2. Dave, U.: A deterministic lot - size inventory model with shortages and a linear trend in demand. Naval Research Logistics 36(4), 507–514 (2006)
3. Chan, G.H., Song, Y.: A dynamic analysis of the single-item periodic stochastic inventory system with order capacity. European Journal of Operational Research 146(3), 529–542 (2003)
4. Kiesmüller, G.P., De Kok, A.G., Dabia, S.: Single item inventory control under periodic review and a minimum order quantity. International Journal of Production Economics 133(1), 280–285 (2011)
5. Lin, G.C., Gong, D.C.: On a production-inventory system of deteriorating items subject to random machine breakdowns with a fixed repair time. Mathematical and Computer Modelling 43(7), 920–932 (2006)

6. Widyadana, G.A., Wee, H.M.: Optimal deteriorating items production inventory models with random machine breakdown and stochastic repair time. Applied Mathematical Modelling 35(7), 3495–3508 (2011)
7. Hishamuddin, H., Sarker, R.A., Essam, D.: A disruption recovery model for a single stage production-inventory system. European Journal of Operational Research 222(3), 464–473 (2012)
8. Hishamuddin, H., Sarker, R.A., Essam, D.: A Recovery Model for a Two–Echelon Serial Supply Chain with Consideration of Transportation Disruption. Computers & Industrial Engineering 64(2), 552–561 (2013)
9. Parlar, M., Perry, D.: Inventory models of future supply uncertainty with single and multiple suppliers. Naval Research Logistics 43(2), 191–210 (1998)
10. Özekici, S., Parlar, M.: Inventory models with unreliable suppliers in a random environment. Annals of Operations Research 91, 123–136 (1999)
11. Mohebbi, E., Hao, D.: An inventory model with non-resuming randomly interruptible lead time. International Journal of Production Economics 114(2), 755–768 (2008)
12. Qi, L., Shen, Z.J.M., Snyder, L.V.: A continuous ‐ review inventory model with disruptions at both supplier and retailer. Production and Operations Management 18(5), 516–532 (2009)
13. Cheng, T.C.E.: An economic production quantity model with flexibility and reliability considerations. European Journal of Operational Research 39(2), 174–179 (1989)
14. Bag, S., Chakraborty, D., Roy, A.R.: A production inventory model with fuzzy random demand and with flexibility and reliability considerations. Computers & Industrial Engineering 56(1), 411–416 (2009)
15. Sana, S.S.: A production–inventory model in an imperfect production process. European Journal of Operational Research 200(2), 451–464 (2010)
16. Sarkar, B.: An inventory model with reliability in an imperfect production process. Applied Mathematics and Computation 218(9), 4881–4891 (2012)
17. Lee, H.M., Yao, J.S.: Economic production quantity for fuzzy demand quantity, and fuzzy production quantity. European Journal of Operational Research 109(1), 203–211 (1998)
18. Chang, S.C.: Fuzzy production inventory for fuzzy product quantity with triangular fuzzy number. Fuzzy Sets and Systems 107(1), 37–57 (1999)
19. Yao, J.S., Chang, S.C., Su, J.S.: Fuzzy inventory without backorder for fuzzy order quantity and fuzzy total demand quantity. Computers & Operations Research 27(10), 935–962 (2000)
20. Dutta, P., Chakraborty, D., Roy, A.R.: A single-period inventory model with fuzzy random variable demand. Mathematical and Computer Modelling 41(8), 915–922 (2005)
21. Islam, S., Roy, T.K.: Fuzzy multi-item economic production quantity model under space constraint: A geometric programming approach. Applied Mathematics and Computation 184(2), 326–335 (2007)
22. Sarker, R.A., Khan, L.R.: An optimal batch size for a production system operating under periodic delivery policy. Computers & Industrial Engineering 37(4), 711–730 (1999)
23. Paul, S.K., Azeem, A., Sarker, R., Essam, D.: Development of a production inventory model with uncertainty and reliability considerations. Optimization & Engineering, Accepted manuscript (2013), doi:10.1007/s11081-013-9218-6
24. Chen, S.H., Hsieh, C.H.: Graded mean integration representation of generalized fuzzy number. Journal of Chinese Fuzzy Systems 5(2), 1–7 (1999)
25. Homaifar, A., Qi, C.X., Lai, S.H.: Constrained optimization via genetic algorithms. Simulation 62(4), 242–253 (1994)

Author Index