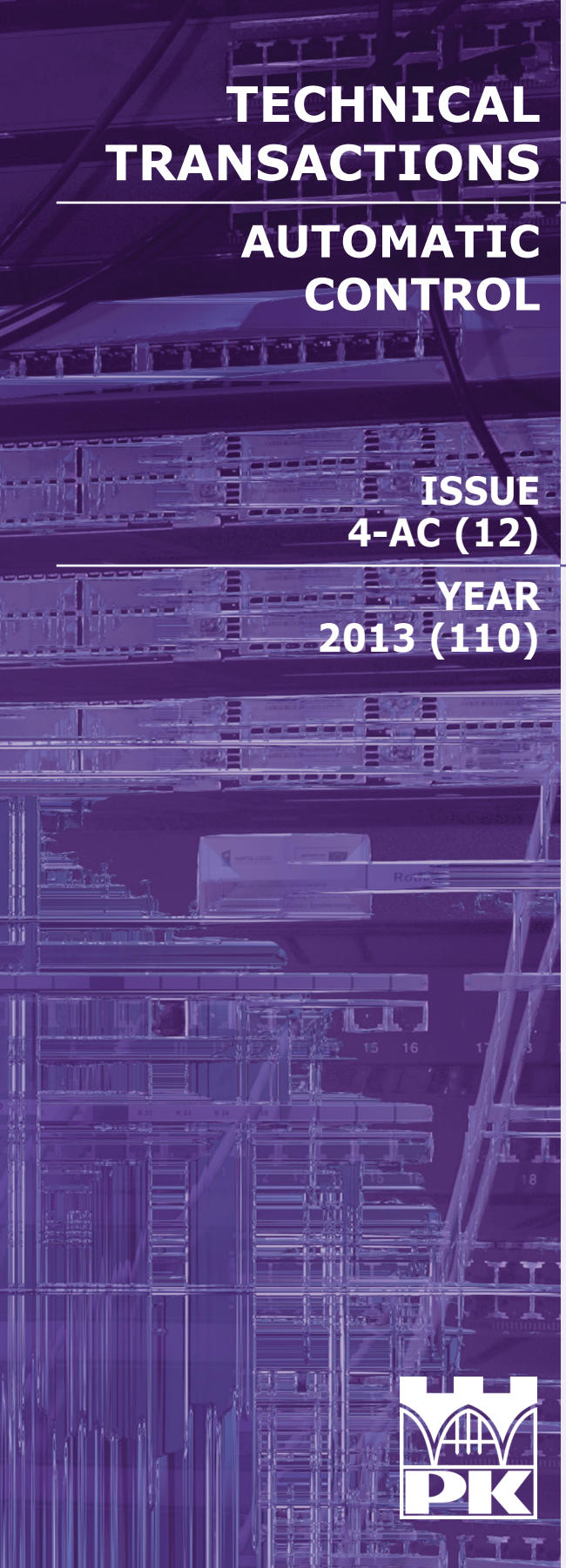

TECHNICAL TRANSACTIONS

CZASOPISMO TECHNICZNE

AUTOMATIC CONTROL

AUTOMATYKA

ISSUE 4-AC (12)

ZESZYT 4-AC (12)

YEAR 2013 (110)

ROK 2013 (110)

WYDAWNICTWO POLITECHNIKI KRAKOWSKIEJ

TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE

AUTOMATIC CONTROL | AUTOMATYKA

ISSUE 4-AC (12) | ZESZYT 4-AC (12)
YEAR 2013 (110) | ROK 2013 (110)

Pierwotną wersją każdego Czasopisma Technicznego jest wersja on-line
www.czasopismotechniczne.pl www.technicaltransactions.com

Technical Transactions
series *Automatic Control*
vol. 4-AC/2013

Advances in Information Technology and Systems Research

Eds. Piotr A. Kowalski, Szymon Łukasik, Piotr Kulczycki

Editorial Board

Editor-in-Chief of the *Automatic Control* series
Prof. Piotr Kulczycki, D.Sc., Ph.D., M.Sc.EE, M.Sc.Math.

A NOTE FROM THE EDITOR-IN-CHIEF

Dear Readers,

This issue sees the end of the second year of the series *Automatic Control*, appearing as part of the *Technical Transactions* journal, published by the *Cracow University of Technology Press*. In the first year, 2012 – to be more specific, in the three months following the summer vacation – the pilot issue left the presses, and in the second, 2013, the series became a fully-fledged quarterly. Now the series *Automatic Control* of the *Technical Transactions* journal has an entry on the portal *Scientific Journals Online* (http://www.ejournals.eu).

Together with the current development trends in automatic control, the subjects covered in issues of the series *Automatic Control* to date have spanned a broad and varied range: from new aspects of classical control engineering and mathematical modeling to modern information technology. Specific articles gave extensive representation to temporary aspects of data analysis and exploration, artificial intelligence, information processing and decision-making support.

In bringing this initial phase to a close, I would like to extend my warmest gratitude to all members of the first **Editorial Board**. Its names, listed on the second page of this issue, would undoubtedly bring esteem to any journal.

My thanks also go to the **Executive Editors** for their assistance in dealing with a magnitude of daily duties.

Aware of the particularly thankless task that is reviewing, I especially appreciate the efforts of the **reviewers**:

Krassimir Atanassov (Bulgarian Academy of Sciences, BULGARIA),
Valentina Emilia Bălaş (Aurel Vlaicu University of Arad, ROMANIA),
Ewa Bednarczuk (Systems Research Institute, Polish Academy of Sciences, POLAND),
Robert Cierniak (Czestochowa University of Technology, POLAND),
Antoni Leon Dawidowicz (Jagiellonian University, POLAND),
Guy De Tré (Ghent University, BELGIUM),
Jan Tadeusz Duda (AGH University of Science and Technology, POLAND),
Ewa Dudek-Dyduch (AGH University of Science and Technology, POLAND),
Uliana Dzelendziak (Lviv Polytechnic National University, UKRAINE),
Abdelali El Aroudi (Universitat Rovira i Virgili, SPAIN),
Alexander Gegov (University of Portsmouth, GREAT BRITAIN),
Olgierd Hryniewicz (Systems Research Institute, Polish Academy of Sciences, POLAND),
Andrzej Janczak (University of Zielona Góra, POLAND),
Jerzy Józefczyk (Wrocław University of Technology, POLAND),
Jacek Kabziński (Lodz University of Technology, POLAND),
Janusz Kacprzyk (Systems Research Institute, Polish Academy of Sciences, POLAND),
Jerzy Klamka (Silesian University of Technology, POLAND),
László T. Kóczy (Széchenyi István University in Györ, HUNGARY),
Heikki Koivo (Aalto University, FINLAND),
Józef Korbicz (University of Zielona Góra, POLAND),
Zdzisław Kowalczuk (Gdańsk University of Technology, POLAND),
Wiesław Krajewski (Systems Research Institute, Polish Academy of Sciences, POLAND),
Lech Kruś (Systems Research Institute, Polish Academy of Sciences, POLAND),
Halina Kwaśnicka (Wrocław University of Technology, POLAND),
Rita Lovassy (Óbuda University, HUNGARY),
Piotr Malecki (Institute of Nuclear Physics, Polish Academy of Sciences, POLAND),
Jacek Malinowski (Systems Research Institute, Polish Academy of Sciences, POLAND),
Krzysztof Malinowski (Warsaw University of Technology, POLAND),
Radko Mesiar (Slovak University of Technology, SLOVAKIA),
Andrzej Myśliński (Systems Research Institute, Polish Academy of Sciences, POLAND),
Andrian Nakonechnyy (Lviv Polytechnic National University, UKRAINE),
Andrzej Obuchowicz (University of Zielona Góra, POLAND),
Krzysztof Patan (University of Zielona Góra, POLAND),
Irena Pawłow-Niezgódka (Systems Research Institute, Polish Academy of Sciences, POLAND),
Milan Petrík (Institute of Computer Science, Academy of Sciences of the Czech Republic, CZECH REPUBLIC),
Andrzej Pieczyński (University of Zielona Góra, POLAND),
Oleksandr Rolick (National Technical University of Ukraine, UKRAINE),

Leszek Rutkowski (Czestochowa University of Technology, POLAND),
Dominik Sankowski (Lodz University of Technology, POLAND),
Modesto Castrillón Santana (University of Las Palmas de Gran Canaria, SPAIN),
Henrik Schiøler (Aalborg University, DENMARK),
Jan Studziński (Systems Research Institute, Polish Academy of Sciences, POLAND),
Krzysztof Szkatuła (Systems Research Institute, Polish Academy of Sciences, POLAND),
Eulalia Szmidt (Systems Research Institute, Polish Academy of Sciences, POLAND),
Ryszard Tadeusiewicz (AGH University of Science and Technology, POLAND),
Sergii Telenyk (National Technical University of Ukraine, UKRAINE),
Lipo Wang (Nanyang Technological University, SINGAPORE),
Rafael Wisniewski (Aalborg University, DENMARK),
Marcin Witczak (University of Zielona Góra, POLAND),
Sławomir Zadrożny (Systems Research Institute, Polish Academy of Sciences, POLAND).

Their subject input and criticism (the rate of refused texts reached 35%) was a particularly valuable help in this early period, always difficult by nature.

I am grateful to everyone, especially the authors who sent manuscripts, at the end of such a fruitful term, for their cooperation and for what we have created together.

Piotr Kulczycki

TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE
AUTOMATIC CONTROL | AUTOMATYKA
4-AC/2013

SŁAWOMIR ZADROŻNY*, JANUSZ KACPRZYK**, MAREK GAJEWSKI***, MACIEJ WYSOCKI****

A NOVEL TEXT CLASSIFICATION PROBLEM AND ITS SOLUTION

O PEWNYM ZADANIU KLASYFIKACJI DOKUMENTÓW TEKSTOWYCH I JEGO ROZWIĄZANIACH

Abstract

A new text categorization problem is introduced. As in the classical problem, there is a set of documents and a set of categories. However, in addition to being assigned to a specific category, each document belongs to a certain sequence of documents, referred to as a *case*. It is assumed that all documents in the same case belong to the same category. An example may be a set of news articles. Their categories may be sport, politics, entertainment, etc. In each category there exist cases, i.e., sequences of documents describing, for example evolution of some events. The problem considered is how to classify a document to a proper category and a proper case within this category. In the paper we formalize the problem and discuss two approaches to its solution.

Keywords: *text categorization, sequences of documents, sequence mining, hidden Markov models*

Streszczenie

W artykule proponuje się nowe zadanie kategoryzacji dokumentów tekstowych. Podobnie jak w zadaniu klasycznym rozważa się zbiór dokumentów tekstowych i zbiór kategorii. W odróżnieniu od zadania klasycznego, dokumenty są przypisane nie tylko do kategorii, ale również do określonej sekwencji dokumentów w ramach danej kategorii, zwanej *sprawą*. Zakłada się, że wszystkie dokumenty danej sprawy należą do tej samej kategorii. Przykładem może być kolekcja wiadomości prasowych. Mogą one należeć do kategorii takich, jak sport, polityka, rozrywka itp. W ramach każdej kategorii występują sekwencje wiadomości (sprawy) opisujące np. rozwój pewnych zdarzeń. Zadanie polega więc na zaklasyfikowaniu dokumentu do właściwej kategorii i właściwej sprawy w jej ramach. W artykule formalnie definiuje się nowe zadanie kategoryzacji i proponuje się dwa podejścia do jego rozwiązania.

Słowa kluczowe: *kategoryzacja dokumentów tekstowych, sekwencje dokumentów, odkrywanie wzorców sekwencji, ukryte modele Markowa*

* D.Sc. Ph.D. Sławomir Zadrożny, e-mail: slawomir.zadrozny@ibspan.waw.pl, Systems Research Institute, Polish Academy of Sciences, Warsaw; Warsaw School of Information Technology.

** Prof. D.Sc. Ph.D. Janusz Kacprzyk, Systems Research Institute, Polish Academy of Sciences, Warsaw; Department of Automatic Control and Information Technology, Faculty of Electrical and Computer Engineering, Cracow University of Technology.

*** M.Sc. Marek Gajewski, Doctoral Studies, Systems Research Institute, Polish Academy of Sciences, Warsaw.

**** M.Sc. Maciej Wysocki, Warsaw School of Information Technology.

1. Introduction

Textual documents have always been an important component of broadly meant business processes. Nowadays, more and more of these documents are available in the electronic form. This calls for effective methods of their automatic processing. The relevant research problems are dealt with in the framework of information retrieval (IR). The basic task addressed concerns finding documents satisfying the user's information needs. This requires a proper representation of the documents, information needs (usually in the form of a query) as well as appropriate techniques to match them to each other. Another crucial task is the automatic classification of documents to a set of predefined categories, usually referred to as the text categorization [10, 11]. The categories of interest are often of a topical character, i.e., two documents belong to the same category if they are thematically related. For example, news may be classified in such a way for the purposes of an editorial office of a journal, documents served by a website may be grouped based on their main topics, etc. However, other origins of the categories may be also found in practical applications of the text categorization paradigm. For example, poems may be classified according to their authors, documents received by a company may be grouped according to their type (ads, analyses, reports etc.), or according to their language etc.

Some variants of the categorization task may be distinguished, in a similar way as in the general case of classification. Namely, the binary case where there are just two categories (e.g., of relevant and non-relevant documents with respect to user's preferences, in the *information filtering* task) is of a special interest due to the fact that many well-known classification algorithms are originally meant for the case of two classes. On the other hand, in practical settings the *multiclass* case is much more typical (e.g., item news may be assigned to the politics, economy, sport, etc. category). An important parameter of a text categorization task is the number of categories to be assigned to a document. It may be limited to *at most one*, *exactly one*, or may be unlimited, i.e., a few categories may be assigned to the same document. The latter case is referred to as *multilabel* categorization. Another distinction can be made concerning the mode in which documents are classified, i.e., if they are classified individually, one-by-one (*on-line categorization*), or in groups (*batch categorization*). In the latter mode, the system has more information available while classifying documents but the former mode may be dictated by a practical application at hand.

The text categorization problem, as sketched above, is thus an example of the general classification task. It is most often dealt with in the supervised learning mode. Hence, a training data set is assumed to exist which gathers examples of documents with known class assignment. It is used to construct a classifier which is then used to classify new documents, 'unseen' in the training data set. If the *vector space model* (cf., e.g., [9]) is assumed then documents are represented as numerical vectors making it possible to employ any of the multitude of classifier construction techniques. On an abstract level, these classifier construction techniques may be seen as discovering some regularities characterizing documents belonging to particular categories in the training data set.

In this paper we introduce a new problem of textual documents classification which is an extension of the standard text categorization problem. The problem is inspired by a real life task of the acquisition, maintenance and handling of documents in commercial companies and institutions, with an emphasis on public institutions in Poland. These institutions are

obliged to organize their documents in a precisely specified way. There is a hierarchy (a tree) of topics, so-called JRWA, which comprises some high level topics defined by the appropriate acts of law and low level topics which are adjusted to the specificity of a given institution. Every business process carried out by an institution is assigned to a node of the JRWA tree and all documents related to a given instance of such a process form a *case* (also referred to as a *story*). For example, there may be a JRWA tree node corresponding to the tenders for the equipment purchase. A case is then a particular tender and related documents may comprise a tender announcement, submitted offers, protocols of the tender commission meetings, etc. The documents within a case are chronologically ordered according to the date a document has been created or received.

The classification problem considered in this paper may thus be briefly described as follows. Let us assume a JRWA tree with a number of cases assigned to its nodes. Usually these cases will be at a different stage of development. For example, one tender may have been just announced and its list of related documents consists of only the announcement, while other tenders may be close to their closing. The same applies to the cases gathered in other nodes of the JRWA tree. The problem which we face is a proper classification of a document which has been just received or produced. It has to be classified both to an appropriate JRWA node and to a specific case within this node. The former classification is somehow easier and close to the standard text categorization problem. The latter is much more complex and requires taking into account the relationship between subsequent documents in the chronological order of particular cases. Both classification problems are intermingled as, first of all, assigning a document to a case implies its assignment to the JRWA node to which this case belongs. On the other hand, such a direct classification to a case may be difficult and preceding it by first classifying a document to a JRWA node may be advantageous.

In the following sections we formally state a new text categorization problem and briefly discuss its possible solutions.

2. Formal statement of the new text categorization problem

We assume a collection of documents arranged in ordered sequences, referred to earlier as cases, which are assigned to nodes of a hierarchy of categories. We will adopt the following notation:

- $D = \{d_1, ..., d_n\}$ is a set of documents,
- $C = \{c_1, ..., c_m\}$ is a set of categories of documents, arranged in a hierarchy (tree),
- $\sigma_k = \langle d_{k1}, ..., d_{kl} \rangle$ is a *sequence of documents* (*case*) of documents,
- $\Sigma = \{\sigma_1, ..., \sigma_p\}$ is a set of cases; all documents of a case belong to the same category or, equivalently, each case is assigned to a category; all cases are pairwise disjoint.

Thus, there are two orthogonal classification schemes in place here. We assume that documents in the same category are somehow similar thematically while documents belonging to the same case form some logical sequence. For example, let us consider a set of news articles. The categories here may be *sport*, *politics*, *entertainment,* etc. In each category there exist sequences of documents (cases) describing, for example evolution of some events. For instance, in the *sport* category there may be a case for the Olympic games, another one for football world championships, etc. The documents (articles) in the former

case may report results of consecutive matches/competitions while in the latter subsequent articles, may discuss stages of the preparations for the championship. Of course, these are just examples and a different hierarchy of topics and interpretation of cases may be assumed implying a different arrangement of the articles.

Let us now consider a new document d^*. By definition it belongs to a case in some category. Our goal is to devise a way of constructing a classifier which will suggest the assignment of d^* to a proper case.

In the next sections, we propose two ways to construct such a classifier. Both approaches follow the standard framework of the supervised learning. A collection of documents D, arranged in a number of cases, Σ, is assumed to be available which is split into a training and testing part. The testing part has to be further split as for testing we need:

– a set of cases at various stages of evolvement, forming a structure in context of which the documents have to be classified,
– a set of individual documents which have to be classified with respect to the above mentioned testing structure.

Let us denote the original testing part of the collection of documents as $\Sigma_T = \{\sigma_1, ..., \sigma_p\}$. The testing structure (set of cases), $\Sigma' = \{\sigma'_1, ..., \sigma'_w\}$ is constructed in such a way that $w \leq p$ and if $\sigma'_k \in \Sigma'$, $\sigma'_k = \langle d_{k1}, ..., d_{km} \rangle$ then there have to exist $\sigma_k \in \Sigma_T$, $\sigma_k = \langle d_{k1}, ..., d_{kl} \rangle$, where $m \leq l$. Thus, in general, not all cases of Σ_T appear in the testing structure Σ', and those which do appear are, in general, subsequences of a certain number of initial documents of the original cases from Σ_T. The set of individual documents to be a subject of the test classification comprises all documents omitted in Σ'. It is assumed that the documents are presented to the system in 'chronological' order, i.e., if $\sigma_k = \langle d_{k1}, ..., d_{km}, ..., d_{kl} \rangle$ and $\sigma'_k = \langle d_{k1}, ..., d_{km} \rangle$ then d_{km+i} has to be classified before d_{km+i+1}. Of course, due to the above described construction, for each test document a proper case to which it should be assigned is known.

The problem of classification documents to cases is more difficult than the standard text categorization problem. It is not enough to decide which category a document belongs to. It has to be attached to an existing case, at its end, or should initiate a new case. Thus, the essence of the classifier construction is to learn some rules linking documents in cases of a given category. It should be noted that such a classification is carried out manually at the companies by their employees. Their job is somehow easier as they can understand the meaning of the document and take advantage of the metadata usually accompanying the document (the date, the addressee and sender, some reference in the document to other documents belonging to its case etc.). In particular, documents which are generated by a given company are usually clearly related to some cases. Hence, our formulation of the problem is more general and requires a classifier to make a decision based only on the content of the given document and the knowledge of the characteristic features of cases in particular categories, learned from a training data set. However, such a general formulation still applies to many practical problems. For example, if the legacy of a person has to be organized then the metadata available may be scarce and there are practically no additional clues except the content of the documents.

The novelty of the problem is related to the need to combine two perspectives: a standard text categorization perspective and a case-based classification one. The former problem has been thoroughly studied in the literature and various well-known techniques of supervised learning have been applied to solve it. The latter classification problem has not yet been clearly and explicitly identified in the literature and only some slightly related formulations

and solution have been proposed, cf., e.g., the Topic Detection and Tracking (TDT) problem [2]. Here we are mostly interested in dealing with this latter classification problem but the first formulation will also play an important role in our considerations, and – from a more general perspective – a synergistic combination of these two problems and their solutions is a real challenge to be tackled.

3. A Hidden Markov Model based approach

In what follows, we assume that the documents in question are represented as vectors over an appropriate space. The dimensions may correspond to particular *keywords* (*terms*) from a set T, $t = \{t_1, ..., t_m\}$, as in the classical vector space model (cf. [3]), to the *topics* identified using the Latent Dirichlet Allocation modelling or to any other entities used in various approaches to the modelling of documents within the information retrieval realm.

We want to model the succession of documents in case, specific for particular categories. Intuitively, a subsequent document in a case may be treated as corresponding to a step in the development of a given case. Referring to a previous example of a case meant as a tender for the purchase of equipment, we can distinguish such steps as: preparation of the terms and conditions of the tender; receiving questions of potential providers and answering them; receiving offers; making a choice of the provider; preparing a contract etc. The steps may overlap in time, e.g., some potential providers may submit they offers while others may still pose questions concerning the terms and conditions. Thus, the succession of documents has a probabilistic character and, moreover, the steps cannot be directly identified based on the documents themselves. Hence, for modelling the cases, we propose to employ Hidden Markov Models (HMM), cf., e.g., [7]. We distinguish the following elements:

- *hidden states* $S = \{S_1, S_2, ..., S_L\}$ which may be interpreted in the context of our problem as corresponding to particular stages of a given case type; e.g., various steps in the tender procedure mentioned above; a state in a time moment t will be denoted as q_t,
- *observations* generated by an HMM in subsequent steps, corresponding here to the whole documents (in fact, vectors representing them) d forming a case; thus a multidimensional continuous space of observations is assumed here,
- *a state transition matrix* $A = [a_{ij}]$ defining the probability of going from one state to another, $a_{ij} = P(q_{t+1} = S_j | q_t = S_i)$, $1 \le i, j \le L$,
- observation probability distributions b_j defined for each state j in the space of documents D; we will assume that these distributions are mixtures of M normal distributions: $b_j(d) = \sum_{m=1}^{M} w_{jm} \mathrm{N}(d, \mu_{jm}, U_{jm})$, $\sum_{m=1}^{M} w_{jm} = 1$, $w_{jm} \ge 0$, $1 \le j \le L$, where w_{jm} is a weight of the m-th component of the mixture for a state S_j, $\mathrm{N}(d, \mu_{jm}, U_{jm})$ is a normal distribution in the space of document vectors with the mean vector μ_{jm} and the covariance matrix U_{jm}; a vector of all such distributions will be denoted as B; a matrix of the weights of the components of the mixtures for particular states will be denoted as W; set of the matrices of the mean vectors of these components will be denoted Ξ, and a set of the matrices of corresponding covariance matrices will be denoted Ψ,
- an initial probability distribution in the space of states, $\pi = [\pi_1, \pi_2, \ldots, \pi_L]$ where $\pi_j = P(q_1 = S_j)$, $1 \le j \le L$.

A separate training collection of complete cases for each category is assumed to be available. We start by setting a number of states L and then a separate HMM, $\lambda_c = (A_c, B_c, \pi_c, W_c, \Xi_c, \Psi_c)$, for each category c is trained which results in the learning of the probability distributions A_c, B_c and π_c as well as the parameters Ξ_c and Ψ_c of the mixtures and normal distributions mentioned above. The standard EM based algorithm is used to train each HMM [7].

The classification of a new incoming document d^* is carried out as follows. A set of on-going cases forming a current collection of documents arranged in cases for each category is available. For each such case $\sigma = (d_1, d_2, ..., d_p)$ a *matching degree md* of the document d^* against the case σ is computed as the conditional probability that the HMM λ_c will generate the sequence of documents currently forming the case extended with the document d^* added at its end:

$$md(\sigma, d^*) = \mathrm{P}_\sigma(d_1, d_2, ..., d_p, d^* \mid d_1, d_2, ..., d_p, \lambda_c) = \frac{\mathrm{P}(d_1, d_2, ..., d_p, d^* \mid \lambda_c)}{\mathrm{P}(d_1, d_2, ..., d_p \mid \lambda_c)} \tag{1}$$

For each category an 'empty' story is also considered and then (1) takes the following form:

$$\mathrm{P}_\sigma(d^* \mid \lambda_c) = \sum_{j=1}^{L} \pi_j b_j(d^*) \tag{2}$$

The document d^* is assigned to such a case σ^* that:

$$\sigma^* = \arg\max_{\sigma} md(\sigma, \theta^*) \tag{3}$$

There is a number of parameters which have to be set before the above described classification can be carried out. Among them, the number of states of HMMs representing cases in particular categories have to be determined. They may be set experimentally and/or via the analysis of a collection at hand. For instance, the number of states may be related to the average length (number of documents) of completed cases in a given category.

4. A sequence mining based approach

The second approach to solving the new text categorization problem we propose in this paper consists of using the *sequence mining* approach [1, 15] to model logical relations between the documents in a case of within a given category. Let us adopt the following notation [15]. Documents are now treated as sets of keywords, $d_i \subseteq T$, and the task of sequence mining boils down in our context, to finding *sets of keywords* $f_i \subseteq T$ frequently appearing in stories of a given category.

Let $F = \langle f_1, f_2, ..., f_r \rangle$ and $G = \langle g_1, g_2, ..., g_s \rangle$ denote sequences of sets of keywords. The sequence F is said to be a *subsequence* of sequence G, denoted as $F \prec G$, if there exists such an injection $h,\sim h : \{1, 2, ..., r\} \rightarrow \{1, 2, ..., s\}$, such that:

$$\forall_{f_i \in F} \; f_i \subseteq g_{h(i)} \wedge \left((i < j) \Rightarrow \left(h(i) < h(j) \right) \right) \tag{4}$$

It should be noted that a sequence of documents (a case) σ may be viewed to be a sequence of sets of keywords because we assume here that a document is represented by a set of keywords. It is then said that a case σ *contains* a sequence of sets of keywords F if $F \prec \sigma$. The *support of a sequence of sets of keywords F* in a set of cases Σ is defined to be the number of cases containing sequence F, which may be denoted as:

$$supp\left(F, \Sigma\right) = \left\{ \sigma_i \in \Sigma | F \prec \sigma_i \right\} \tag{5}$$

where $\|\cdot\|$ denotes the cardinality of the corresponding set.

A sequence of sets of keywords is said to be *frequent* in a given set of stories Σ if its support is greater than some threshold value *min_supp*:

$$supp\left(F_{cz\square sta}, \Sigma\right) \geq min\_supp \tag{6}$$

There are many algorithms, exemplified by SPADE [15], which make it possible to discover all frequent sequences of sets of keywords for a given set of cases.

Using frequent sequences one may determine *rules* describing dependencies between the occurrence of particular sets of keywords. For example, if:

$$\begin{gathered} supp\left(F, \Sigma\right) = x \\ supp\left(G, \Sigma\right) = y \\ F \prec G \end{gathered} \tag{7}$$

then it is said that the rule:

$$F \Rightarrow G$$

holds with a confidence level *conf* equal:

$$conf\left(F \Rightarrow G\right) = \frac{y}{x} \tag{8}$$

We are interested in particular in so-called *strong rules* $F \Rightarrow G$ such that:

$$conf\left(F \Rightarrow G\right) \geq min\_conf \tag{9}$$

if:

$$F = \langle f_1, f_2, ..., f_r \rangle \quad \text{then} \quad G = \langle f_1, f_2, ..., f_r, g_{r+1} \rangle \tag{10}$$

where *min_conf* denotes some required minimal level of the rule confidence.

Such rules will be denoted in a simpler form as:

$$F \Rightarrow g_{r+1} \tag{11}$$

We will now describe how frequent sequences are used to solve our new text categorization problem. During the training phase a collection of cases for each category is assumed to be available. This time, as mentioned earlier, a document is represented by a set of keywords. For each collection of cases, frequent sequences are discovered and their corresponding strong rules are generated, cf. (11).

To classify a new document d^*, we proceed as follows:

1) for each case σ all *active rules* are considered which match this case, i.e., such rules in which the left hand side sequence of sets of keywords F is a subsequence of the case $\sigma : F \prec \sigma$; for a given case, only the rules generated for the category to which this case belongs are taken into account;
2) among the rules $F \Rightarrow g_{r+1}$ we count those for which the right hand side g_{r+1} is a subset of the document to be classified, i.e., $g_{r+1} \subseteq d^*$ (it should be noted that both the documents and the right hand sides of the rules are sets of keywords); rules having the same right hand sides are counted only once;
3) the document d^* is classified to such a case for which the number of rules counted in step 2 is the highest, and also higher than a certain threshold value *min_count*; if there is more than one such a case, then one of them is randomly selected;
4) if there is no such case for which the number of rules counted as in step 2 is higher than *min_count*, then such a document d^* starts a new story in the category which is selected using a standard text categorization algorithm, e.g., based on the Naïve Bayes approach [6, 13].

In the next section, we present the results of some computational experiments using this algorithm.

5. Computational experiments

In section 2, we have formally introduced a new text categorization problem. There are no standard datasets to test the proposed methods of its solution. Thus, we adapted the ACL Anthology Reference Corpus (ACL ARC) [4] for our purposes. It consists of 10291 scientific papers on computational linguistics. In our preliminary experiments, we employed a subset of this corpus. Each paper comprises a number of explicitly distinguished sections. For our purposes we identified each paper with a sequence of documents σ and its sections are particular documents in such a sequence. This way we obtained 113 sequences, consisting of 11 documents on average.

All documents were represented using the vector space model [3] and the tf x IDF weighting scheme with normalization with respect to the vector length, in particular. Standard document processing techniques were applied, such as stopwords elimination and stemming.

The documents were grouped into categories using the k-means algorithm based cluster analysis. The number of categories was chosen experimentally to be equal 7. Two clusters were ignored due to their small cardinality and finally, a set of 98 document sequences was obtained.

The cSPADE algorithm was used to generate a set of rules (11) for each category. The arulesSequences package for the system R [8] was employed. For sequences mining the representation of each document was limited to the 10 upper-most keywords with respect to their tf x IDF weight in a given document.

The classification algorithm was run four times, each time randomly selecting the test set of documents using the procedure described in section 2. The results of these four runs are briefly presented in Table 1.

Table 1

Results of the computational experiments with the algorithm based on sequences mining

Run No.	Total number of classified documents	Number of correctly classified documents	Microaveraged precision	Macroaveraged precision over the categories
1	252	208	0.8254	0.8531
2	245	204	0.8326	0.8576
3	241	197	0.8174	0.8481
4	252	201	0.7976	0.8101

In virtually all runs, precision of at least 80% was obtained. The results are thus encouraging but the experiments have to be continued with larger data sets as well as with real data sets of cases. Such data are not easy to get but we are in the process of building a collection of documents of one of the public administration institutions in Poland.

6. Conclusions

We have defined a novel and extended text classification related problem that combines issues relating to the acquisition, maintenance and handling of documents in a corporate and institutional setting. We proposed a formal statement of the problem and two approaches to solve it. A pilot implementation of one of the algorithms has been implemented and some preliminary computational experiments hve been carried out [12]. The results obtained are promising but some further experiments are needed to confirm the effectiveness of the proposed method.

This research was partially supported by the National Science Centre (contract No. UMO-2011/01/B/ST6/06908).

TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE
AUTOMATIC CONTROL | AUTOMATYKA
4-AC/2013

MAŁGORZATA CHARYTANOWICZ*

NONPARAMETRIC ESTIMATION FOR SOIL PORE SIZE DISTRIBUTION

NIEPARAMETRYCZNA ESTYMACJA ROZKŁADU WIELKOŚCI PORÓW GLEBOWYCH

Abstract

The study is concerned with the nonparametric kernel estimation to determine the soil porosity and pore size distribution. The kernel density estimation, the kernel estimation of cumulative distribution function, and the kernel estimator of quantile are considered. After a short description of the method, practical aspects and applications in agricultural science are presented. The nonparametric kernel estimation does not require a priori assumptions relating to the choice of the density function shape. Moreover, its natural interpretation together with its suitable properties makes them an adequate tool among others in estimation methods.

Keywords: *nonparametric estimation, kernel estimators, cumulative distribution function, kernel estimator of quantile, pore size distribution, pore space, total porosity*

Streszczenie

Przedmiotem niniejszego artykułu jest zastosowanie nieparametrycznej estymacji jądrowej do scharakteryzowania rozkładu wielkości porów glebowych. W artykule przedstawiono jądrowy estymator gęstości i dystrybuanty oraz opisano algorytm wyznaczania jądrowego estymatora kwantyla, istotne ze względu na badanie porowatości agregatów glebowych. Zagadnienia te zostały zilustrowane przykładowymi zastosowaniami w naukach rolniczych. Nieparametryczna estymacja jądrowa nie wymaga *a priori* założeń dotyczących kształtu funkcji gęstości rozkładu prawdopodobieństwa i jest uzasadniona w sytuacji braku znajomości jej teoretycznego modelu. Ze względu na swobodę w doborze jądra oraz procedur wyznaczania parametrów estymatora możliwe jest dostosowanie jego własności do uwarunkowań konkretnego problemu.

Słowa kluczowe: *estymacja nieparametryczna, estymatory jądrowe, agregaty glebowe, rozkład wielkości porów, porowatość gleby*

* Ph.D. Małgorzata Charytanowicz, e-mail: malgorzata.charytanowicz@ibspan.waw.pl, Institute of Mathematics and Computer Science, The John Paul II Catholic University of Lublin; System Research Institute, Polish Academy of Sciences, Warsaw.

1. Introduction

Along with the development of statistical methods, it is evident that classical parametric techniques are widely used in empirical studies [5, 10, 14]. This approach to density estimation assumes that the data are drawn from one of a known parametric family of distributions, determined by their parameters. The density underlying the data could then be estimated by finding from the data estimates of unknown parameters, using, for example, maximum likelihood estimation and substituting these estimates into the formula for the chosen density [3]. Such attitude requires performing goodness-of-fit tests on the data, in which the null hypothesis states that our data follows a specific distribution. When performing a statistical test, one never concludes that the null hypothesis is accepted. Whenever it is plausible that the data are consistent with the null hypothesis, the reached conclusion is not significant and does not mean that this hypothesis is true. The value of the test is statistically significant however, only in a negative case, when a decision of rejecting the tested hypothesis has been made. This leads to some subjectivism in the presented attitude, increasing rapidly in the multidimensional case, in which normal distribution is most commonly used. Moreover, for skewed distributions, a mathematical transformation, for example, logarithmic, tending the data to normal distribution, is recommended. Some data cannot be transformed satisfactorily and some should not to be. Additionally, this proceeding hinders the interpretation of the results obtained. These difficulties of the parametric approach tend to cause researchers to look for other estimation methods.

The rigidity of parametric models can be overcome by removing the restriction that the density belongs to a parametric family and assumes no pre-specified functional form for a density function. This approach leads to a nonparametric estimation method [5, 15]. Some of these methods have the advantage of being very intuitive and relatively simple to analyze mathematically. The oldest and most widely used nonparametric density estimator is the histogram. The histogram has several problems – like estimating density by a step function, the extension to multivariate settings, and finally, not using the data efficiently. Kernel smoothing provides a simple method of finding structures in data sets without these deficiencies. This technique is most relevant in many practical tasks [6, 7, 13], for example, as an effective tool for quantile estimators, especially when the underlying distribution is skewed. Moreover, because of ongoing research into the computer implementation of the algorithm, it is worth noting that all parameters appearing in the model can be effectively calculated using convenient numerical procedures based on optimizing criteria.

The main aim of this paper is to elaborate on a nonparametric estimation method, based on statistical kernel estimators, which can be successfully applied for determining the soil pore size distribution. The studies of employing pore space have been reported as a general method for defining soil structures. The most common measure characterizing the pore space within a solid is the total porosity defined as a fraction of the total pore volume that is taken up by the pore space. On the other hand, soils may be nearly uniform in regard to their total porosity but differ in pore size distribution [12, 20]. Their individual properties have different influences on fluid retention and conduction within the soil. Therefore, besides the total pore volume, their size and distribution is very important with respect to soil structure and soil fertility. These characteristics impact the majority of physical and physicochemical soil parameters, such as water retention, water conductivity, aeration, erosion susceptibility and gas diffusivity. They vary significantly due to several factors, including mutual interactions between soil fauna,

soil macro-organisms, roots, inorganic agents and environment factors. Weather conditions, vegetation, fertilization and soil tillage operations can cause both soil loosening or compaction. As far as water-air soil properties are concerned, pores are usually divided into three groups: micropores, mesopores and macropores, with the division between them being arbitrary. Macropores, relevant to coarse-grained soils, drain freely by gravity and allow easy movement of water and air. They provide a habitat for soil organisms and plant roots, which can grow into them. On the other hand, macropores cannot hold water under tension induced by gravity when allowed to drain after saturation. Inverse properties are found in fine-grained soils, containing a major amount of micropores. These soils retain large amounts of water, due to the fact that this water would be considered unavailable to plants. Mesopores, i.e. medium-size pores, are essential for capillary water distribution. They provide water storage sites which retain water useful to plants. Soils with a predominance of mesopores and a moderate system of micro- and macropores possess most favorable physical properties relevant to plant growth.

A more detailed analysis can be obtained by developing a pore-size distribution curve. Soil pore size distribution is often determined by mercury intrusion porosimetry or low temperature nitrogen adsorption. However these methods do not provide the all the information about pore size and shape, as they are not appropriate for pore measurements. Recent advances in computed tomography and digital image processing algorithms [4, 11, 17, 19] provide technologically advanced measurement tools for studying the internal structures of soil aggregates.

In elaborate investigations, the internal structure of an aggregate was visualized by microtomography scanning. A research study was conducted using image analysis algorithms appropriate for pore measurements, and in turn, kernel estimation techniques. After a short description of the method, practical aspects and applications in agricultural science are presented.

2. Statistical Kernel Estimators

Let (Ω, Σ, P) be a probability space. Let a real random variable $X : \Omega \to R$, whose distribution has the density function f, also be given. The corresponding kernel estimator $\widehat{f} : R \to [0, \infty)$, calculated using experimentally obtained values for the m-element random sample $x_1, x_2, ..., x_m$, in its basic form is defined by:

$$\widehat{f}(x) = \frac{1}{mh} \sum_{i=1}^{m} K\left(\frac{x - x_i}{h} \right) \tag{1}$$

where $m \in \mathbf{N}/\{0\}$, the positive coefficient h is known as a smoothing parameter, whereas the measurable function $K : R \to [0, \infty)$ of unit integral, symmetrical with respect to zero and having a weak global maximum at this point, takes the name of a kernel. The influence of the smoothing parameter on particular kernels is the same for the basic definition of the kernel estimator (1). Advantageous results are obtained thanks to the individualization of this effect, achieved through a so-called modification of the smoothing parameter [6, 15, 18]. It relies on mapping the positive modifying parameters $s_1, s_2, ..., s_m$ on particular kernels, described as:

$$s_i = \left(\frac{\widehat{f}(x_i)}{\overline{s}} \right)^{-c} \tag{2}$$

where $c \in [0, \infty)$, $\widehat{f}$ denotes the kernel estimator without modification, and $\overline{s}$ is the geometrical mean of the numbers $\widehat{f}(x_1), \widehat{f}(x_2), ..., \widehat{f}(x_m)$. The parameter c stands for the intensity of the modification procedure and based on indications for the criterion of the integrated mean square error, the value $c = 0.5$ can be suggested. Finally, the kernel estimator with the smoothing parameter modification is defined in the following formula:

$$\widehat{f}(x) = \frac{1}{mh} \sum_{i=1}^{m} \frac{1}{s_i} K\left(\frac{x - x_i}{hs_i} \right) \tag{3}$$

Specifying the kernel estimator of a density function f, gives a natural description of the distribution of X, and allows the estimator of the distribution function, denoted hereinafter as $\widehat{F} : R \to [0, 1]$, to be found from the relation:

$$\widehat{F}(x) = \int_{-\infty}^{x} \widehat{f}(u)\,\mathrm{d}u \tag{4}$$

where $\widehat{f}$ denotes the kernel density estimator (3). Denoting the primitive of a kernel K as $K : R \to [0, 1]$, that is:

$$I(x) = \int_{-\infty}^{x} K(u)\,\mathrm{d}u \tag{5}$$

the kernel estimator of the distribution function can be expressed as:

$$\widehat{F}(x) = \frac{1}{m} \sum_{i=1}^{m} I\left(\frac{x - x_i}{hs_i} \right) \tag{6}$$

If for the estimator (3), one uses a kernel with positive values, then the function I, and thus $\widehat{F}$ are strictly increasing.

In the case when the kernel estimator of the distribution function $\widehat{F}$ is used, the kernel estimator of the quantile of order r denoted as $\widehat{g} \in R$, may be uniquely defined by the solution of the equation:

$$\widehat{F}(x) = r \tag{7}$$

The equation (7) can be expressed equivalently in a form:

$$\widehat{F}(x) - r = 0 \tag{8}$$

If the left side of the equation (8) is denoted by $L(x) = \widehat{F}(x) - r$, then $L'(x) = \widehat{f}(x)$, and the kernel estimator of the quantile can be effectively calculated on the basis of Newton's algorithm [2, 8, 16] as the limit of the sequence $\left\{\hat{q}_k\right\}_{k=0}^{\infty}$ defined by:

$$\hat{q}_0 = \frac{1}{m}\sum_{i=1}^{m} x_i \tag{9}$$

$$\hat{q}_{k+1} = \hat{q}_k - \frac{\widehat{F}(\hat{q}_k) - r}{\widehat{f}(\hat{q}_k)} \quad \text{for} \quad k = 0, 1, \ldots \tag{10}$$

The choice of the kernel K form and the calculation of the smoothing parameter h is made most often with the criterion of the mean integrated square error [1, 9]. From a statistical point of view, the choice of the kernel form has no practical meaning and thanks to this, it becomes possible to take into account primarily properties of the estimator obtained or calculation aspects, advantageous from the viewpoint of the application problem under investigation. The standard normal kernel given by:

$$K(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} \tag{11}$$

is used most often. It is differentiable up to any order and assumes positive values in the whole domain.

The fixing of the smoothing parameter h has significant meaning for the quality of estimation. A smoothing parameter controls the tradeoff between bias and variance in the result. A large bandwidth leads to a very smooth density distribution, whereas a small bandwidth leads to a ragged density distribution. A frequently used bandwidth selection technique, called the 'cross-validation method', chooses h to minimize the function $g : R \rightarrow R$ defined as:

$$g(h) = \frac{1}{m^2 h}\sum_{i=1}^{m}\sum_{j=1}^{m}\tilde{K}\left(\frac{x_j - x_i}{h}\right) + \frac{2}{mh}K(0) \tag{12}$$

where:

$$\tilde{K}(x) = K^{*2}(x) - 2K(x) \tag{13}$$

whilst K^{*2} denotes convolution function of k, i.e.:

$$K^{*2}(x) = \int_R K(u)K(x-u)\,\mathrm{d}u \tag{14}$$

The tasks concerning the choice of the kernel form, as well as additional procedures improving the quality of the estimator obtained, and all rules needed for calculating the smoothing parameter, are found in [6, 15, 18]. The utility of kernel estimation has been investigated in the context of determining the soil pore size distribution.

3. Material and Methods

The investigated material was sampled from the cultivated soil layer classified as silty loam (WRB Mollic Gleysols), explored at the Institute of Agrophysics, at the Polish Academy of Sciences in Lublin. The proportion of each particle size group in the soil was as follows: sand – 46%, silt – 28%, clay – 26%, pH was: H_2O – 5.9, KCl – 5.4. On the experimental fields, a long-term fertilization trial was executed. The adopted crop rotation from 1955 to 1989 was a cycle of potato – barley – rye, and from 1990 – a cycle of sugar beat – barley – rape – wheat. Three treatments concerning fertilization: control group – plant residues only, mineral fertilization – according to plant needs; pig manure – 80 ton per ha; were studied. Aggregate soil organic matter was measured by the Multi N/C 3100 Autoanalyser (Analitic Jena, Germany). The total organic carbon and total nitrogen contents for three fertilizations (pig manure, mineral fertilizers, control) were respectively: 21.50, 14.89; 13.54 g/kg; 2.10; 1.51; 1.35 g/kg. The total organic carbon shows the same tendency as total nitrogen, i.e. increasing in the same order: the lowest – control, middle – mineral fertilization, the highest – pig manure.

Soil samples were air dried in room conditions, divided into smaller amounts, and gently sieved through 2 and 10 mm sieves. Soil aggregates remaining at 2 mm sieve and ranging from 2 to 10 mm were then detected by means of *X*-ray computational tomography. The direct and nondestructive analysis of internal soil aggregate structure was detected using a GE Nanotom S device, with the voxel-resolution of 2.5 microns per volume pixel. Three 2D sections uniformly located within each aggregate were performed to characterize the aggregate structure. Next, tomography sections were processed using the Aphelion 4.0.1 package. Thus pore size distribution of a particular aggregate was determined by means of image processing techniques. In the initial step, the ROI (region of interest) selection from the original grayscale image was performed. All of the ROI's were selected by hand, around the aggregate, removing the ring artifacts, and next they were saved as a bitmap format. The automatic Otsu binarization method was then employed to separate pores within the sampled aggregate. Subsequently, binary morphological closing with increasing size of square structuring element was processed. The operation was repeated until all pores were filled. Subtraction of the transformed image from the original image gives the total pore volume in the examined aggregate. Pore identification and the determination of pore radius were done automatically using morphological operations, clustering and splitting procedures. Each pore was individually identified and approximated by a circle of radius r, calculated from the surface area.

Finally, the pore size distribution was presented in the form of pore radius distribution curves using kernel density estimation (as described in section 2). A more detailed analysis was obtained by kernel estimation of the distribution function (6) and kernel estimation of the quantile (7).

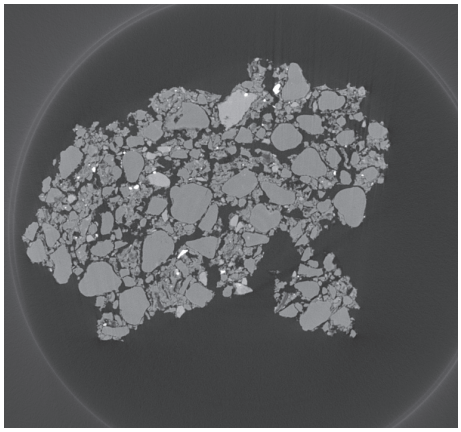

Fig. 1. The soil aggregate microtomographic image

4. Estimation of the Pore Size Distribution

The examined soil samples comprised aggregates belonging to three treatments, different in terms of fertilization: selected for the experiment control (without fertilization); mineral fertilization, and pig manure. High quality visualization of the internal soil structure was detected using a non-destructive soft *X*-ray technique. Tomography sections were processed using the image processing methods described in section 3. The total porosity of the investigated aggregates, calculated as the average of three sections, are as follows: control group – 14.2%, mineral fertilization – 22.95%, manure fertilization – 33.28%. The fraction of the total aggregate volume occupied by soil pores was significantly greater when manure fertilization was used. Moreover the total porosity was higher for both the manure and the mineral fertilization than for the control group.

For each fertilization, the examined group constitutes a one-dimensional sample containing 510 measurements of pore radii ranging from 5 μm to 0.24 mm. Kernel density estimates (3) for soil pore size, based on the data, were constructed using the Cauchy kernel given by the rule:

$$K(x) = \frac{2}{\pi} \frac{1}{(1+x^2)^2} \tag{15}$$

Its primitive has a form convenient for further calculations:

$$I(x) = \frac{1}{\pi}\left(\frac{x}{1+x^2} + \operatorname{arctg}(x) + \frac{\pi}{2} \right) \tag{16}$$

The smoothing parameters h detected by the *cross-validation* method equal to 1.06, 2.96, and 1.49 for the samples belonging to the control group, mineral and manure fertilized aggregates

respectively. The soil pore size distributions evaluated from the databases of examined groups, constructed by means of the kernel density estimates, are shown in Fig. 2.

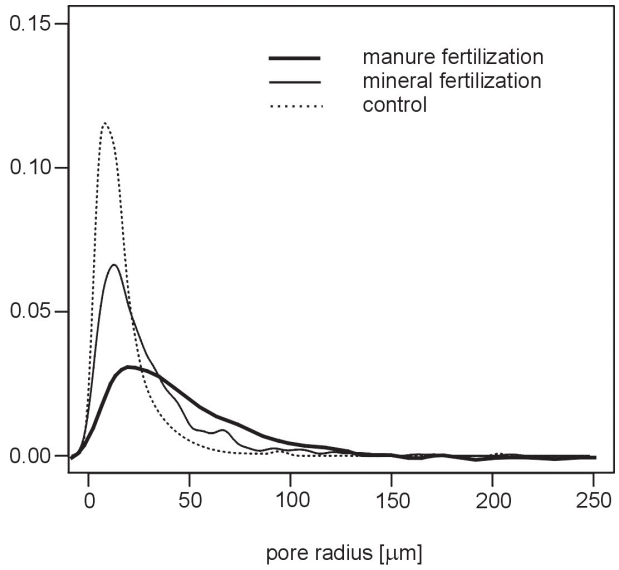

Fig. 2. Kernel density estimates for soil pore size

As shown, both unimodal and right-skewed distributions are well developed for all three data samples. It should also be noted that the region between 0 to 50 μm is the most crucial representation of the pore size interval of the control group, whereas the region between 0 to 100 μm is the most representative pore size interval of the fertilized samples. Moreover, the first considered region is somewhat narrowed compared with the two regions of the fertilized samples. Furthermore, the highest and sharpest peak was found for the control group, followed by the mineral and manure fertilizations. This peak comes about as a result of the substantial contribution of narrow pores evident within the samples, and the control group was seen to contain more narrow pores than the fertilized (both mineral and manure) samples. In our data, upon analysis it was evident that the second and third peaks are moved to the right side. This, we feel, indicates a greater proportionality of less narrow pores. This characteristic feature complies well with the major aim of the study – to present a reliable density estimation method for evaluating pore size distribution patterns.

In order to perform further analysis of micro, meso and macropores the kernel estimator of the distribution function (6) was constructed. The limits of mesopores, taken arbitrarily, were between 30 and 75 μm. Table 1 contains fractions of micro, meso and macropores for each type of fertilization.

Calculated fractions confirm the results obtained by means of kernel density estimates, as given in Figure 2. The largest fraction of mesopores occurs in the soil fertilized with pig manure, this fraction represents 39% of the total pore area. Moreover, it contains percentages of 26% macropores and 35% micropores. This creates the most favorable conditions for plant growth. The largest fraction of micropores, equaling 87%, was observed in the soil without

fertilization. This soil has a small amount of macropores, equaling 1%, and a small amount of mesopores, equaling 12%. This creates the least favorable conditions for plant growth. The soil with mineral fertilization contains 65% of micropores, 28% of mesopores and 7% of macropores.

Table 1

Fractions of micro-, meso- and macropores

Management fertilization	Fraction of micropores	Fraction of mesopores	Fraction of macropores
Pig manure	0.35	0.39	0.26
Mineral fertilization	0.65	0.28	0.07
Control group	0.87	0.12	0.01

Table 2 shows quantile estimators of order 0.25, 0.5, and 0.75, calculated using rules (9)–(10).

Table 2

Quantile estimators of order 0.25, 0.5, and 0.75

Management fertilization	Quantile of order 0.25 [μm]	Quantile of order 0.5 [μm]	Quantile of order 0.75 [μm]
Pig manure	22.63	43.50	76.50
Mineral fertilization	12.83	22.35	38.98
Control group	7.45	13.00	20.90

The largest fraction of large pores occurs in the soil fertilized with pig manure, 50% of its pores have a radius between 22.63 and 76.50 microns. The soil with mineral fertilization incorporates 50% of its pores having a radius between 12.83 and 38.98 microns. The largest fraction of small pores occurs in the soil without fertilization, 50% of its pores has a radius between 7.45 and 20.90 microns.

This study has shown the possibility to determine soil pore size distribution using the nonparametric kernel estimation theory. Generally, the effect of fertilization is to increase amounts of meso- and macropores in relation to the control group. The greater increase is for pig manure fertilization. Soils without fertilization contain significantly more micropores. These soils possess less favorable properties for plant growth.

5. Summary

Recent advances in computed tomography and digital image processing provide non-destructive tools for studying the internal structures of soil aggregates. This seems very useful in characterizing the pore size distribution and in quantifying the differences in pore structures

from the different types of soil. A more detailed analysis may be obtained by deriving various methods to quantify the pore structure and to develop a pore size-distribution curve.

In this paper, an innovative method for characterizing the soil porosity and pore size distribution, based on computed tomography and nonparametric estimation, is proposed. The presented algorithm, based on image processing methods and the kernel estimators technique, is expected to be an effective procedure for this purpose. The presented approach is more objective than classical parametric methods, and can be successfully applied for many tasks in data mining, where arbitrary assumptions concerning the form of density function are not recommended.

This approach is also motivated by the current rapid growth in computational power. Improved real-time data processing and algorithm efficiency having important meaning due to the concurrent increase in the quantity and complexity of the data that are being collected. Historically, such data have been analyzed using classical methods. The presented approach is useful for determining the pore size distribution of any material for which a morphometric analysis is done, as it eliminates estimation subjectivity.

References

[1] Draper N.R., Smith H., *Applied regression analysis*, John Wiley and Sons, New York 1981.

[2] Kincaid D., Cheney W., *Numerical Analysis*, Brooks/Cole, Pacific Grove, 2002.

[3] Koronacki J., Mielniczuk J., *Statystyka dla studentów kierunków technicznych i przyrodniczych*, WNT, Warszawa 2001.

[4] Król A., Niewczas J., Charytanowicz M., Gonet S., Lichner L., Czachor H., Lamorski K., *Water-stable and non-stable soil aggregates and their pore size distributions*, 20th International Poster Day and Institute of Hydrology Open Day 'Transport of water, chemicals and energy in the soil – plant – atmosphere system', 2012, 870-871.

[5] Kruszewski D., *Nonparametric modeling of medical scheme data*, Technical Transactions,1-AC/2013, 93-117.

[6] Kulczycki P., *Estymatory jądrowe w analizie systemowej*, WNT, Warszawa 2005.

[7] Kulczycki P., Charytanowicz M., *Conditional Parameter Identification with Different Losses of Under- and Overestimation*, Applied Mathematical Modelling, 37 (4), 2013, 2166-2177.

[8] Kulczycki P., Dawidowicz A.L., *Kernel Estimator of Quantile*, Universitatis Jagiellonicae Acta Mathematica, 37, 2005, 101-112.

[9] Lange K., *Numerical analysis for statisticians*. *Statistics and Computing*, Springer, New York 2000.

[10] Motulsky H., *Intuitive Biostatistics*, Oxford University Press, New York 1995.

[11] Peth S., Nellesen J., Fischer G., Horn R., *Non-invasive 3D analysis of local soil deformation under mechanical and hydraulic stresses by μCT and digital image correlation*, Soil and Tillage Research, 111 (1), 2010, 3-18.

[12] Pires de Silva A., Imhoff S., Kay B., *Plant response to mechanical resistance and air-filled porosity of soils under conventional and no-tillage system*, Scientia Agricola, 61 (4), 2004, 451-456.

[13] Sheather S.J., Marron J.S., *Kernel quantile estimators*, Journal of the American Statistical Association, 85, 1990, 410-416.
[14] Siegel A.F., *Statistics and data analysis: an introduction*, Wiley and Sons, New York 1988.
[15] Silverman B.W., *Density Estimation for Statistics and Data Analysis*, Chapman and Hall, London 1986.
[16] Stoer J., Bulirsch R., *Wstęp do analizy numerycznej*, PWN, Warszawa 1987.
[17] Tadeusiewicz R., Korohoda P., *Komputerowa analiza i przetwarzanie obrazów*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1997.
[18] Wand M.P., Jones M.C., *Kernel Smoothing*, Chapman and Hall, London 1994.
[19] Wojnar L., Majorek M., *Komputerowa analiza obrazu*, Computer Scanning System, Warszawa 1994.
[20] Zdravkov B., Cermak J., Sefara M., Janku J., *Pore classification in the characterization of porous materials*: *A perspective*, Central European Journal of Chemistry, 5 (2), 2007, 385-395.

TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE
AUTOMATIC CONTROL | AUTOMATYKA
4-AC/2013

LÁSZLÓ GÁL*, RITA LOVASSY**, LÁSZLÓ T. KÓCZY***

BACTERIAL TYPE ALGORITHMS USED FOR FUZZY RULE BASE EXTRACTION

WYKORZYSTANIE ALGORYTMÓW BAKTERYJNYCH DO WYDOBYCIA BAZY REGUŁ ROZMYTYCH

Abstract

The paper gives an overview of various bacterial type evolutionary algorithms used for fuzzy rule based identification. In order to find an optimal rule base from the input-output training data set, several improved algorithms have been developed in recent years. The task is to increase the models' accuracy and convergence speeds by modifying a part of the Mamdani-type inference system.

Keywords: *FRBI, Pseudo-Bacterial Genetic Algorithm, Bacterial Evolutionary Algorithm, Bacterial Memetic Algorithm with Memetic Mutation, Progressive Bacterial Algorithm*

Streszczenie

W artykule zawarto przegląd ewolucyjnych algorytmów bakteryjnych wykorzystywanych do identyfikacji bazy reguł rozmytych. W celu znalezienia optymalnej bazy reguł ze zbioru danych testowych wejściowych i wyjściowych, w ostatnich latach opracowano kilka ulepszonych algorytmów. Zamysłem przedstawionych tu badań jest uzyskanie wzrostu dokładności modeli oraz szybkości ich zbieżności poprzez modyfikację systemów wnioskowania typu Mamdaniego.

Słowa kluczowe: *FRBI, pseudo-bakteryjny algorytm genetyczny, bakteryjny algorytm ewolucyjny, bakteryjny algorytm memetyczny z mutacją memetyczną, progresywny algorytm bakteryjny*

* M.Sc. László Gál, e-mail: laci.gal@gmail.com, Department of Technology and Applied Informatics, University of West Hungary, Szombathely.

** Ph.D. Rita Lovassy, Institute of Microelectronics and Technology, Kandó Kálmán Faculty of Electrical Engineering, Óbuda University Budapest.

*** Prof. D.Sc. Ph.D. László T. Kóczy, Department of Automation, Faculty of Engineering Sciences, Széchenyi István University Györ; Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics.

1. Introduction

Special employment of fuzzy systems, the fuzzy controllers are present in every day applications. The design of fuzzy controllers is concerned with the calculus of fuzzy rules [17]. The construction of fuzzy rules, mathematically, sets of fuzzy relations, is one of the key problems of fuzzy reasoning and control. An important task in fuzzy rule extraction is how to obtain a set of appropriate fuzzy rules for a given system. The application of bacterial type algorithms (Pseudo-Bacterial Genetic Algorithm – PBGA and Bacterial Evolutionary Algorithm – BEA) for fuzzy rule base identification (FRBI) was proposed in [16, 17]. A modified, memetic version of the bacterial evolutionary algorithm called the Bacterial Memetic Algorithm – BMA was also proposed in [3]. The combination of evolutionary and gradient based algorithms was used rather successfully in global optimization approaches. In order to improve the system's convergence speed and their function approximation capabilities a series of new bacterial algorithms has been proposed by us: the Improved Bacterial Memetic Algorithm (IBMA) [4], the Bacterial Memetic Algorithm with Memetic Mutation (BMAM) [5], the Modified Bacterial Memetic Algorithm (MBMA) [6], the 3Step BMA (3BMA) [8], and the Progressive Bacterial Algorithm (PBA) [10]. In our previous papers [7, 9], we had examined how using different t-norms instead of the conventional 'min' fuzzy operator affected the system's learning capability and the convergence speed of the Mamdani-type inference system [12]. We had studied how accurately input-output data samples could be reproduced by using fuzzy rule bases obtained via an automatic rule identification process. The extensive investigations showed that the IBMA or MBMA training algorithms with non-parametric *t*-norms like algebraic, trigonometric [7], Hamacher product and a parametric operator like the Hamacher *t*-norm (with optimized parameter value) definitely improved the system learning capability.

After the Introduction, in Section 2 we will briefly review the PBGA, BEA, BMA, IBMA, BMAM, MBMA and PBA algorithms. This section provides some typical simulation results of fuzzy rule based identification processes using various algorithms. Finally, some conclusions are taken and references listed.

2. Bacterial Type Evolutionary and Memetic Algorithms

2.1. Pseudo-Bacterial genetic Algorithm (PBGA)

Nawa et al. [16] proposed a novel type of evolutionary algorithm called the Pseudo--Bacterial Genetic Algorithm (PBGA) for fuzzy rule based extraction. This is a special kind of genetic algorithm with a core that contains a new genetic operation called bacterial mutation. This method mimics the microbial evolution phenomenon. Its basic idea is to improve the parts of chromosomes contained in each bacterium. Bacteria can transfer genes to other bacteria. This mechanism is used in bacterial mutation. For the bacterial algorithm, the first step is to determine how the problem can be encoded in a bacterium (chromosome). The task is to find the optimal fuzzy rule base for a pattern set. Thus, the parameters of the fuzzy rules must be encoded in the bacterium. In general, the parameters of the rules are the breakpoints of the trapezoids, thus, a bacterium will contain these breakpoints.

The next step is to optimize the parameters. Therefore, a procedure is working on changing the parameters, testing the model obtained by this way and selecting the best. The inference system used for model calculations can be any of the various types of fuzzy inference systems (e.g. FRI [11]).

The main steps of the PBGA are as follows:

– Create the initial population: N_{Ind} individuals are randomly created and evaluated. (N_{Ind} is the number of individuals in the population.) Each individual contains N_{Fuzzy_rules} fuzzy rules encoded in the chromosome (N_{Fuzzy_rules} is the number of fuzzy rules of the desired model).
– Apply the bacterial mutation to each individual:
  – Each individual is selected one by one.
  – N_{Clones} copies of the selected individual are created ('clones').
  – Choose the same part or parts randomly from the clones and mutate it (except one single clone that remains unchanged during this mutation cycle).
  – Select the best clone and transfer its mutated part or parts to the other clones.
  – Repeat the part choosing-mutation-selection-transfer cycle until all the parts are mutated and tested exactly once.
  – The best individual is to remain in the population, all other clones are deleted.
  – This process is repeated until all the individuals have gone through the bacterial mutation.
– Apply conventional genetic operations (selection, reproduction and crossover).
– Repeat the procedure above from the bacterial mutation step until a certain termination criterion is satisfied (e.g. maximum number of generations).

The algorithm works efficiently where weak relationships between the parameters encoded in the chromosome exist.

2.2. Bacterial Evolutionary Algorithm (BEA)

Bacterial Evolutionary Algorithm (BEA) is based on the PBGA supported by a new genetic operation called the gene transfer operation [17]. This new operation establishes relationships among the individuals of the population.

The main steps of the gene transfer operation are:

– Sort the population according to the fitness values and divide it into two halves. The half that contains the better individuals is called 'superior half' while the other half is the 'inferior half'.
– Choose one individual (the 'source chromosome') from the superior half and another one (the 'destination chromosome') from the inferior half.

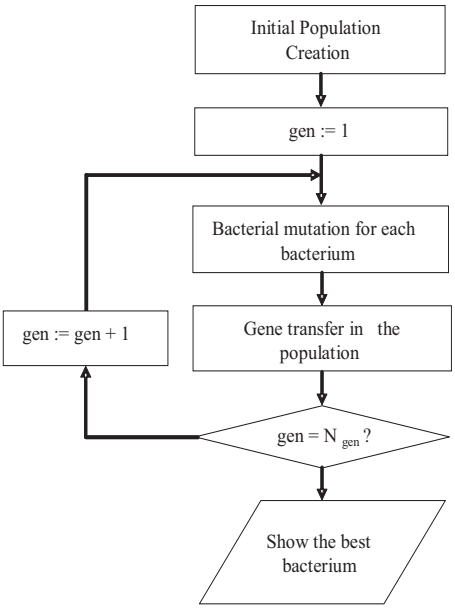

Fig. 1. Flowchart of the BEA

– Transfer a part of the source chromosome to the destination chromosome (select the part randomly or by a predefined criterion).
– Repeat the steps above N_{Inf} times (N_{Inf} is the number of 'infections' to occur in one generation).

The gene transfer operation can be used in place of selection, reproduction, or the crossover in the algorithm described by the PGBA. The BEA flowchart can be seen in Fig. 1.

2.3. Bacterial Memetic Algorithm (BMA)

Bacterial Memetic Algorithm [3] combines evolutionary and local search algorithms [15], in particular the BEA and Levenberg-Marquardt (LM) methods [14]. The algorithm main steps are (after the creation of the initial population):
– the bacterial mutation to each individual,
– a few iterations of the LM method,
– gene transfer operation applied per generation a number of infection times.

The above steps are repeated from the bacterial mutation until a certain stopping criterion is satisfied. When applying this method in the case of the trapezoidal shaped fuzzy membership functions it often happens that the trapezoid breakpoints do not satisfy a certain relationship, namely, the membership function defined by the four breakpoints cannot be interpreted as a fuzzy membership function. In this case (knot order violation, KOV) an update vector reduction factor is applied in the LM method [2].

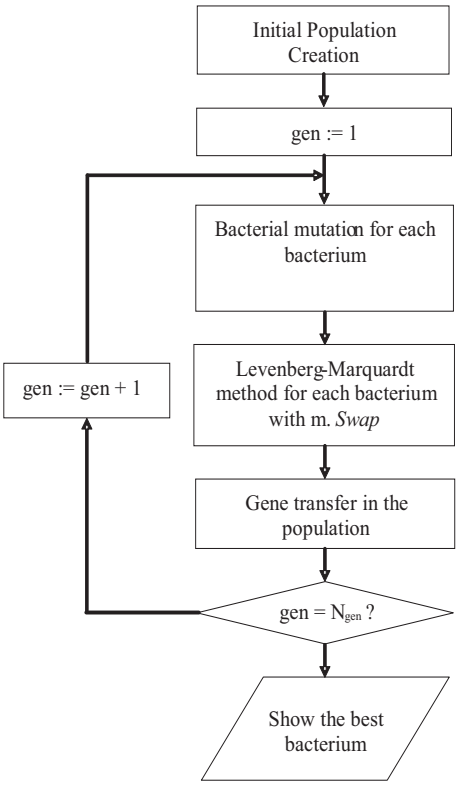

Fig. 2. Flowchart of the IBMA

2.4. Improved Bacterial Memetic Algorithm (IBMA)

Gál et al. proposed in [4] the so called *Improved Bacterial Memetic Algorithm* for a more efficient handling the knot order violations occurred in the bacterial memetic algorithm used for fuzzy rule based extraction. This method performs slightly better than the method used before. The 'merge of the violating knots into a single knot', and 'swapping of the knots that are in the wrong order' methods are introduced, which are easy to implement and to integrate info the BMA. The IBMA flowchart can be seen in Fig. 2.

2.5. Bacterial Memetic Algorithm with Memetic Mutation (BMAM) and Modified Bacterial Memetic Algorithm (MBMA)

Although BMA provides a very good speed of convergence towards the optimal

model parameters, there are some points of the algorithm where the performance could be increased. The *Bacterial Memetic Algorithm with Memetic Mutation* [6] exploits the *Levenberg-Marquardt method* more efficiently. Instead of applying the LM cycle *after* the bacterial mutation as a separate step, the modified algorithm executes several LM cycles during the bacterial mutation after each mutational step.

The bacterial mutation operation changes one or more parameters of the modeled system randomly, then it checks whether the model obtained in this way performs better than the previous models or the models that have been changed concurrently this way in the other clones. The mutation test cycle is repeated until all the parameters of the model have gone through the bacterial mutation.

In the mutational cycle, it is possible to gain a temporary model that has an instantaneous fitness value that is worse than the one in the previous or the concurrent models. However, it is potentially better than those, because it is located in a region of the search space that has a better local optimum than the other models do. In accordance with this, if some *Levenberg-Marquardt* iterations are executed after each bacterial mutational step, the test step is able to choose some potentially valued clones that could be lost otherwise.

In the *Bacterial Memetic Algorithm with Memetic Mutation*, after *each mutational step* of *every single bacterial mutation iteration* several LM iterations are done (*memetic mutation*). Several tests have shown it is enough to run just 3 to 5 of LM iterations per mutation in order to improve the performance of the whole algorithm. The usual test phase of the bacterial mutation operation follows after the LM iterations. After the complete *modified* (*memetic*) *bacterial mutation* follows the LM method that is used in the original BMA, where more, e.g. 10 iterational steps, are done with all the individuals of the population towards reaching the local optimum. After all this, the gene transfer operation is executed if needed.

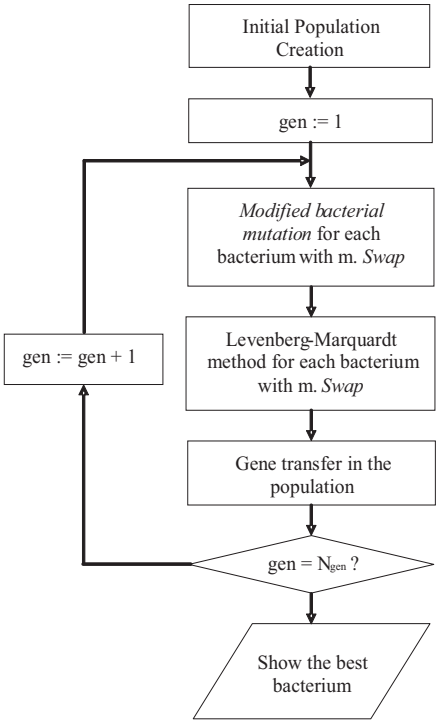

Fig. 3. Flowchart of the MBMA

The advantages of the IBMA and BMAM algorithms are combined in the *Modified Bacterial Memetic Algorithm* (MBMA) [6]. The original bacterial memetic algorithm is modified in the knot order violation handling (affecting the LM method incorporated in the BMA), and in the revised operator execution order (*memetic mutation*). The simulation results [6] proved that this method is superior to the IBMA and BMA algorithms. The MBMA flowchart can be seen in Fig. 3.

We observed that the model convergence speed depends not only on the complexity of the fuzzy rule base, but varies in different phases of the optimization process. For example, we measured less convergence speed in the first 10% of the optimization process, case

of a 2 dimensional test function, and in the 20–30% of the optimization process, case of a 6 dimensional test function. In the various algorithms, the time measured between two generations (one iteration length) is very different. The BEA is the fastest (the cycle includes bacterial mutation and gene transfer), followed by the IBMA (applying the bacterial mutation, LM cycles and gene transfer). A single iteration of the MBMA requires the longest time, including in each cycle modified bacterial mutation (with LM iterations), LM method and gene transfer. Comparing the BEA – IBMA (BMA) resp. the IBMA – MBMA methods we concluded that BEA shows the highest convergence speed in the initial phase of the optimization process.

2.6. Progressive Bacterial Algorithm (PBA)

In [10] we proposed a novel algorithm, the *Progressive Bacterial Algorithm* (PBA), which combines the improvements of BEA, IBMA and MBMA algorithms. In case of unknown applications, the main questions are: What should be the proper sequence of these methods? How can they be combined to obtain an overall high quality model of the optimization process with a very good convergence speed? The main goal is to obtain low model error values during the whole optimization process [1]. We proposed to use BEA in the early stage of the optimization, followed by IBMA, and using MBMA as the last step. In order to determine when to change from one algorithm to another, we concurrently apply two algorithms for different individuals in the population. The individuals were monitored in terms of time. For all those, whom the simplest algorithm provides no more better model errors we switch the individuals training algorithm to the other one. In this way, in each stage of the optimization process, favorable overall MSE values and training characteristics can be obtained.

The PBA flowchart can be seen in Fig. 4. The different algorithms we denoted by: M1 = BEA, M2 = IBMA and M3 = MBMA. The first step is to create the initial population, after that, we apply a different learning algorithm for each individual, for example: at the beginning of the optimization process for the main part of the population, we choose the method M1, while for the rest (for example: 1 + int(NPopulation / 5)), we choose M2. Our proposed approach starts in parallel with two versions, and we chose the best one by calculating and comparing the models MSE value. In order to have comparable MSE data, we have to ensure the same optimization time for each individual. Their learning times are measured and stored for each bacterium along with the current training method. In this novel approach, regarding to one PBA generation, any version (individual's) generation iteration time corresponds to the more complicated method's generation iteration time. The following step of the PBA is to create the individual's next generation. First, we try for a number of individuals the 'complicated' IBMA (M2 iterations without gene transfer). Then the BEA method (M1 iterations without gene transfer) is applied for a few times for the corresponding remainder individual's, while each individual's total optimization time (TM1) is less than a medium total time per individual (TM2) reached with the M2 algorithm.

Then the gene transfer operation within method groups is made (and if a certain termination criterion is satisfied the optimization process is over). During the gene transfer operation a model parameter sequence is transferred from the better individuals into individuals with less fitness values. For them, in many cases, the infections result in temporary high MSE values. That is the reason why the gene transfer operation, in case of sufficient number of individuals, is applied just among individuals which belongs to the same current training method.

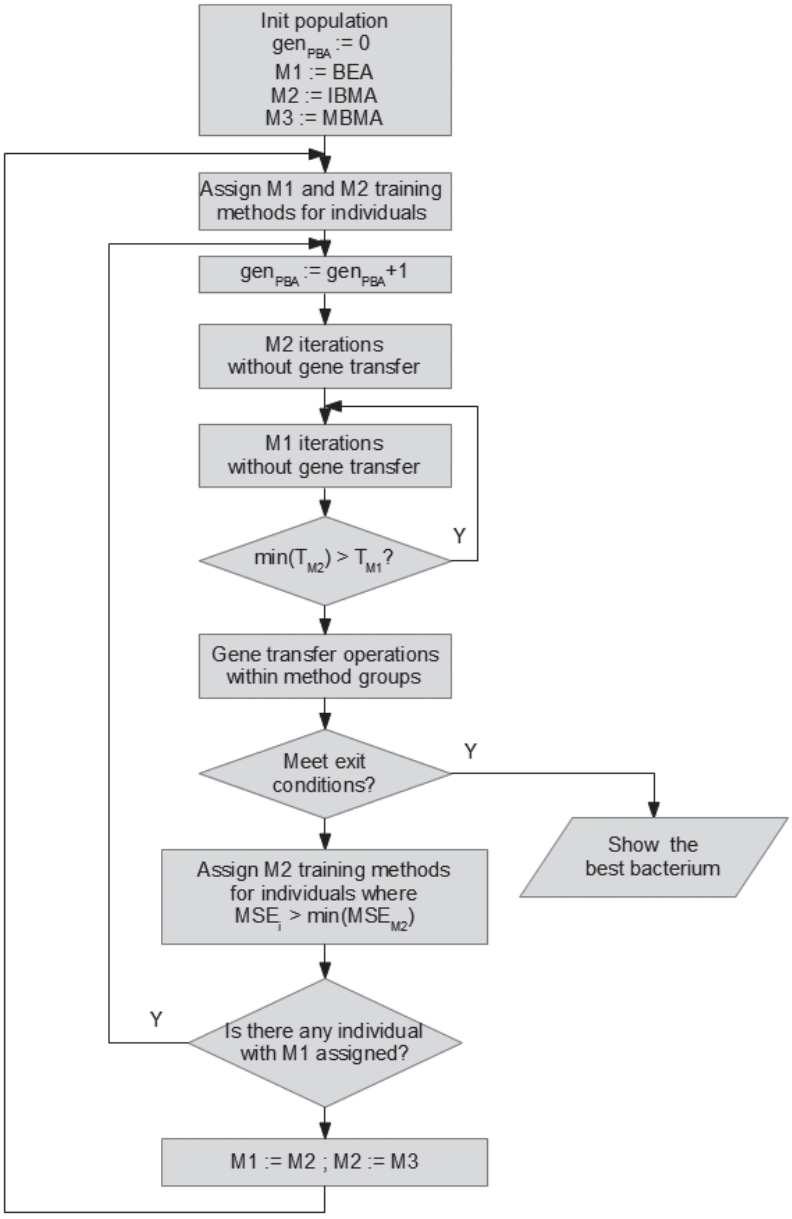

Fig. 4. Flowchart of the PBA

If we have in the population, for example four individuals trained with BEA and one trained with IBMA: application of the gene transfer operation to individuals trained with various methods, it probably happens that the gene transfer (which causes repeated temporary deterioration) will always affect that one trained with IBMA method, thus reducing the model efficiency.

When the minimum MSE value of individuals trained by M2 is higher than the corresponding maximum MSE value of individuals trained by M1, the procedure is repeated from the M2 – M1 training steps until a stopping criterion is satisfied (e.g. maximum number of generations). For individuals where the inequality MSEi < min(MSEM2) is not satisfied, the M2 method is assigned. After all, the population's individuals are switched to M2 algorithm, we assign the method M2 to M1, furthermore, method M3 to M2, and then we start from the beginning, with reassigning methods M1 and M2 for individuals (this method can be extended for more than three suitable training algorithms). In contrast to the former algorithms (e.g. BEA, IBMA, MBMA), in this new algorithm (PBA), the course of the simulation is not replicable due to the granularity in the processing time measurement. Not even for the same initial conditions and pseudo-random number sequences.

2.7. Experimental results for PBA performance evaluation

We investigated the performance properties of the PBA and predecessor algorithms, involving the model accuracy and convergence speed. During the simulations, we examined the relationship between simulation time and model accuracy while training the model parameters of a (fuzzy rule based) Mamdani-type inference system [13] with various bacterial type training algorithms. Two test functions were used, as described below:

a) Test function with 2 input variables (2iv), 200 samples:

$$f_1(\underline{x}) = \sin^5(0.5 \cdot x_1) \cdot \cos(0.7 \cdot x_2) \quad \text{where} \quad x_1 \in [0...3\pi],\ x_2 \in [0...3\pi] \tag{1}$$

b) Test function with 6 input variables (6iv) [17]; 500 samples:

$$f_2(\underline{x}) = x_1 + x_2^{0.5} + x_3 \cdot x_4 + 2 \cdot e^{2 \cdot (x_5 - x_6)} \quad \text{where} \quad \begin{array}{l} x_1 \in [1...5],\ x_2 \in [1...5],\ x_3 \in [0...4], \\ x_4 \in [0...0.6],\ x_5 \in [0...1],\ x_6 \in [0...1.2] \end{array} \tag{2}$$

The algorithms parameters are: fuzzy rules: 5; population size: 7; clones: 7; gene transfers per generation: 3; LM iterations per memetic bacterial mutation: 5; Mamdani inference system aggregation operator: min.

Fig. 5 and 6 present some graphs of typical simulation results with a less complex test function (2 dimensional 2 iv) and a more complex test function (6 dimensional 6 iv) comparing the characteristics for BEA, IBMA, MBMA and PBA algorithms [10]. The graphs show the relationship between the simulation time (horizontal axis) and the model MSE values (vertical axis). Comparing the simulation results, in the case of both test functions, it is shown that the PBA method provides favorable behavior: high model accuracy in the final stage of the optimization and low model error compared to the other algorithms from the first stage. However, in the initial phase of the optimization, the BEA still provides higher convergence speed. The BEA is fast because the main steps of the algorithm are the bacterial mutation and the gene transfer operation. In the PBA, gene transfer operations are not applied until every single (M1) individual's processing time exceeds the method M2 (IBMA) iteration time. This could be the reason for the higher initial convergence speed of BEA. During the optimization process, the BEA is followed by the PBA, IBMA and finally by the MBMA algorithms. In the middle and in the final phase of the process, when the BEA efficiency decreased, the PBA algorithm MSE values are the lowest, highlighting the method's main advantage.

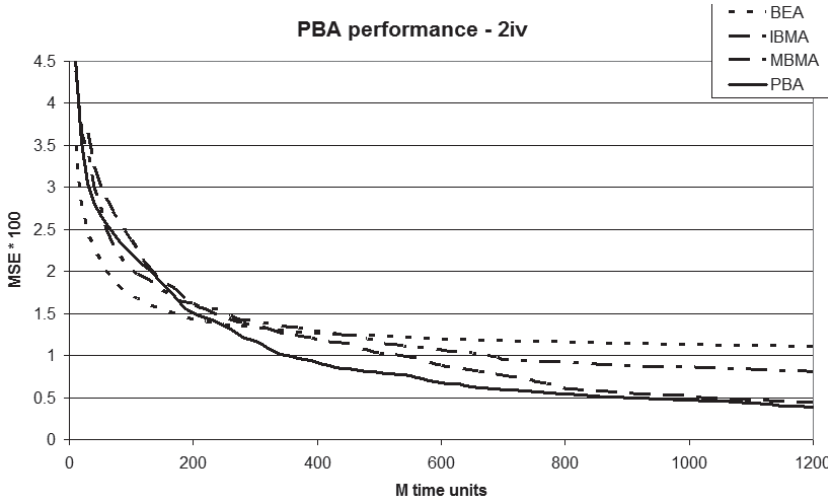

Fig. 5. Typical simulation results – less complex (two dimensional) test function; BEA, IBMA, MBMA, PBA

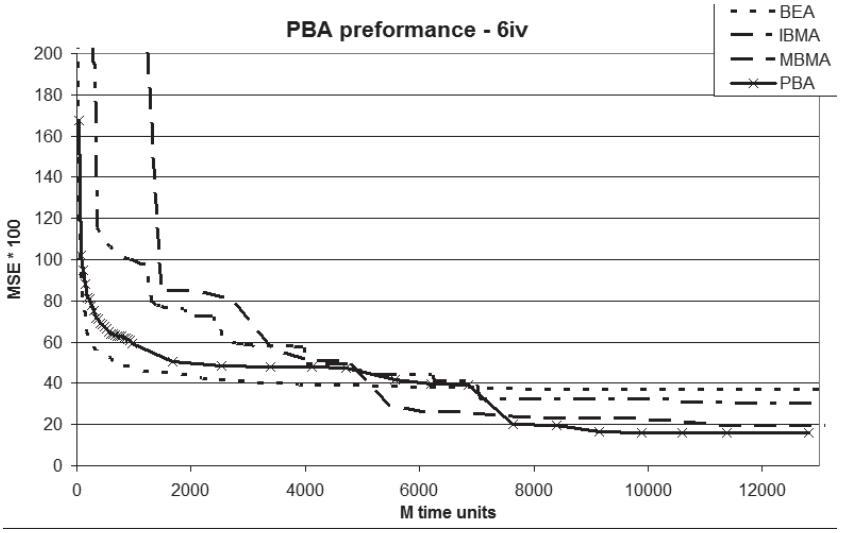

Fig. 6. Typical simulation results – complex (six dimensional) test function; BEA, IBMA, MBMA, PBA

3. Conclusions

In this paper, we revisited various methods to improve the bacterial algorithms performance used for fuzzy rule base extraction. A few evolutionary approaches have been proposed for fuzzy rule base extraction from input-output data such as the *Pseudo-Bacterial Genetic*

Algorithm, the *Bacterial Evolutionary Algorithm*, and various bacterial memetic algorithms developed by us. All these methods have turned out to be helpful with the construction of fuzzy rule based models. Finally, we proposed a special combination of bacterial evolutionary and memetic algorithms. The novel *Progressive Bacterial Algorithm's* (PBA) speed of convergence is comparable with the listed approaches; nevertheless, it can be used as a useful tool by finding a good compromise between the model accuracy and the complexity.

This work is supported by Hungarian Scientific Research Fund (OTKA) K105529, K108405, TÁMOP-4.2.2.A-11/1/KONV-2012-0012, TÁMOP-4.2.2.C-11/1/KONV-2012-0012.

References

[1] Balázs K., Kóczy L.T., *A Remark on Adaptive Scheduling of Optimization Algorithms*, IPMU 2, Vol. 81, 2010, 719-728.

[2] Botzheim J., Kóczy L.T., Ruano A.E., *Extension of the Levenberg-Marquardt algorithm for the extraction of trapezoidal and general piecewise linear fuzzy rules*, IEEE World Congress on Computational Intelligence, Honolulu 2002, 815-819.

[3] Botzheim J., Cabrita C., Kóczy L.T., Ruano A.E., *Fuzzy Rule Extraction by Bacterial Memtic Algorithm*, IFSA, Beijing 2005, 1563-1568.

[4] Gál L., Botzheim J., Kóczy L.T., Ruano A.E., *Fuzzy Rule Base Extraction by the Improved Bacterial Memetic Algorithm*, 6th international Symposium on Applied Machine Intelligence and Informatics, Herlany 2008, 49-53.

[5] Gál L., Botzheim J., Kóczy L.T., *Improvements to the Bacterial Memetic Algorithm used for Fuzzy Rule Base Extraction*, Computational Intelligence for Measurement Systems and Applications, CIMSA, Istanbul 2008, 38-43.

[6] Gál L., Botzheim J., Kóczy L.T., *Modified Bacterial Memetic Algorithm used for Fuzzy Rule Base Extraction*, 5th International Conference on Soft Computing as Transdisciplinary Science and Technology, CSTST, Paris 2008, 425-431.

[7] Gál L., Lovassy R., Kóczy L.T., *Function Approximation Performance of Fuzzy Neural Networks Based on Frequently Used Fuzzy Operations and a Pair of New Trigonometric Norms*, WCCI 2010 IEEE World Congress on Computational Intelligence, FUZZ-IEEE, Barcelona 2010, 1514-1521.

[8] Gál L., Kóczy L.T., Lovassy R., *Three Step Bacterial Memetic Algorithm*, Proc. of 14th IEEE International Conference on Intelligent Engineering Systems INES 2010, Las Palmas of Gran Canaria 2010, 31-36.

[9] Gál L., Kóczy L.T., *Fuzzy Rule Base Extraction by Bacterial Type Algorithms using selected T-norms*, Acta Technica Jaurinensis 4, Series Intelligentia Computatorica, Vol. 3, Györ 2011, 157-175.

[10] Gál L., Lovassy R., Kóczy L.T., *Progressive Bacterial Algorithm*, IEEE 13th International Symposium on Computational Intelligence and Informatics, CINTI, Budapest 2012, 317-322.

[11] Johanyák Zs.Cs., Tikk D., Kovács Sz., Wong K.W., *Fuzzy Rule Interpolation Matlab Toolbox – FRI Toolbox*, Proc. of the IEEE World Congress on Computational

Intelligence (WCCI'06), 15th Int. Conf. on Fuzzy Systems (FUZZ-IEEE'06), Vancouver, BC, Canada, Omnipress, 2006, 1427-1433.

[12] Mamdani E.H., *Application of fuzzy algorithms of simple dynamic plant*, Proceedings of the IEEE, 121, No. 12, 1974, 1585-1588.

[13] Mamdani E.H., Assilian S., *An experiment in linguistic synthesis with a fuzzy logic controller*, International Journal of Man-Machine Studies, Vol. 7, No. 1, 1975, 1-13.

[14] Marquardt D., *An Algorithm for Least-Squares Estimation of Nonlinear Parameters*, SIAM J. Appl. Math., 11, 1963, 431-441.

[15] Moscato P., *On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithm*, Technical Report Caltech Cocurrent Computational Program, Report.826, California Institute of Technology, Pasadena 1989.

[16] Nawa N.E., Hashiyama T., Furuhashi T., Uchikawa Y., *A Study on Fuzzy Rules Discovery Using Pseudo-Bacterial Genetic Algorithm with Adaptive Operator*, Proc. of IEEE Int. Conf. on Evolutionary Computation, ICEC'97, 1997, 589-593.

[17] Nawa N.E., Furuhashi T., *Fuzzy Systems Parameters Discovery by Bacterial Evolutionary Algorithms*, IEEE Transactions on Fuzzy Systems 7, 1999, 608-616.

TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE
AUTOMATIC CONTROL | AUTOMATYKA
4-AC/2013

SERGII TELENYK, OLEKSANDR ROLIK, MAKSYM BUKASOV,
DMYTRO HALUSHKO*

MODELS AND METHODS OF RESOURCE MANAGEMENT FOR VPS HOSTING

MODELE I METODY ZARZĄDZANIA ZASOBAMI DLA VPS HOSTING

A b s t r a c t

The paper summarizes models and methods of data center resource management for VPS hosting. The approach for the allocation of computing resources in the form of particles of a predetermined size (virtual nodes) was proposed. Different cases of this problem for both an excess and a lack of computing resources were considered. These problems belong to the classes of linear and nonlinear Boolean programming. To solve the mentioned problems, heuristic and guided genetic algorithms have been proposed. A comparison of their effectiveness was carried out.

Keywords: *virtualization, virtual private server, VPS, virtual node, resource management, guided genetic algorithm*

S t r e s z c z e n i e

Niniejszy artykuł podsumowuje modele i metody zarządzania zasobami centrum danych dla VPS hosting. Przedstawione w nim podejście dotyczy alokacji zasobów obliczeniowych w formie cząstek o określonym wymiarze (wirtualne węzły). Rozważono przy tym różne przypadki tego problemu, obejmujące zarówno nadmiar, jak i braki zasobów obliczeniowych. Problemy te należą do klas liniowego i nieliniowego programowania logicznego. Do ich rozwiązania wskazano odpowiednie heurystyki i nadzorowane algorytmy genetyczne. Artykuł podsumowano wnioskami na temat efektywności poszczególnych rozwiązań.

Słowa kluczowe: *wirtualizacja, prywatny serwer wirtualny, VPS, wirtualny węzeł, zarządzanie zasobami, nadzorowany algorytm genetyczny*

* Prof. D.Sc. Ph.D. Sergii Telenyk, e-mail: telenyk@acts.kpi.ua, Ph.D. Oleksander Rolik, Ph.D. Maksym Bukasov, M.Sc. Dmytro Halushko, Department of Automatic and Control in Technical Systems, Natyional Technical University of Ukarine.

Definitions

VPS – virtual private server
VN – virtual node
S_i – server, $i = 1, ..., n$
r_i – resources of S_i [VN]
V_j – VPS, $j = 1, ..., m$
p_j – required resources for V_j [VN]
p_{0j} – committed resources for V_j [VN]
x_{ij} – equals to 1 if VPS V_j is deployed on server S_i, otherwise equals to 0

1. Introduction

Recently, through the development and spread of virtualization and cloud computing technologies, there has been a trend to consolidate computational resources, data storage and communicational equipment in data centers. The implementation of globalization ideas in the IT-area has caused the development of corporative, national and global IT-infrastructures as organized complexes of interconnected networks, information technologies and resources, end-users' equipment and their environment (organized complexes of information applications, user applications and information services) [1]. Interaction between components of an IT-infrastructure provides support for the collection of information, its storage and processing. A special management system supports the effective functioning of the IT-infrastructure. Forrester analytics classified the problems of IT-infrastructure management system into 15 groups [2].

One of the most significant problems is resource management in data centers. Like the other problems in the information and communication services area, the mentioned problem is focused on the user's needs and ensures an appropriate level of service [3–5].

There are several approaches to solving the resource management problem in data centers. Creating a tool that is able to associate changes in the IT-infrastructure state with a degradation of quality of services and to take the appropriate actions is complicated because of the large number of users, the complexity of the IT-infrastructure, the variety of the equipment types and other factors [6–8]. In general, the problem of resource management is relevant for different types of networks and technologies. Researchers and engineers have developed a lot of generalized and specific methods to solve this problem [9–11]. Among these methods, there are a few that take into consideration the peculiarities of the data center allocation models, particularly in the case of the resources limitation [12–16].

However, the development of IT empowers companies to create data centers that take into account the business and user requirements more thoroughly. On the other hand, it is necessary to develop new models and methods for resource allocation in order to ensure the effective use of the new features. Recently, there has been a trend to transfer web applications from virtual hosting to virtual private servers (VPS), so that they would provide higher cost-effectiveness compared to dedicated servers, and could also ensure the necessary guaranteed number of computational resources, which are not always achieved via virtual hosting.

Until recently, the VPS services were provided by a not so flexible scheme – the customer signed an agreement under which s/he was granted the VPS with a fixed amount of the resources. If the customer wanted to ensure the quality of the functioning of applications during peak loads, s/he was forced to request resources with reserve and also pay for them during the idle time while the load is reduced.

Today, the VPS service providers usually allow customers to change their VPS options using the web management console. This enables customers to increase the VPS resources during periods of increased load to ensure the operational quality of their web applications, and to reduce the VPS resources during periods of the decreased load to save money. However, the major disadvantage of this approach is that it requires a customer to increase or decrease VPS resources in manual mode.

2. The problem

The IT-infrastructure resources (networks, servers, data storage, application etc.) require accounting and analysis of the compliance with user requirements to avoid customer outflow and financial losses. It is necessary to maintain the required level of information and communication services, including the peak load, because a lack of resources can cause a degradation of the service level. The solution of this problem by allocating additional resources is not always reasonable. It is necessary to create flexible solutions that are built on the load balancing and the resource allocation. This in turn requires appropriate mathematical models and methods to solve these problems.

Let's consider a situation with the VPS service provision when server resources are allocated by virtual nodes (VN), and are being accounted for in node-hours. While signing the contract for the provision of the services, a customer indicates a fixed number of VN that is guaranteed to be available at any time, and which will be paid for even during idle time. Also, let's assume a customer is able to specify the number of VNs to be additionally provisioned when necessary in case of the availability of appropriate resources in the cloud. A customer pays for those additional VNs only if they are used during peak loads. It is obvious that a customer is interested in obtaining additional resources while increasing his clients' request number to ensure the highest quality of service. A provider is also interested in providing additional resources to customers since they will have to pay more. Herewith, the provider guarantees to provide the resources to other customers, i.e. as additional resources may be used only those, which were not given to any other customer. It is necessary to develop cloud data centers, resource allocation and load models, and methods that meet the above features of the cloud IT-infrastructures. These models and methods are to be based on reasonable criteria for the providers and take into account resource, technological and other constraints.

The VPS providing service implies granting to a customer a virtual server in the form of a virtual machine (VM) that is actually running under hypervisor management on a single physical server [17]. In other words, a provider is unable to allocate to the VM the resources of different physical servers. Therefore, the main instrument of the resource allocation is the VM migration between servers to place the VMs most densely. To me, this phrase seems unclear, but ignore this comment if you think it would make sense to someone who understands the subject.

We suppose that there are several physical servers S_i, $i = 1, ..., n$, where VPS V_j, $j = 1, ..., m$ are running under hypervisor management.

It is clear that the following condition should be fulfilled for business models of the service provision mentioned above:

$$p_j \geq p_{oj}, j = 1, ..., m \tag{1}$$

If ISP does not offer an opportunity to request additional non-guaranteed computing resources, i.e.

$$p_j = p_{oj}, j = 1, ..., m \tag{2}$$

the problem is reduced to the problem that is described in [15] and can be solved by the proposed methods.

Let's impose the following constraints. Since each VPS can be located on only one server, the following condition should be fulfilled:

$$\sum_{i=1}^{n} x_{ij} = 1, \quad j = 1, ..., m \tag{3}$$

As a provider guarantees that each VPS will receive the resources not less than p_{oj}, the condition is:

$$\sum_{j=1}^{m} x_{ij} p_{0j} \leq r_i, \quad i = 1, ..., n \tag{4}$$

In order to meet the user requirements and to maximize their own profit in the best way possible, a provider can solve one of the three problems according to the availability of the resources.

Problem 1. If the data center resources substantially exceed the users' requirements to the resources, a provider will attempt to distribute VPS among the servers in the densest way to release some servers that could be turned off to save power. However, a provider will try to satisfy all of the customers' needs in additional resources because payment for even a single VN exceeds savings from shutting down of that server (Fig. 1).

Instead of condition (4) let's impose the following constraints:

$$\sum_{j=1}^{m} x_{ij} p_j \leq r_i, \quad i = 1, ..., n \tag{5}$$

Let's denote by e_i the power of server S_i, when it is not running any VPS. The indication that any VPS does not run on server S_i will express as follows:

$$d_i = \prod_{j=1}^{m} \overline{x_{ij}}, \quad i = 1, ..., n \tag{6}$$

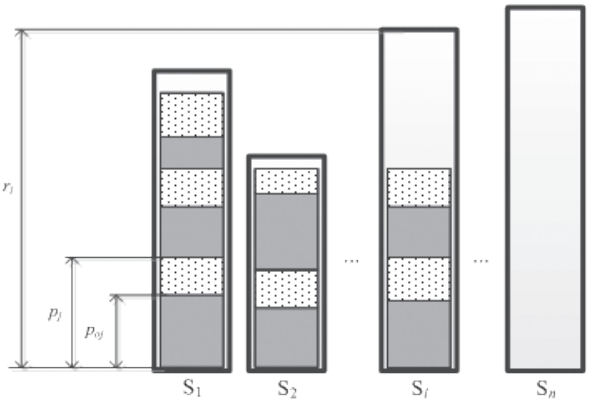

Fig. 1. Desired VPS distribution in the case of the resources excess

Then the problem of power consumption minimization can be formulated as follows:

$$\max \sum_{i=1}^{n} e_i \prod_{j=1}^{m} \overline{x_{ij}} \tag{7}$$

under the constraints (3) and (5). This problem is also similar to the problem that is described in [15] and can be solved by the proposed methods.

Problem 2. If the available resources do not significantly exceed the user requirements (problem 1 solution does not represent a solution that satisfies the requirements of (3) and (5)), a provider has to provide a guaranteed amount of VN for all the users, as well as meeting the maximum number of requests for additional resources in order to maximize own profit and customer satisfaction (Fig. 2). In order to meet this requirement, a provider has to place all of the VPS on the servers most tightly, but with less strict constraints: (4) instead of (5).

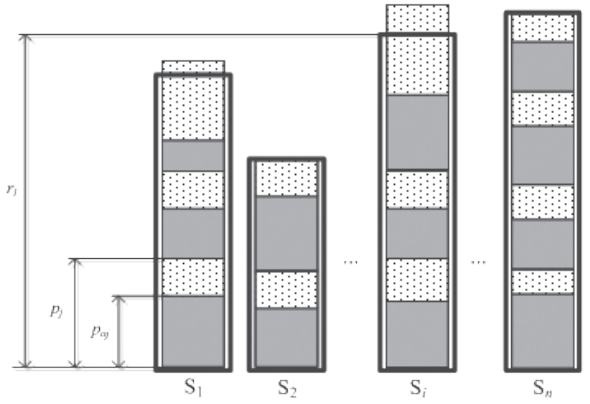

Fig. 2. Desired VPS distribution in the case of the resources lack

Then problem 2 can be formulated as follows:

$$\min \sum_{i=1}^{n} \left| r_i - \sum_{j=1}^{m} x_{ij} p_j \right| \tag{8}$$

under the constraints (3) and (4).

Problem 3. In the case of equipment failure, it becomes impossible to ensure all VPS with even the guaranteed amount of resources. The obvious solution is to support the most important services by providing the resources especially for VPS with related services at the expense of those which are less important. Since each VPS can be located on a single server, the following condition should be fulfilled:

$$\sum_{i=1}^{n} x_{ij} \le 1, \quad j = 1, ..., m \tag{9}$$

Let's denote by $w_j, j = 1,..., m$ the importance of services ensured by V_j.
Then the problem can be formulated as follows:

$$\max \sum_{j=1}^{m} \sum_{i=1}^{n} x_{ij} w_j \tag{10}$$

under the constraints (4) and (9).

3. Cloud IT-infrastructure resource allocation methods

The problems described above belong to a broad class of Boolean programming problems. Problems 1 and 3 are examples of the type of problem that is described in [15]. To solve problem 2 we use greedy and guided genetic algorithms, based on a new combination of ideas [18–21].

Greedy algorithm. Since we are interested in the most uniform distribution of VPS through the servers, let's formulate an idea of the algorithm as follows:

```
while (the list of unallocated VPS has at least one VPS)
{
    find VPS with the highest requirements to the resources;
    place that VPS on the least loaded server;
}
```

Using a greedy algorithm for the resource allocation between VPS is highly effective because the biggest VPS that requires a lot of the resources will be placed first. After that, all the free space on the servers will be filled by the smaller VPS.

Genetic algorithm (GA). Since each VPS can be placed on not more than a single server, for encoding genes let's move from $n*m$ matrix x_{ij} of the Boolean variables to the length m vector y_j of the discrete variables. Each element of that vector is the server's number $i = 1, ..., n$, which contains the appropriate VPS. For example:

$$x_{ij} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$y_j = [3 \quad 2 \quad 4 \quad 2 \quad 1 \quad 3 \quad 1 \quad 1]$$

This method of genes coding allows, firstly, to reduce the dimension of the problem, and secondly, to provide an automatic execution of the constraints (3). Therefore, the mutation operation will correspond to the VPS transferring from one server to another and the crossover operation, to the multiple VPS migrations between servers.

The fitness function is a very important concept in the genetic algorithms. It is a measure of an individual's fitness in the population. In our case, the fitness function has to take the large values for the most uniform VPS distribution on the servers when the entire major and most of the additional user resource requirements are satisfied. As the fitness function, we use the number of VNs which are allocated to the VPS. If all the basic and additional requirements for the resources were satisfied, i.e. the constraints (4) and (5), the fitness function will be equal to the number of VNs that were listed in the users' requests and were allocated for VPS.

If all the basic and only part of the additional users requirements were satisfied, i.e. all the constraints (4) are satisfied and (5) are not satisfied, the fitness function will be equal to the number of VNs that were actually allocated for VPS.

If the constraint (4) is not satisfied, the fitness function is penalized, as a result, it takes a small value and the corresponding individual becomes less attractive, reducing the likelihood of its participation in progeny formation.

This fitness function reflects the real economic situation, when a provider increases its income from the provision of a larger number of the VNs on a customers' order and responses in the case of failure to provide the guaranteed amount of resources.

To solve the problem, we use the guided genetic algorithm. The basic idea of this algorithm is to provide a balance between the 'research' and the 'use' by introducing a system of rules, based on which an operator (crossover (C) or mutation (M) for obtaining population of the next epoch is chosen [16]. These rules have to work in such a way that approach to the best solution is controlled If the approaching steps are quite large, it is necessary to speed up the search, and if they slow down, it is necessary to expand the useful schemas. At the same time, these rules should create a barrier for convergence to non-optimal solutions. Therefore, we dedicate every single epoch of the search process to the 'use' of the existing material (fixing optimal solutions) or to the 'research' of the new areas of the space solutions. The selection will happen in each case by keeping in the new epoch only the best individuals from a set of the previous epoch population, and by obtaining 'children' on its basis.

When the number of ‘children’ is too small, the genetic material collection will be filled in with an additional infusion of randomly generated individuals, this ensures the diversity of the population’s genetic material.

In order to form a guided GA system of rules, it is necessary to choose the process parameters for evaluation of convergence and necessity of the specific algorithm operator (C or M) for each particular epoch. Let’s introduce the following parameters.

Population growth rate. During the generation of the new epoch population by using one of the operators (C or M), the obtained number of ‘children’ could be very small in comparison to the population size. This fact indicates a degeneracy in the population due to the insufficient variety of genetic material and requires an appropriate action (an additional ‘infusion’ of new genetic material)

$$k = \frac{l}{N}$$

where:

l – the number of obtained ‘children’,
N – population size.

Population prospects. During the GA execution, it is necessary to have a parameter that would characterize the coming degree of the current population to the optimum.

As is known, the search algorithm stops when the maximum value of fitness function matches (or is close enough) to the mean value of fitness function of the entire population.

Let’s introduce the ‘population prospects’ index as a maximum value of the fitness function to its average value ratio for the current epoch population. We assume that the optimal criterion is a maximization of the fitness function value.

$$\rho = \frac{f_{\max}(y)}{f_{\text{avg}}(y)}$$

where:

$f(y)$ – objective function of search (fitness function),
$f_{avg}(y)$ – average value of the current population objective function.

As we can see from the formulated parameter $\rho \geq 1$. The prospects value equals 1 in the case when the fitness of all the individuals within the population is equal.

Convergence speed. It is necessary to have a parameter that would describe the variation trends of GA convergence during transition from epoch to epoch.

Let’s introduce the convergence speed concept as the difference between the prospect values of the previous $(i-1)$ and the current (i) epochs’ population:

$$\Delta_{i-1,i} = \rho_{i-1} - \rho_i$$

Let’s state a set of rules for the selection of the operation in order to obtain the next epoch population.

If the population growth rate is smaller than the threshold, then it is necessary to make an 'infusion' of the new genetic material (operator G).

The formalization of the rule is as follows:

$$\textbf{IF } (k \le k_0) \textbf{ AND } (N \le N_{\max}) \textbf{ THEN G}$$

where:

k_0 – limit for the value of the population growth rate,
$N_{\max}$ – constraint on the population size.

If the convergence speed becomes negative during the transition from $(i - 1)$ to (i) epoch, the crossover (C) operator will be used to form the next epoch population.

Assuming that during the transition from epoch to epoch the population fitness does not decrease (only the best individuals are being selected), convergence speed could be negative only if current epoch prospects become greater than in the previous one. This becomes possible when the new maximum value of the optimum (the best solution to the problem) is found. In order to save the better *best?* solution it is necessary to move from the 'research' to the 'use' strategy and, therefore, to the crossover.

The formalization of this rule is as follows:

$$\textbf{IF } (\Delta_{i-1,i} < 0) \textbf{ THEN C}$$

If GA convergence speed and value of the population prospects of the current epoch are not smaller than the threshold (nature of the convergence process is uncertain, there is no convergence to the 'local' optimum), then crossover should be used to obtain the next epoch population.

The formalization of the rule:

$$\textbf{IF } ((\rho_i \ge \rho_0) \textbf{ AND } (\Delta_{i-1,i} \ge \Delta_0)) \textbf{ THEN C}$$

where:

ρ_0 – limit value of the population prospects,
Δ_0 – limit value of the convergence speed.

If the convergence speed or the population prospects value of the current epoch are smaller than the corresponding limit values, then mutation should be used to obtain the next epoch population. In this case, the search for the optimal solution converges to a certain 'local' optimum and it is necessary to move from the 'use' to the 'research' strategy, and, therefore, to the mutation.

The formalization of the rule is as follows:

$$\textbf{IF } ((\rho_i < \rho_0) \textbf{ OR } (\Delta_{i-1,i} < \Delta_0)) \textbf{ THEN M}$$

As we can see from the proposed rules for the guided GA process management, it is necessary to set the limit values for the population growth rate, the convergence speed and

the prospects. By adjusting these factors, it is possible to regulate the GA convergence search speed and the nature of the basic processes. These processes take place at the stage of rules modification, which allows for the organizing of a kind of feedback in terms of the optimality of the obtained results.

4. Experimental results

The effectiveness of the proposed algorithms was estimated as follows. A cluster of 10 servers were divided into 16 VNs. Each of the proposed algorithms solved the problem of resource allocation for the cases of small and medium-sized VPS with respect to the server size. Two series of experiments were performed, which differed by the spread of the guaranteed number of VNs. In each experiment, the number of the additional VNs, which had been ordered by the users, ranged from 0 to half of the guaranteed amount of the VNs. The requirements for the virtual machines were chosen according to the following table:

Table 1

Input data for the experiment

Series of experiments 1 (small variation in the user requirements)				Series of experiments 2 (average variation in the user requirements)			
p_{0min}	p_{0max}	p_{min}	p_{max}	p_{0min}	p_{0max}	p_{min}	p_{max}
1	2	1	3	1	2	1	3
1	3	1	5	1	3	1	5
2	4	2	6	1	4	1	6
2	5	2	7	1	5	1	7
3	6	3	9	1	6	1	9

For each case, 20 samples were randomly generated. The average results of the solutions are presented in the Fig. 3. The guaranteed and desired average number of VNs was laid off as the x-axis respectively. The number of VNs that was successfully assigned to the VPS was laid off as the y-axis. The results of the heuristic algorithm are labeled as «E», the genetic one – as «GA», the series of the experiments are labeled with numbers 1 and 2.

As can be seen, if VPS has low resource requirements relative to the size of the servers, both algorithms give good results close to the possible maximum (16 VN*10 servers = = 160 VN on cluster). With an increase in the users' requirements (which has to be guaranteed by a provider) it becomes more difficult to place densely the larger VPS to servers and efficiency of the both algorithms decreases because the servers have unused VN. Therefore, the efficiency of GA is consistently higher than the heuristic algorithm.

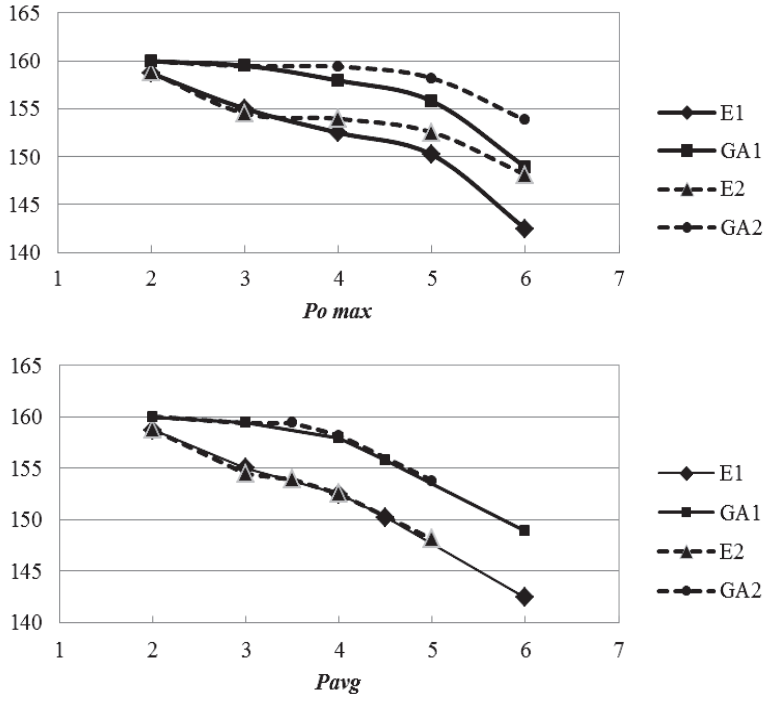

Fig. 3. Comparison of the effectiveness of the heuristic and genetic algorithms

5. Conclusions

Models and methods for solving the resource allocation problem in data centers that provide VPS services in cases where a customer is able to order services in pre-defined units (virtual nodes) were proposed.

Formulated problems were reduced to problems of Boolean programming. Heuristic and guided genetic algorithms [16] were used.

The results of the experiments confirmed the efficiency of the proposed approach as well as the appropriate level of time and costs for service providers.

References

[1] Bon J., Pieper M., Veen A., *Foundations of IT Service Management, based on ITIL*, Van Haren Publishing, 2005.

[2] Garbani J.-P., Mende T., *Market Overview: The IT Management Software Market In 2008*, Forrester Research, March 2008.

[3] *SLA Management Handbook. Volume 1. Executive Overview. GB 917.1. Version 2.1.*, TeleManagement Forum.Morrittown, NJ, January 2005.

[7] Cai Z., Chen Y., Kumar V., Milojicic D., Shwan K., *Automated Availability Management Driven by Business Policies*, 10th IFIP/IEEE Symposium on Integrated Management (IM), 2007, 264-273.

[8] Buco M.J., Chang R.N., Luan L.Z., Ward C., Wolf J.L., Yu P.S., *Utility computing SLA management based upon business objectives*, IBM Systems Journal, Vol. 43, No. 1, 2004, 159-178.

[9] Chen H., Huang L., Kumar S., Kuo C.C., *Radio resource management for multimedia QoS support in wireless network*, Kluwer Academic Publishers, Boston 2004.

[10] Herminghaus V., Scriba A., *Storage Management in Data Centers*, Springer 2009.

[11] *Information Storage and Management: Storing, Managing, and Protecting Digital Information*, EMC Education Services, John Wiley & Sons, 2010.

[12] Wolf J., Yu P., *On balancing load in a clustered web farm*, ACM Transactions on Internet Technology, Vol. 1, #2, 2001, 231-261.

[13] Ardagna D., Trubian M., Zhang L., *SLA based profit optimization in multi-tier web application systems*, Proc. of Int'l Conference On Service Oriented Computing, New York 2004, 173-182.

[14] Kimbrel T., Steinder M., Sviridenko M., Tantawi A., *Dynamic application placement under service and memory constraints*, Proc. of Int'l Workshop on Efficient and Experimental Algorithms, Santorini Island, May 2005, 1-12.

[15] Теленик С.Ф., Ролик О.І., Букасов М.М., Лабунський А.Ю., *Моделі управління віртуальними машинами при серверній віртуалізації*, Вісник НТУУ «КПІ»: Інформатика, управління та обчислювальна техніка, Київ 2009, № 51, 147-152.

[16] Теленик С.Ф., Ролік О.І., Букасов М.М., Андросов С.А., *Генетичні алгоритми вирішення задач управління ресурсами і навантаженням центрів оброблення даних*, 'Автоматика. Автоматизація. Електротехнічні комплекси та системи', 2010, № 1(25), 106-120.

[17] Marshall D., Reynolds W.A., McCrory D., *Advanced Server Virtualization: VMware and Microsoft Platforms in the Virtual Data Center*, Auerbach Publications, 2006.

[18] Turban E., Aronson J.E., *Decision support systems*, Prentice Hall, New Jersey 2001.

[19] Nong Y., *The Handbook of Data Mining*, Arizona State University Publishers, New Jersey 2003.

[20] Leung K.S., Duan Q.H., Xu Z.B., Wong C.K., *A new model of simulated evolutionary computation – convergence analysis and specifications*, IEEE Transactions on evolutionary computation,Vol. 5, #1, 2001, 3-16.

[21] Chambers D.L., *Practical handbook of genetic algorithms*, 2nd ed., Applications Vol. 1, Chapman & Hall, 2001.

TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE
AUTOMATIC CONTROL | AUTOMATYKA
4-AC/2013

PAWEŁ DRĄG, KRYSTYN STYCZEŃ*

THE INEXACT NEWTON BACKTRACKING METHOD AS A TOOL FOR SOLVING DIFFERENTIAL-ALGEBRAIC SYSTEMS

NIEDOKŁADNA METODA NEWTONA Z NAWROTAMI JAKO NARZĘDZIE DO ROZWIĄZYWANIA UKŁADÓW RÓŻNICZKOWO-ALGEBRAICZNYCH

Abstract

The classical inexact Newton method was presented as a tool for solving nonlinear differential-algebraic equations (DAEs) in a fully implicit form $F(y, y, t) = 0$. This is especially in chemical engineering where describing the DAE system in a different form can be difficult or even impossible to realize. The appropriate rewriting of the DAEs using the backward Euler method makes it possible to present the differential-algebraic system as a large-scale system of nonlinear equations. To solve the obtained system of nonlinear equations, the inexact Newton backtracking method was proposed. Because the convergence of the inexact Newton algorithm is strongly affected by the choice of the forcing terms, new variants of the inexact Newton method were presented and tested on the catalyst mixing problem.

Keywords: *differential-algebraic equations, systems of nonlinear equations, inexact Newton method*

Streszczenie

Klasyczna niedokładna metoda Newtona została przedstawiona jako narzędzie do rozwiązywania równań różniczkowo-algebraicznych zapisanych w formie niejawnej $F(y, y, t) = 0$. Zapisanie układu równań różniczkowo-algebraicznych w innej postaci w różnych zastosowaniach może być trudne lub niewskazane. Odpowiednie przekształcenie układów różniczkowo-algebraicznych z wykorzystaniem wstecznej metody Eulera umożliwia przedstawienie układu równań różniczkowo-algebraicznych jako układu równań nieliniowych dużej skali. W celu rozwiązania otrzymanego układu równań zaproponowano niedokładną metodę Newtona z nawrotami. Na zbieżność niedokładnej metody Newtona znacząco wpływa wybór czynnika wymuszającego. Nowe warianty niedokładnej metody Newtona zostały zastosowane do rozwiązania układu opisującego proces mieszania w obecności katalizatora.

Słowa kluczowe: *równania różniczkowo-algebraiczne, układy równań nieliniowych, niedokładna metoda Newtona*

* M.Sc. Paweł Drąg, e-mail: pawel.drag@pwr.wroc.pl, Prof. D.Sc. Ph.D. Krystyn Styczeń, Institute of Computer Engineering, Control and Robotics, Faculty of Electronics, Wrocław University of Technology.

Symbols

G, E, F	–	definition of functions,
h	–	length of the interval,
k	–	number of a current iteration,
n	–	number of mesh points,
p_1, p_2, p_3	–	parameters used in the new algorithm,
r	–	variable in the algorithm,
s	–	step of the algorithm,
t	–	independent variable, for example time,
u	–	control variable,
x	–	vector of decision variables,
y	–	differential variables,
z	–	algebraic variables,
$\beta, \varepsilon, \theta, \nu$	–	constants used in theorem,
$\alpha, \gamma, \vartheta, \omega$	–	parameters used in classical algorithms,
η	–	forcing term,
R	–	set of real numbers.

1. Introduction

Solving the differential-algebraic equations (DAEs) stands out as an important task in the scientific computing. Thus, methods for solving systems described by both differential and algebraic relations have gained increasing commercial importance over the last 30 years [19, 20]. With the increased academic research, potential applications of DAE systems have also been identified. Chemical engineering stands out as one of the most important application fields for these equations. In particular the control, modeling and identification of the chemical processes, which can take place in reactors, especially in the presence of a catalyst [3, 8].

These equations constitute the proper way to describe dynamic systems with slowly variable dynamics, as well as dynamic systems with conservation laws. The additional algebraic relations can model connections between the considered model and both the environment and the internal elements of the system. In this way, large-scale complex systems with dynamics and conservation laws can be designed and controlled [2].

In the article, attention was focused on the situation when the chemical process was described by the general differential-algebraic equations. It is important because in some cases, describing the system in a different form can be difficult or even impossible to realize [17].

The paper is constructed as follows. In the next section, the backward differentiation formula (BDF) is presented as an approach for discretization of the DAE systems. New aspects of the inexact Newton method were presented the in 3rd and 4th sections. The presented algorithms were tested on the catalyst mixing problem. The results are discussed in 5th section.

2. The backward Euler method

The backward differentiation formula was the first general technique for solving the differential-algebraic equations. The algorithm presented in [14] was the beginning of the specialized solvers for DAE systems.

The idea of this technique was that the derivative $\frac{dy(t)}{dt}$ can be approximated by a linear combination of the solution $y(t)$ at the current mesh point and at several previous mesh points.

At the beginning, the backward differentiation formula was defined for the systems of the differential equations coupled to algebraic equations. Then, this method was extended to apply any fully-implicit differential-algebraic systems:

$$G\left(\frac{dy(t)}{dt}, y(t), z(t), t\right) = 0 \tag{1}$$

where $G(\cdot)$ is the vector-valued DAE system.

The first order backward differentiation formula, known as the backward Euler method, is the simplest method for solving differential-algebraic systems. It consists of replacing the derivative in (1) by a backward difference:

$$E\left(\frac{y_n - y_{n-1}}{h}, y_n, z_n, t_n\right) = 0 \tag{2}$$

where $h = t_n - t_{n-1}$ and $E(\cdot)$ is a large-scale vector-valued system of nonlinear algebraic equations.

The simultaneous approach for solving algebraic systems is one of the main direction in modern optimization algorithms [3]. The system (2) can be solved by the Newton or inexact Newton methods [4]. Additionally, it was assumed, that $y(t_0)$ is known and t (for example time) is the independent variable. In the practical applications in chemical engineering, the length of the reactor can be used as the independent variable. If the time interval, in which the systems has to be considered is known, for numerical purposes, it can be scaled to the interval [0, 1].

3. The inexact Newton method

The approach given in the previous section and presented in eq. (2) can be expressed as:

$$F(x) = 0 \tag{3}$$

System of algebraic equations presented in eq. (3) is often found in the scientific computing and stands a common point of a lot of the real-life engineering problems. One was assumed, that a nonlinear mapping F: $\mathrm{R}^n \rightarrow \mathrm{R}^n$ has the following properties:

(i) there exists an $x^* \in \mathrm{R}$, for which $F(x^*) = 0$,
(ii) F is continuously differentiable in a neighborhood of x^*,
(iii) $\det F'(x^*) \neq 0$.

The Newton method is one of the best known methods for solving system of nonlinear algebraic equations (3). This method converges quadratically from any sufficiently good initial solution. It can be computationally very expensive, especially, when the size of the problem is very large. Then, in each iteration of the algorithm, the Newton equation (4) should be solved:

$$F(x_k) + F'(x_k)s_k = 0 \tag{4}$$

where x_k is a solution at a current iterate and $F'(x_k)$ denotes the Jacobian matrix of $F(x_k)$ at point x_k.

The solution of the Newton equation (4) computed as $s_k = (F'(x_k))^{-1}F(x_k)$ is the Newton step and denoted s_k^N. In this way, the next iterate can be obtained by:

$$x_{k+1} = x_k + s_k^N \tag{5}$$

The system (4) can be solved by the inexact Newton method. This inexact method is any method which for given an initial guess x_0, generates a sequence x_k as presented in the Algorithm 1.

Algorithm 1. The inexact Newton method [10]
Begin
1. Given $x_0 \in \mathrm{R}^n$
2. For $k = 1, 2, ...,$ until x_k convergence
2.1. Choose some $\eta_k \in [0, 1)$
2.2. Inexactly solve the Newton equation (4) and obtain a step s_k such that:

$$F(x_k) + F'(x_k)s_k \leq \eta_k F(x_k) \tag{6}$$

2.3. Let $x_{k+1} = x_k + s_k$.
End

One can see, that η_k in the Algorithm 1 is the forcing term in the k-th iteration. Depending on the forcing term, the inexact Newton step s_k, which satisfies the inexact Newton condition (6), should be obtained.

The role of the forcing terms is to control the degree of accuracy of solving the Newton equation (4). Therefore, at each iteration step of the inexact Newton method, a value of $\eta_k \in [0, 1)$ has to be chosen. Then, the inexact Newton step s_k can be obtained by solving the Newton equation approximately.

The inexact Newton condition (6) reflects two important features of the iteration process:
(i) reflects the reduction in the norm of the local linear model,
(ii) accuracy in solving the Newton equation (4).

Theorem 1 ([10]). Assume that F: $R^n \to R^n$ is continuously differentiable, $x^* \in R^n$ such that $F'(x^*)$ is nonsingular. Let $0 < \eta_{max} < \beta < 1$ be the given constants. If the forcing terms $\{\eta_k\}$ in the inexact Newton method satisfy $\eta_k < \eta_{max} < \beta < 1$ for all k, then there exists $\varepsilon > 0$, such that for any $x_0 \in N_\varepsilon(x^*) \equiv \{x: \|x - x^*\| < \varepsilon$, the sequence $\{x_k\}$ generated by the inexact Newton method is converged to x^*, and:

$$\|x_{k+1} - x^*\|_* \le \beta \|x_k - x^*\|_* \tag{7}$$

where $\|\upsilon\|_* \le \|F'(x^*)\upsilon\|_*$.

By Theorem 1 the inexact Newton method is locally convergent, if the forcing terms $\{\eta_k\}$ are uniformly less than 1. The convergence rate is stated by the following theorem.

Theorem 2 ([10]). Assume that F: $R^n \to R^n$ is continuously differentiable, $x^* \in R^n$ such that $F'(x^*)$ is nonsingular. If the sequence $\{x_k\}$ generated by the inexact Newton method is converged to x^*, then $\{x_k\}$ is converged to x^*, superlinearly if $\eta_k \to 0$, or quadratically if $\eta_k = 0(\|F(x_k)\|)$.

From Theorem 2 one can see, that the convergence rate of the Algorithm 1 is determined by the appropriate choice of the sequence of the forcing terms.

Since the initial point for the inexact Newton method cannot be guaranteed to be near a solution of the nonlinear system, the inexact Newton method was globalized by backtracking strategy [12]. Inexact Newton backtracking method was presented as Algorithm 2.

Algorithm 2. The inexact Newton backtracking method [12]
Begin
1. Given $x_0 \in R^n$, $\eta_{max} \in [0, 1)$, $\alpha \in (0, 1)$, and $0 < \theta_{min} < \theta_{max} < 1$
2. For $k = 0, 1, 2, ...,$ until $\{x_k\}$ convergence
2.1. Choose some $\tilde{\eta}_k \in [0, \eta_{max}]$
2.2. Inexactly solve Newton equation (4) and obtain a step $\tilde{s}_k$, such that:

$$F(x_k) + F'(x_k)\tilde{s}_k \le \tilde{\eta}_k F(x_k) \tag{8}$$

2.3. Backtracking loop:
2.3.1. Let $s_k = \tilde{s}_k$, $\eta_k = \tilde{\eta}_k$
2.3.2. While $\|F(x_k + s_k)\| > [1 - \alpha(1 - \eta_k)]\|F(x_k)\|$
(i) Choose $\theta \in [\theta_{min}, \theta_{max}]$
(ii) Update $s_k \leftarrow \theta s_k$ and $\eta_k \leftarrow 1 - \theta(1 - \eta_k)$
2.4. Let $x_{k+1} = x_k + s_k$
End

In each iteration of the inexact Newton backtracking method, the backtracking loop along $\tilde{s}_k$ is implemented until the condition:

$$\|F(x_k + s_k)\| > [1 - \alpha(1 - \eta_k)]\|F(x_k)\| \tag{9}$$

is satisfied [12]. Equation (9) has been called the sufficient decrease condition. It has been used to guarantee that $||F(x_{k+1})||$ has a certain decrease in each iteration. In practical applications a positive integer has been given in advance to control the maximal backtracking loop number along $\tilde{s}_k$.

4. A choice of forcing terms

There are some strategies to determine preferable and effective sequences for forcing terms. Four of them were selected and presented. The first strategy was proposed in [11].

1st Given $\eta_0 \in [0, 1)$, then:

$$\eta_k = \min\left\{\frac{1}{k+2}, F\left(x_k\right)\right\} \tag{10}$$

Two others strategies presented in [13] can be expressed as.

2nd For given $\eta_0 \in [0, 1)$, then:

$$\eta_k = \begin{cases} \vartheta_k, & \eta_{k-1}^{(1+\sqrt{5})/2} \le 0.1, \\ \max\left\{\vartheta_k, \eta_{k-1}^{(1+\sqrt{5})/2}\right\}, & \eta_{k-1}^{(1+\sqrt{5})/2} > 0.1, \end{cases} \tag{11}$$

where $\vartheta_k = \dfrac{F\left(x_k\right) - F\left(x_{k-1}\right) - F'\left(x_{k-1}\right)s_{k-1}}{F\left(x_{k-1}\right)}$, $k = 1, 2, \ldots$

3rd Given $\gamma \in (0, 1]$, $\omega \in (1, 2]$, $\eta_0 \in [0, 1)$, choose:

$$\eta_k = \begin{cases} \gamma\left(\dfrac{F\left(x_k\right)}{F\left(x_{k-1}\right)}\right)^{\omega}, & \gamma\left(\eta_{k-1}\right)^{\omega} \le 0.1, \\ \max\left\{\gamma\left(\dfrac{F\left(x_k\right)}{F\left(x_{k-1}\right)}\right)^{\omega}, \gamma\left(\eta_{k-1}\right)^{\omega}\right\}, & \gamma\left(\eta_{k-1}\right)^{\omega} > 0.1, \end{cases} \tag{12}$$

4th The last presented strategy was introduced in [1]. Let us denote:

$$\mathrm{Ared}_k(s_k) = ||F(x_k)|| - ||F(x_k + s_k)|| \tag{13}$$

$$\mathrm{Pred}_k(s_k) = ||F(x_k)|| - ||F(x_k) + F'(x_k)s_k) \tag{14}$$

then

$$r_k = \frac{\mathrm{Ared}_k\left(s_k\right)}{\mathrm{Pred}_k\left(s_k\right)} \tag{15}$$

In the 4th approach, the forcing term η_k is adjusted depending on the value of r_k. One can distinguish four situations which might take a place in practical applications. If $r_k \approx 1$, then the local linear model and nonlinear model will agree well on their scale and $\|F(x)\|$ will usually be reduced. If r_k nears 0, but $r_k > 0$, then the local linear model and nonlinear model disagree and $\|F(x)\|$ can be reduced very little. If $r_k < 0$, then the local linear model and nonlinear model disagree and $\|F(x)\|$ will be enlarged. In the last situation, if $r_k >> 0$, then the local linear model and nonlinear model also disagree, but $\|F(x)\|$ will be reduced greatly. If $\mathrm{Pred}_k(s_k) = 0$, then the solution is achieved and computations are stopped. One can choose forcing terms according to the value of r_k:

$$\eta_k = \begin{cases} 1-2p_1, & r_{k-1} < p_1, \\ \eta_{k-1}, & p_1 < r_{k-1} < p_2, \\ 0.8\eta_{k-1}, & p_2 < r_{k-1} < p_3, \\ 0.5\eta_{k-1}, & r_{k-1} \geq p_3, \end{cases} \tag{16}$$

where $0 < p_1 < p_2 < p_3 < 1$ are prescribed at first and $p_1 \in \left(0, \frac{1}{2}\right)$.

The choice of forcing terms is to determine η_k by the magnitude r_{k-1}.

The 2nd strategy reflects the agreement between $F(x)$ and its local linear model at the previous step. The 3rd choice reflects the reduction rate of $\|F(x)\|$ from x_{k-1} to x_k.

5. Case study: Catalyst mixing problem

The main goal is to determine the optimal mixing policy of two catalysts along the length of a tubular reactor [15]. The mixing ratio of the catalysts represents the control variable. The formulation of the dynamic optimization problem was described by the system of differential-algebraic equations:

$$\max_u z_3(1.0) \tag{17}$$

subject to:

$$\dot{y}_1 - u\left(10y_2 - y_1\right) = 0 \tag{18}$$

$$\dot{y}_2 - u\left(y_1 - 10y_2\right) + \left(1-u\right)y_2 = 0 \tag{19}$$

$$z_3 + y_1 + y_2 - 1 = 0 \tag{20}$$

$$x(0) = [y_1(0), y_2(0), z_3(0)]^T = [1.0; 0.0; 0.0]^T \tag{21}$$

$$u(t) \in [0.0; 1.0] \tag{22}$$

There are some reasons to describe the process using the DAE model. Depending on the application, it may be difficult to reformulate the problem as an ODE especially, when nonlinearities are present. The algebraic equations typically describe conservation laws or explicit equality constraints and they should be kept invariant. Furthermore, it is easier to vary design parameters in an implicit model. In the presented application, the main advantage of a such formulation is that the implicit model does not require the modeling simplifications often necessary to get an ODE and the variables keep their original physical interpretation.

In our research we concentrated upon how to solve the DAE system (18)-(20) with known feasible initial conditions (21) and assumed control function $u(t) = t$. The DAE system was considered in the time domain $t \in [0.0; 1.0]$. Then, the equations were discretized into equidistant points with a distance of 0.001. It resulted in 2000 differential and 1000 algebraic state variables. Afterwards, 3000 equality constraints from the backward Euler method were imposed. The Jacobian matrix was obtained analytically and stored as a 1000×1000 sparse matrix. This large-scale system of the algebraic equations was solved using GMRES algorithm [5, 18]. The inexact Newton method was used with the forcing terms adjusted by four presented approaches.

The simulations were executed in the Matlab environment using Wroclaw Centre for Networking and Supercomputing for a vector of initial conditions.

Let us denote the initial conditions vector as follows:

$$x(0) = [x_{1,1}, ..., x_{1,1000}, x_{2,1}, ..., x_{2,1000}, x_{3,1}, ..., x_{3,1000}]^T \tag{23}$$

In this case the vector $x(0) = [1.0, ..., 1.0, 0.0, ..., 0.0, 0.0, ..., 0.0]^T$, and $||F(x_0)|| = 0.0258$. The initial conditions like presented in eq. (23) are feasible for the discretized DAE model only at the mesh points.

The simulations were executed with the parameters $\gamma = 0.5$, $\omega = 1.5$, for proposition 3 and $p_1 = 0.25$, $p_2 = 0.6$ and $p_3 = 0.8$ for the 4th proposition.

The results after 6 iterations for proposition 1 and initial conditions as presented in (23) were plotted on Fig.1.

Results presented in Table 1 indicate, that the solution obtained by the inexact Newton algorithm with forcing terms adjusted as in proposition 1, have a high accuracy in 6 iterations. A similar situation can be observed in Table 2 for proposition 2, but the convergence is slower. The forcing terms adjusted as in 4th proposition give progress, which is slower than the previous two, but comparable with results for the 3rd proposition.

Using the Modellica DASSL solver, the derivatives $\dot{y}(t)$ are approximated by backward differentiation formulae (BDF), and the resulting nonlinear system at each time-step is solved by the Newton method. The implemented in Matlab procedure ode15s, utilizes the same algorithms as Modellica. These approaches are definitely different to the one presented in our researches. They belongs to the sequential methods. In the article, a simultaneous approach to solving differential-algebraic systems was investigated. In solving the catalyst mixing problem described as the DAE model, the differences are unnoticeable.

Table 1

Results for proposition 1

Iteration k	Proposition 1	
	η_k	$\|\|F(x)\|\|$
1	0.0258	0.0012
2	0.0012	6.7233e-4
3	6.7233e-4	2.0389e-4
4	2.0389e-4	2.2336e-5
5	2.2336e-5	1.1930e-8
6	1.1930e-8	2.4685e-13

Table 2

Results for propositions 2 and 3

Iteration k	Proposition 2		Proposition 3	
	η_k	$\|\|F(x)\|\|$	η_k	$\|\|F(x)\|\|$
1	0.9000	0.0039	0.9000	0.0256
2	0.8433	0.0033	0.4917	0.0253
3	0.7589	0.0025	0.4917	0.0250
4	0.6400	0.0016	0.4917	0.0247
5	0.4857	7.6601e-4	0.4917	0.0244
6	0.3108	2.3772e-4	0.4917	0.0242
7	0.1510	3.5672e-5	0.4917	0.0239
8	1.2772e-11	6.6590e-9	0.4917	0.0236

Table 3

Results for proposition 4

Iteration k	Proposition 4		
	η_k	r_k	$\|\|F(x)\|\|$
1	0.5000	0.0145	0.0256
2	0.4000	0.0145	0.0253
3	0.4000	0.0145	0.0250
4	0.4000	0.0145	0.0247
5	0.4000	0.0145	0.0244
6	0.4000	0.0145	0.0242
7	0.4000	0.0145	0.0239
8	0.4000	0.0145	0.0236

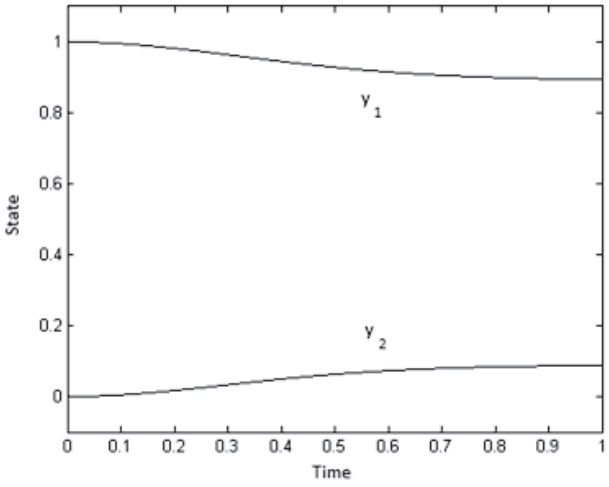

Fig. 1. Trajectory of the differential variables y_1 and y_2. Results for the forcing terms as in as proposition 1 after 6 iterations

6. Conclusion

In the article, new aspects of the inexact Newton methods for solving differential-algebraic models of the chemical processes in the fully implicit form were considered. The methods for the choice of the forcing terms for the inexact Newton method were presented and tested on the catalyst mixing problem.

The presented approach has some specific advantages which we want to emphasize. First of all, description a technological process using differential-algebraic equations enables us to exploit the system structure by problem-specific solvers. Hence, the implicit model does not require the modeling simplifications and the variables keep their original physical interpretation.

Rewriting the differential-algebraic model as a large-scale system of algebraic equations has a positive impact on the stability of the obtained solution. Recursive solutions do not have this property. This is the first step to take advantage of the effective numerical procedures for large and sparse matrices [3].

In the computations, matrix inversion was efficiently avoided.

Solving the large scale nonlinear differential-algebraic systems, especially with unbalanced nonlinearities, is a challenge. Attempts to solve these equations are still made [6]. One of the most famous examples of this type is the model of a kinetic batch reactor [7, 9].

For solving the Newton equation, the GMRES algorithm was used. The largest difficulty is associated with the calculation of the matrix-vector product. This calculation is time consuming and need storage of the matrix, which often has thousands of both rows and columns.

The future work will be concentrating on new reliable Jacobian-free Newton-Krylov methods, which could be successfully applied in optimal control of the chemical processes modeled by the nonlinear differential-algebraic equations.

This work was supported by the grant 'Młoda Kadra' B30036/I6 at Wroclaw University of Technology. The authors would like to thank all the reviewers for their helpful comments on an earlier version of the manuscript.

Referencess

[1] An H.-B., Mo Z.-Y., Liu X.-P., *A choice of forcing terms in inexact Newton method*, Journal of Computational and Applied Mathematics, Vol. 200, 2007, 47-60.

[2] Betts J.T., *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming. Second edition*, SIAM, Philadelphia 2010.

[3] Biegler L.T., *Nonlinear Programming. Concepts, Algorithms, and Applications to Chemical Processes*, SIAM, Philadelphia 2010.

[4] Brenan K.E., Campbell S.L., Petzold L.R., *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, SIAM, Philadelphia 1996.

[5] Brown P.N., Hindmarsh A.C., Petzold L.R., *Using Krylov methods in the solution of large-scale differential-algebraic systems*, SIAM Journal on Scientific Computing, Vol. 15, 1994, 1467-1488.

[6] Cai X.-C., Keyes D.E., *Nonlinearly Preconditioned Inexact Newton Algorithms*, SIAM Journal on Scientific Computing, Vol. 24, 2002, 183-200.

[7] Caracotsis M., Stewart W.E., *Sensitivity analysis of Initial Value Problems with mixed ODEs and algebraic equation*, Computers and Chemical Engineering, Vol. 9, 1985, 359-365.

[8] Drąg P., Styczeń K., *A Two-Step Approach for Optimal Control of Kinetic Batch Reactor with electroneutrality condition*, Przegląd Elektrotechniczny (Electrical Review), Vol. 6, 2012, 176-180.

[9] Drąg P., Styczeń K., *Inexact Newton method as a tool for solving differential-algebraic system*, Proceedings of the 2013 Federated Conference on Computer Science and Information Systems (FedCSIS), 2013, 639-642.

[10] Dembo R.S., Eisenstat S.C., Steihaug T., *Inexact Newton Methods*, SIAM Journal on Numerical Analysis, Vol. 19, 1982, 400-408.

[11] Dembo R.S., Steihaug T., *Truncated-Newton algorithm for large-scale unconstrained optimization*, Mathematical Programming, Vol. 26, 1983, 190-212.

[12] Eisenstat S.C., Walker H.F., *Globally convergent inexact Newton methods*, SIAM Journal on Optimization, Vol. 4, 1994, 393-422.

[13] Eisenstat S.C., Walker H.F., *Choosing the forcing terms in an inexact Newton method*, SIAM Journal on Scientific Computing, Vol. 17, 1996, 16-32.

[14] Gear C.W., *The simultaneous numerical solution of differential-algebraic equations*, IEEE Transactions on Circuit Theory, Vol. 18, 1971, 89-95.

[15] Huang Y.J., Reklaitis G.V., Venkatasubramanian V., *Model decomposition based model for solving general dynamic optimization problems*, Computers and Chemical, Vol. 26, 2002, 863-873.

[16] Knoll D.A., Keyes D.E., *Jacobian-free Newton-Krylov methods: a survey of approaches and applications*, Journal of Computational Physics, Vol. 193, 2004, 357-397.

[17] Petzold L., *Differential/Algebraic Equations are not ODEs*, SIAM Journal on Scientific Computing, Vol. 3, 1982, 367-384.

[18] Saad Y., Schultz M.H., *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput., Vol. 7, 1986, 856-869.

[19] Vassiliadis V.S., Sargent R.W.H., Pantelides C.C., *Solution of a Class of Multistage Dynamic Optimization Problems. 1. Problems without Path Constraints*, Ind. Eng. Chem. Res., Vol. 33, 1994, 2111-2122.

[20] Vassiliadis V.S., Sargent R.W.H., Pantelides C.C., *Solution of a Class of Multistage Dynamic Optimization Problems. 2. Problems with Path Constraints*, Ind. Eng. Chem. Res., Vol. 33, 1994, 2123-2133.

TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE
AUTOMATIC CONTROL | AUTOMATYKA
4-AC/2013

ÁRON BALLAGI*, LÁSZLÓ T. KÓCZY**

ROBOT PATH PLANNING MAP PRE-OPTIMIZATION BY MULTILAYER FUZZY SITUATIONAL MAPS

WSTĘPNA OPTYMALIZACJA PLANOWNIA RUCHU ROBOTA ZA POMOCĄ WIELOWARSTWOWYCH ROZMYTYCH MAP SYTUACYJNYCH

Abstract

Intelligent robot path planning require have very complex decision-making and computational processes. Collecting and calculating a high amount of data is one of the weakest points of a such system. In addition, it is necessarily processed in real-time on a limited computational capacity. In this paper, we propose some novel algorithms for coping with these problems and give some information about Fuzzy Situational Maps and their use as a multidimensional extension of Fuzzy Signatures. An example takes to the field of path planning map pre-optimization by Fuzzy Situational Map.

Keywords: *fuzzy situational maps, mobile robotics*

Streszczenie

Zadania inteligentnego planowania ruchu robota obejmują bardzo złożone procesy decyzyjne oraz obliczeniowe. Agregacja i przetwarzanie dużej ilości danych są jednym z najsłabszych punktów tego typu systemów. Dodatkowo, przetwarzanie konieczne jest w czasie rzeczywistym przy ograniczonej mocy obliczeniowej. W artykule zaproponowano kilka nowych algorytmów, które radzą sobie z tymi problemami oraz przedstawiono informacje na temat rozmytych map sytuacyjnych oraz wielowarstwowych rozmytych map sytuacyjnych jako wielowymiarowego rozwinięcia rozmytych sygnatur. Przedstawiony przykład dotyczy metody wstępnej optymalizacji planowania ruchu robota z użyciem rozmytych map sytuacyjnych.

Słowa kluczowe: *rozmyte mapy sytuacyjne, roboty mobilne*

* M.Sc. Áron Ballagi, e-mail: aronball@gmail.com, Department of Automation, Széchenyi István University, Györ.

** Prof. D.Sc. Ph.D. László T. Kóczy, Department of Automation, Faculty of Engineering Sciences, Széchenyi István University Györ; Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics.

1. Introduction

Intelligent mobile robot path planning tasks are a challenging and very perspective research field. Optimal path planning in a mainly autonomous manner involves many difficult decision making processes and consumption of calculation. Our research is focused on a special case of map based robot path planning systems, where intelligent map pre-optimization takes place before the path planning process. One of the main problems in a map based path optimization system is the processing of very complex data structures, where the data are often noisy, distorted or simply absent (therefore unmeasurable).

In this paper, we propose a novel approach for describing and processing high-scale multidimensional structured data in a simple way. This method is called Fuzzy Situational Map (FSM), a multidimensional geometric extension of Fuzzy Signatures.

First, we give a short overview of Fuzzy Signatures as the 'ancestors' of Fuzzy Situational Maps. Then, the theory of FSM and Multilayer FSM will be presented. Finally, we will see an example of a path planning map pre-optimization task by Multilayer Fuzzy Situational Map.

2. Fuzzy Signatures

The original definition of fuzzy sets [1] was $A: X \rightarrow [0, 1]$, and was soon extended to *L-fuzzy sets* by Goguen [2].

This definition is $A_L: X \rightarrow L$, L being an arbitrary algebraic lattice. A practical special case, *Vector Valued Fuzzy Sets* was introduced in [3], where $A_{V,k}: X \rightarrow [0, 1]^k$, and the range of membership values was the lattice of k-dimensional vectors with components in the unit interval. A further generalization of this concept is the introduction of fuzzy signatures and signature sets, where each vector component is possibly another nested vector (Fig. 1).

3. Fuzzy situational maps

We propose a novel approach to support difficult decision-making and to depict situational or contextually dependent structured data. A special form of fuzzy signatures [4–7] with a spatial structure is used, namely, the Fuzzy Situational Map (FSM).

Fuzzy situational maps as multidimensional extended fuzzy signatures (FS) are suitable for describing complex multidimensional system conditions in cases where the information is fragmented, distorted or noisy [8–11].

The FSM can be two-, three- or even n-dimensional. Let us see the simplest case, two-dimensional fuzzy situational map (Fig. 2).

3.1. Two-dimensional Fuzzy Situational Map

Two-dimensional FSM may be considered as a geometric lattice, where each node has a fuzzy value or a whole fuzzy set, in extended case.

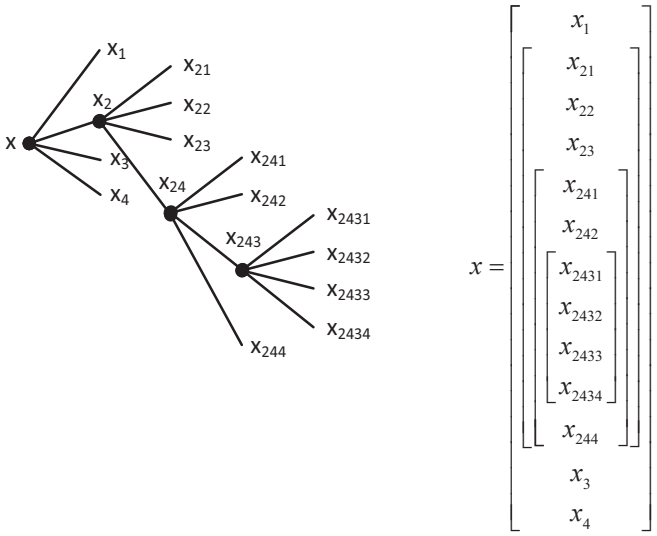

Fig. 1. Example of a Fuzzy Signature Structure

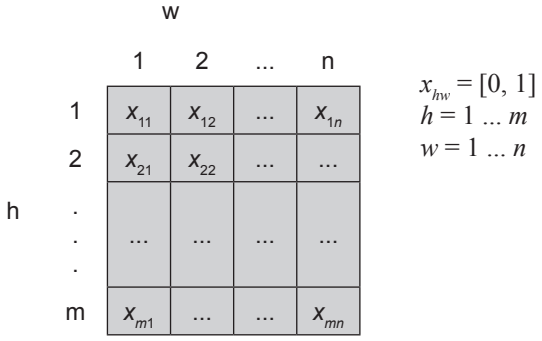

Fig. 2. A Fuzzy Situational Map

FSM can be represented as lattices or in matrix form as in Fig. 2 and in .

$$\text{FSM} = \begin{bmatrix} x_{011} & x_{012} & x_{013} & \begin{bmatrix} x_{111} & & x_{112} \\ & \begin{bmatrix} x_{211} & x_{212} & x_{213} \\ x_{221} & x_{222} & x_{223} \\ x_{231} & x_{232} & x_{233} \end{bmatrix} & \\ x_{121} & & \end{bmatrix} \\ x_{021} & x_{022} & x_{023} & x_{024} \\ x_{031} & x_{032} & x_{033} & x_{034} \\ x_{041} & x_{042} & x_{043} & x_{044} \end{bmatrix} \quad (1)$$

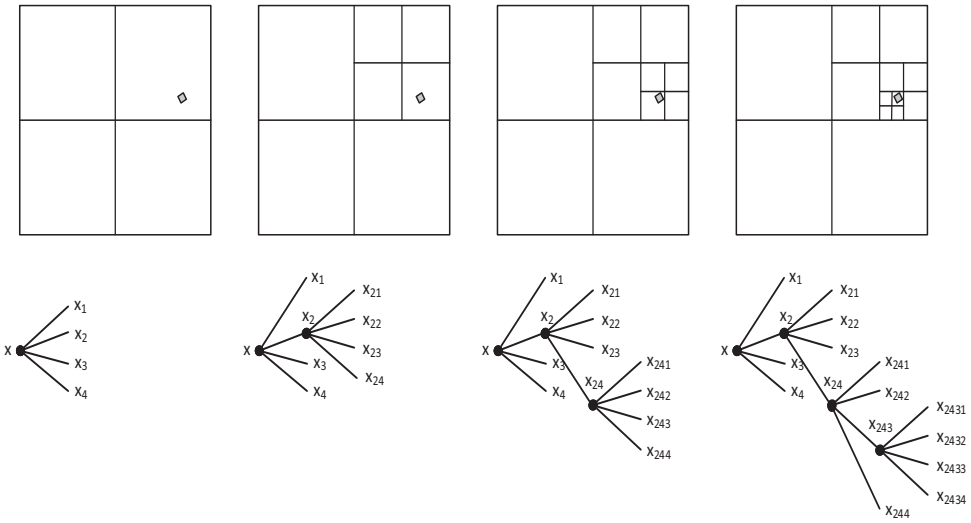

Fig. 3. FSM as multidimensional Fuzzy Signature

The values in the individual nodes can be interpreted as elements of a fuzzy signature, so a fuzzy situational map can be described as a multidimensional spatially structured fuzzy signature, see Fig. 3.

In the example, a very simple case of 'refining' the situational map by 2 × 2 grids is presented.

Following this interpretation, it can be said that each node in a FSM can be a further nested FSM and continued iteratively, this extension may go to depth *z* (applying increased resolution). The approach can lead to a fine structured FSM in each node as Fig. 4 shows. The resolution of nodes are independent of each other.

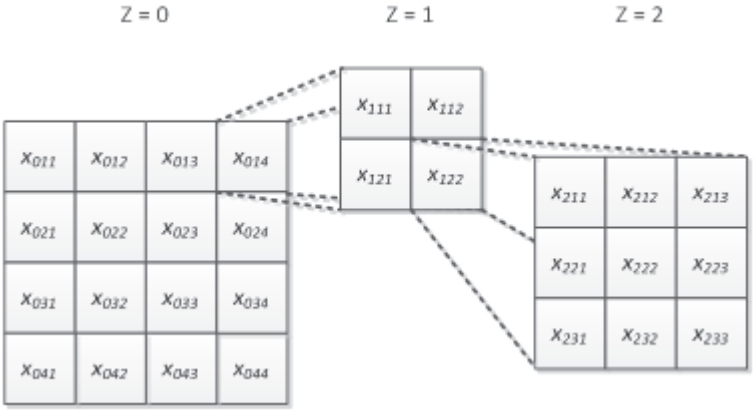

Fig. 4. Sub-lattices of a Fuzzy Situational Map

The individual nodes and the corresponding sub-lattices (high-resolution lattices) are related in the sense that the subgroups of sub-lattices jointly determine the features of the higher (parent) level. This structure, where each node can store significant amount of additional information which is processed only in the necessary resolution depth, can greatly reduce the computational requirements. FSM can describe hierarchically structured multidimensional data in a more concise way than simple fuzzy signatures.

4. *n*-dimensional fuzzy situational map

In the previous section, the properties of the two-dimensional fuzzy situational map were presented. Easily conceivable, these properties can be extended from the two-dimensional fuzzy situational map to a three-dimensional situational map, which may be used extensively in practice (eg. as a description of 3D robotic tasks).

Theoretically, there is no limit to increasing the dimension of a fuzzy situational map. Mathematically, any *n*-dimensional fuzzy situational map can be prepared, but from a practical point of view, increasing the dimension of a situational map exponentially increases the complexity of the inference on the situational map.

Consider a two-dimensional fuzzy situational map with a $h \times w$ node (h and w are the height and the width of the map, respectively), where each node is a leaf, so this situational map can be used in an inference system without any reduction or aggregation.

The computational complexity of a rule in such an inference system that is based on a fuzzy situational map can be calculated in a similar way to how Kóczy calculated it on a classic fuzzy inference system in [8, 9]. The C_{time} computational complexity is proportional to the resolution of the fuzzy situational map.

$$C_{\text{time}} = O(h \cdot w) \tag{2}$$

Let us increase the dimension of the situational map to n. In this case, the computational complexity turns into:

$$C_{\text{time}} = O(h \cdot w)^{n-1} \tag{3}$$

It can be concluded that the complexity increases exponentially, which can quickly result in an unacceptable level of computation time.

Of course, if the number of rules are increased in a FSM based inference system, then the computational complexity grows exponentially too [8, 9].

From a practical point of view, such a growth rate of requirements of computational capacity greatly reduce, the usability of the *n*-dimensional fuzzy situational maps. Therefore, we developed a new method for implementing the complex structured data description capacity of n-dimensional situational maps, whilst bypassing such computational capacity problems. The new algorithm is a multilayer fuzzy situational map.

5. The multilayer fuzzy situational map (MFSM)

The multilayer fuzzy situational map can be considered as several two-dimensional situational maps which are superimposed to each other, as Fig. 5 shows. Each individual layer can be handled as a two-dimensional situational map.

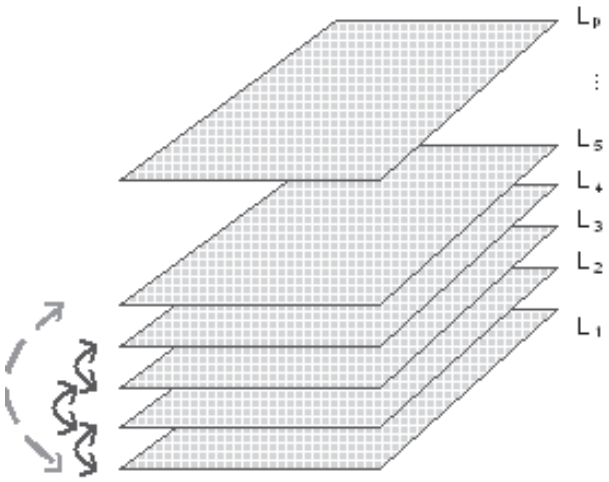

Fig. 5. Multilayer Fuzzy Situational Map

A multilayer fuzzy situational map always has a master-layer (or map), which contains the main situational map and carries the other layers. The sub-layers can be sorted over this master-layer for adding special information or modifying the data in the main layer. The sub-layers are called layer-maps.

The structure of a multilayer fuzzy situational map is always defined by the master-layer. It means that the basic structure of the master-layer and the layer-maps must be the same, that is, the master-layer structure can always be obtained from layer-maps by some reduction or aggregation.

In this case, the relationship between the layers and effects to each other are interpreted only on the identically indexed nodes . Of course, indirect effects may occur at other nodes.

The nodes with identical index in different layer may be connected in three type of relation:

– directed dependency,
– interdependency,
– independency.

Directed dependency means the nodes of one layer have influence over the nodes of another layer, but the second do not have any influence over the first mentioned layer. The directed dependency is not mutual. For example, in a robot navigation tasks the layer-map of the obstacles influences some nodes on the path planning layer, but the path planning layer does not influence the obstacle sub-layer.

Interdependency means mutual interaction between the layers. In this case, a more complex relationship must be described than in the other two cases. Often, these layers should be decomposed to additional directionally dependent layers for an acceptably practical result.

In the previous example, if multiple layers are used for route optimization processes, e.g. one for time optimization and one for energy consumption optimization, then these layers consist of interdependent nodes which require complex calculation and high computational capacity.

The layers are *independent* or *partly independent* if there is not any effective relation between the nodes of layer-maps, or the layers are relationship by only a few points.

5.1. Node connections between the layers

After the restructure of the layers of the situational map, the leaf nodes of the layer-maps are located in the same lattice points in each layer.

The node connections and effects between the layers vary depending on the task. From simple weighting and scaling relations, to complex functions, everything is conceivable. The weighting and other fuzzy operators are used on fuzzy situational maps actually.

The value of l_{p} leaf of the *MA* multilayer fuzzy situational map is obtained by the *Con* function which calculates the connection between the l_{p} leaf nodes of A_i and A_j layer-maps . These leaves may be the result of some reduction or aggregation.

$$\forall l_{\mathrm{p}} \text{ leaf } MA\left(l_{\mathrm{p}}\right) = Con\left(A_i\left(l_{\mathrm{p}}\right), A_j\left(l_{\mathrm{p}}\right)\right), \quad i = 1, ..., n; \;\; j = 1, ..., n; \;\; i \neq j \tag{4}$$

where n is the number of layers and $Con(x_{\mathrm{p}}, y_{\mathrm{p}})$ function calculates the modified value of connected leaves.

6. The benefits of multilayer fuzzy situational maps

The multilayer fuzzy situational maps with an n-layer have many practical advantages over n-dimensional non-layered fuzzy situational maps.

The information in the n-layered fuzzy situational map can be processed as n two-dimensional fuzzy situational maps, thus the demand on the computational capacity is dramatically reduced. In addition, the operation can be parallelized layer by layer on multi-processor or multi-core systems.

The layered structure of the situational map can be designed in a modular manner, which means that the individual layers can be removed, replaced or adapted in the multilayered situational map. Thus a complex information descriptor can be created that is as flexible as a classical fuzzy rule base. Deleting or modifying individual layers does not cause the destruction of the whole system, the situational map remains viable even in the absence of certain layers.

The modularity feature of the multilayer fuzzy situational map makes very flexible its applicability and allows for continuous development, in such autonomous and adaptive cases

too. Modular multilayered fuzzy situational maps are good base for machine learning systems. These are ensured by the definition of independency-dependency between the individual sub-layers, each layer affects only certain other layers in certain cases.

7. Route map pre-optimization with Fuzzy Situational Maps

In robotics, logistics and other disciplines, it plays an important role in finding the optimal routes, in other words, the minimization of one or more cost functions. There are a lot of methods to resolve these problems[12–17], but these procedures need high computational capacity, so in a real-time system, the size and the dimension of the function input data and the number of iteration steps to find the optimum solution makes big difference.

We have developed an algorithm based on multilayer fuzzy situational maps, which pre-optimizes the input data for the map based path planning functions. This algorithm reduces the amount of required input data (the size of the map) compared to the traditional procedures, so that the quality of path planning does not deteriorate.

The essence of this method is that more FSM layers are used to depict the separable system parameters. In our case, three FSMs are used: the target position FSM, the robot actual position FSM and the descriptive map of obstacles. The robot position FSM is a dynamic map, with resolution on the basis of the actual refining of the robot sensors (Fig. 6a)). The target FSM is a rough situational map, where only the target area has a finer resolution, due to the expected accuracy of positioning (Fig. 6b)). These two map layers may be combined and virtually, only the positions are important to us in terms of optimization. If the obstacles are not taken into account, then this map will be used to plan the route. Fig. 6c) shows an example for the planned route of the robot which is limited only to horizontal and vertical movements. In this paper, this limitation will be assumed.

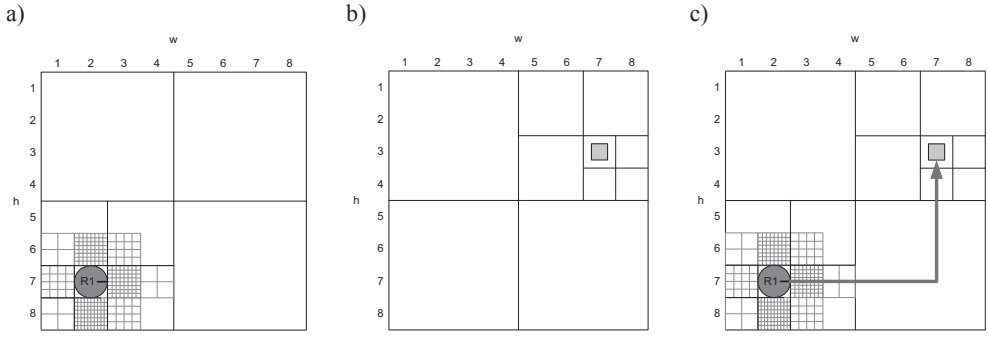

Fig. 6a) robot position FSM, b) goal position FSM, c) planed route

The situational map of obstacles has a similar structure as the preceding (Fig. 7), where the obstacles are represented by fuzzy values in the lattice. In the actual algorithm the map structure is taken in account, and here the fuzzy values have only secondary roles.

The obstacles FSM, as a new layer, is superimposed on the top of the position situational map.

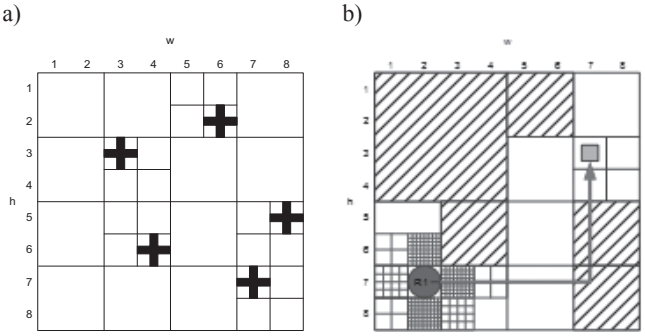

Fig. 7a) Situational map of obstacles, b) multilayered FSM

In order to make the initial contacts between the layers, the new layer should be converted into the common structure of the master layer. In this case, the master layer is the robot and goal position situational map and the additional layer is the obstacles situational map, so the structure of the obstacles situational map has to be converted. The resulting formed structures of layers are illustrated in Fig. 7.

The marked locations indicate the nodes where obstacles are located in the planned route. In these nodes the resolution must be refined, but only in these nodes. The refining steps are iterated until the accessible nodes, which map can be used to a final route optimization, they are used in as rough resolution as possible.

The next pseudo-code writes down the essential steps of this algorithm.

```
procedure
    while struct(robot_FSM) <> struct(goal_FSM)
        if struct(robot_FSM) > struct(goal_FSM) then      // > means higher resolution
          robot_FSM = reduce(robot_FSM)
        else
            goal_FSM = reduce(goal_FSM)
        endif
    endwhile
    pathmap_FSM = add(robot_FSM, goal_FSM)
    while struct(obstacle_FSM) > struct(pathmap_FSM)
        obstacle_FSM = reduce(obstacle_FSM)
    endwhile
    pathmap_MLFSM = addlayer(pathmap_FSM, obstacle_FSM)
    path = pathplanning(pathmap_MLFSM)
endproc
```

This way, the prepared FSM may contain far less grid points (Fig. 8a), b)) than a traditional route map (Fig. 8c)), so the path planning function can get results with a smaller amount of data and less iteration steps.

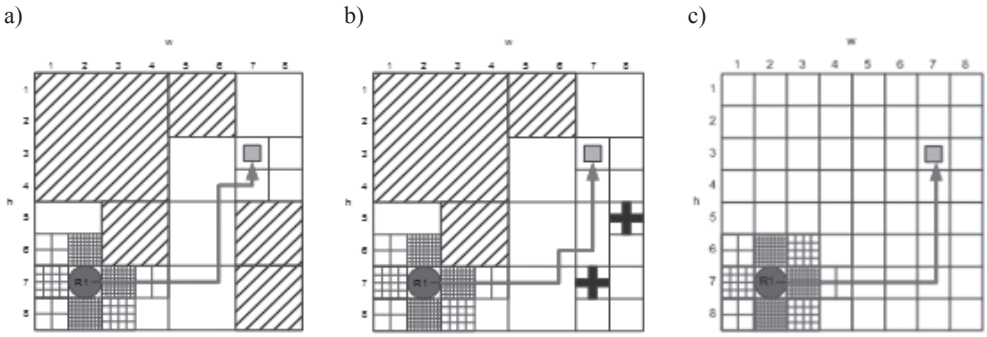

Fig. 8a) Fast route planning, b) route in a zoomed FSM, c) traditional route planning map

8. Conclusion

In this paper, we presented the essential idea of Fuzzy Situational Maps and the use of FSM algorithms in the field of mobile robotics map based route planning. These methods open a new way for complex decision-making. With Fuzzy Situational Maps, the complex structured and multidimensional data can be described in a compact and manageable manner. Thus, the difficult data management processes are becoming a more easily preparable system.

Here, we illustrated information processing by Fuzzy Signature Map might lead to an effective pre-optimized route planning map. In this example, the FSM allows optimization of the route with 15% less data than in the case of the same structured classical grid or map.

The research was supported by TÁMOP-4.2.2.A-11/1/KONV-2012-0012, Hungarian Scientific Research Fund Grant OTKA K108405 and OTKA K105529.

References

[1] Zadeh L.A., *Fuzzy Sets*, Information and Control, Vol. 83, 1965, 338-35.
[2] Goguen J.A., *L-fuzzy sets*, J. Math. Anal. Appl, Vol. 18, 1967, 145-174.
[3] Kóczy L.T., *Vectorial I-fuzzy Sets*, [in:] *Approximate Reasoning in Decision Analysis*, M.M. Gupta, E. Sanchez, Eds., ed Amsterdam: North Holland, 1982, 151-156.
[4] Vámos T., Bíró G., Kóczy L.T., *Fuzzy Signatures*, SIC-EUROFUSE'99, Budapest 1999, 210-217.
[5] Gedeon T.D., Kóczy L.T., Wong K.W., Liu P., *Effective Fuzzy Systems for Complex Structured Data*, IASTED International Conference on Control and Applications (CA 2001), 2001, 184-187.
[6] Vámos T., Kóczy L.T., Bíró G., *Fuzzy Signatures in Data Mining*, IFSA World Congress and 20th NAFIPS International Conference, Vancouver 2001, 2842-2846.
[7] Wong K.W., Chong A., Gedeon T.D., Kóczy L.T., Vámos T., *Hierarchical fuzzy signature structure for complex structured data*, in International Symposium on Computational Intelligence and Intelligent Informatics (ISCIII), Nabeul, Tunisia, 2003, 105-109.

[8] Ballagi Á., Kóczy L.T., Pozna C., *Fuzzy Situational Maps as Decision Support in Mobile Robotics*, Fourth Györ Symposium on Computational Intelligence, Györ 2011, 10-13.

[9] Ballagi Á., Kóczy L.T., Pozna C., *Intelligent Multi-Robot Cooperation*, Factory Automation 2011, Györ 2011, 248-251.

[10] Ballagi Á., Kóczy L.T., Pozna C., *Effective Multi-Robot Cooperation without Explicit Communication*, Fifth Györ Symposium and First Hungarian-Polish Joint Conference on Computational Intelligence, Györ 2012, 118-124.

[11] Ballagi Á., Kóczy L.T., Pozna C., *Man-Machine Cooperation without explicit communication*, World Automation Congress (WAC-ISSCI 2010), Kobe 2010.

[12] Ge S.S., Cui Y.J., *New potential functions for mobile robot path planning*, Ieee Transactions on Robotics and Automation, Vol. 16, Oct 2000, 615-620.

[13] Pruski A., Atassi A., *Sensor information space for robust mobile robot path planning*, Robotica, Vol. 18, Jul–Aug 2000, 415-421.

[14] Seneviratne L.D., Ko W.S., Earles S.W.E., *Triangulation-based path planning for a mobile robot*, Proceedings of the Institution of Mechanical Engineers Part C-Journal of Mechanical Engineering Science, Vol. 211, 1997, 365-371.

[15] Alexopoulos C., Griffin P.M., *Path Planning for a Mobile Robot*, Ieee Transactions on Systems Man and Cybernetics, Vol. 22, Mar–Apr 1992, 318-322.

[16] Acosta C., Moras R.G., *Path Planning Simulator for a Mobile Robot*, Computers & Industrial Engineering, Vol. 19, 1990, 346-350.

[17] Palmavillalon E., Dauchez P., *World Representation and Path Planning for a Mobile Robot*, Robotica, Vol. 6, Jan–Mar 1988, 35-40.

TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE
AUTOMATIC CONTROL | AUTOMATYKA
4-AC/2013

KRZYSZTOF SCHIFF*

ANT COLONY OPIMIZATION ALGORITHMS FOR CLUSTERING PROBLEMS

ALGORYTMY MRÓWKOWE DLA PROBLEMÓW KLASTERYZACJI

Abstract

The clustering problem is one of the main problems which can be encountered in a data analysis. This problem can be modelled by means of a graph; finding clusters means finding cliques in the graph. Often there is a need to find clusters (cliques) in a graph in different ways and to construct a list of clusters. This paper describes two such ways, these can be stated as the cluster minimum covering problem and the vertex cluster minimum partitioning problem. This paper describes new ant algorithms which were used in order to make a list of clusters in both presented problems, and also discusses the results of their comparison.

Keywords: *clustering, clique covering problem, clique vertex partitioning problem, ant algorithms*

Streszczenie

Problem klasteryzacji jest jednym z często spotykanych problemów w analizie danych. Problem klasteryzacji może być zamodelowany przy pomocy grafów i znajdowanie klasterów sprowadza się wówczas do znajdowania klik w grafach. W tym artykule opisano dwa sposoby wyznaczania klasterów, czyli klik w grafach, takich jak: problem pokrycia klastrami (klikami) grafu oraz problem wierzchołkowego podziału grafu na klastry (kliki) oraz także przedstawiono dwa nowe algorytmy bazujące na zachowaniu mrówek służące do wyznaczania klastrów (klik) dla obu problemów, a także dokonano porównania ich ze znanymi algorytmami rozwiązującymi te problemy.

Słowa kluczowe: *klasteryzacja, pokrycie klikami, wierzchołkowy podział na kliki, algorytm mrów*

* Ph.D. Krzysztof Schiff, e-mail: kschiff@pk.edu.pl, Department of Automatic Control and Information Technology, Faculty of Electrical and Computer Engineering, Cracow University of Technology.

1. Introduction

The clustering problem can be encountered in many optimisation problems, which are often difficult – finding solutions to such problems takes a great deal of time. Clustering techniques are used to shorten the time needed to find solutions to these problems. Data and dependency between data can be modelled by means of a graph. These data can be grouped into clusters according to the dependency of their characteristics. Finding clusters means finding cliques in a graph. The clique covering problem and the vertex clique partitioning problem are NP-difficult problems [1, 16]. Many papers have been devoted to clique problems [6, 8–12, 18]. Ant algorithms have been used to find maximum cliques or to find all cliques in a graph [2–5, 15]. The clique covering problem and the vertex clique partitioning problem were solved by means of a neural network [7, 13] and a genetic algorithm [14, 17]. This paper describes two new ant algorithms for the problems in question.

2. Models for cluster analysis

Let $G = (V, E)$ be a graph, where $V = \{v_1, v_2, ..., v_n\}$ is a set of vertices in graph G and $E = \{e_1, e_2, ..., e_m\}$ is a set of edges $m \leq n^2$. Graph $G = (V, E)$ is complete if each pair of its vertices $v_i, v_j \subseteq V$ is connected by edge $e_{ij} = \{v_i, v_j\} \subseteq E$. The clique C is a subset of all vertices which constitute the set V and which, when combined, constitute a complete graph. Clique C is called the cluster. Clique C is maximal if it is not included in another clique. It is maximum if this is a maximal clique and there is any other maximal clique in the graph with a higher number of vertices than this maximum clique. Relations between objects and their features and relations between features of different objects can be modelled by graphs or matrices. Vertices can represent only objects or objects and features. Edges can represent the existence of a common feature between two objects or the possession of a feature; this is shown in Fig. 1a and in Fig. 1b, respectively. A value of 1 in each matrix represents an existence and a value of 0 represents an absence.

a)

c\n	1	2	3	4	5	6
1	1	0	0	0	0	1
2	0	1	1	1	1	0
3	1	1	1	1	1	1
4	0	1	1	1	0	0
5	0	1	1	1	1	0
6	1	0	0	0	0	1

b)

c\n	1	2	3	4	5	6
1	1	1	1	1	0	1
2	1	1	1	1	1	0
3	1	1	1	1	0	0
4	1	1	1	1	0	0
5	0	1	0	0	1	0
6	1	0	0	0	0	1

Fig. 1. Matrices for cluster models

An object is represented by variable n in Fig. 1a. If there is a feature common to two objects, a value of 1 is used, and if there is no common feature, a value 0 of is used in the

matrix. The cluster which consists of objects {2, 3, 4, 5} means that objects 2, 3, 4 and 5 possess the same feature. If there is a need to investigate many features between many objects, then for each of these features, a graph or matrix model should be built and for each of these features in each of these models, a cluster analysis should be performed. This means that a list of maximal cliques should be indicated.

An object is represented by variable *n* and a feature is represented by variable *c* in Fig. 1b. Both are represented by a graph vertex, matrix column or matrix verse. If an object possesses a feature *c,* a value of 1 is used, and if it does not possess feature *c*, a value of 0 is used in the matrix at the crossing of the matrix column and matrix verse. A cluster which consists of objects {1, 2, 3, 4} and features {1, 2, 3, 4} means that objects 1, 2, 3, 4 possess common features 1, 2, 3 and 4. The graph shown in Fig. 1b can be used only in cases where the number of objects and features is equal. If the number of objects and features is not equal, these numbers have to be made equal. Either the verses of the matrix should be filled with a value of 1 in cases where the number of objects is higher than the number of features, as shown in Fig. 2a, or columns of the matrix should be filled with a value of 1 in the opposite case, as shown in Fig. 2b; thus, both of these cases are transformed to the model shown in Fig. 1b.

A case in which the number of features is less than the number of objects is shown in Fig. 2a. There are 5 features and 6 objects, so the 6th verse is filled with a value of 1; this means that a common feature was added to all objects. A case in which the number of features is greater than the number objects is shown in Fig. 2b. There are 6 features and 5 objects, so the 6th column is filled with a value of 1; this means that an object was augmented with features which are common to other objects. After the maximum clique has been indicated, the added features or added objects are removed.

a)

c\n	1	2	3	4	5	6
1	1	0	0	0	0	1
2	0	1	1	1	1	0
3	1	1	1	1	1	1
4	0	1	1	1	1	0
5	0	1	1	1	1	0
6	1	1	1	1	1	1

b)

c\n	1	2	3	4	5	6
1	1	1	1	1	0	1
2	1	1	1	1	1	1
3	1	1	1	1	0	1
4	1	1	1	1	0	1
5	0	1	0	0	1	1
6	1	0	0	0	0	1

Fig. 2. Matrix: a) when the number of features is lower than the number of objects, b) in the opposite case

If there is a need to perform a multi-feature cluster analysis, then for each of these features, a cluster matrix and later, a cumulative matrix are created. Matrices for 4 different features of 6 objects are shown in Fig. 3a, b, c and d. There is matrix of 6 columns by 6 verses. Cumulative matrices for a probability equal to 0.6 are shown in Fig. 4a; a probability equal to 0.8 is shown in Fig. 4b. Matrices 4a and 4b have been obtained such that the number of features for each object in all 4 matrices shown in Fig. 3a, b, c and d were added and divided by the number of matrices. Thus the obtained average values of feature existence were compared with the value of probability and

a)

n\n	1	2	3	4	5	6
1	1	0	0	0	0	1
2	0	1	1	1	1	0
3	1	1	1	1	1	1
4	0	1	1	1	1	0
5	0	1	1	1	1	0
6	1	0	0	0	0	1

b)

n\n	1	2	3	4	5	6
1	1	1	1	1	0	1
2	1	1	1	1	1	0
3	1	1	1	1	0	0
4	1	1	1	1	0	0
5	0	1	0	0	1	0
6	1	0	0	0	0	1

c)

n\n	1	2	3	4	5	6
1	1	0	0	0	0	1
2	0	1	0	1	1	0
3	1	0	1	0	1	1
4	0	1	0	1	1	0
5	0	1	1	1	1	0
6	1	0	0	0	0	1

d)

n\n	1	2	3	4	5	6
1	1	0	1	1	0	1
2	0	1	1	1	1	0
3	1	1	1	1	0	0
4	1	1	1	1	1	1
5	0	1	0	1	1	0
6	1	0	0	1	0	1

Fig. 3. Matrices – for features a, b, c and d

a)

n\n	1	2	3	4	5	6
1	1	0	0	0	0	1
2	1	1	1	1	1	0
3	1	1	1	1	0	0
4	0	1	1	1	1	0
5	0	1	0	1	1	0
6	1	0	0	0	0	1

b)

n\n	1	2	3	4	5	6
1	1	0	0	0	0	1
2	0	1	0	1	1	0
3	1	0	1	0	0	0
4	0	1	0	1	1	0
5	0	1	0	0	1	0
6	1	0	0	0	0	1

Fig. 4. Cumulative matrices

then, if the value of probability was lower than the average value of the existence of features, a value of 1 was written as needed in the cumulative matrix. For example: there is a cellule at the crossing of verse 2 and column 1 in all 4 matrices; in this cellule, a value of 1 is encountered 3 times in all 4 matrices. If the number 3 is divided by 4 (since there were 4 matrices), then the number 0.75 is obtained. When the probability value is equal to 0.6, then in this cellule, a value of 1 should be written as in Fig. 4a, but when the probability value is equal to 0.8, then in this cellule, a value of 0 should be written as in Fig. 4b. A cumulative matrix for cluster analysis with a probability of 0.6 is shown in Fig. 4a and with a probability of 0.8, is shown in Fig. 4b.

3. Two cluster problems

Set $S = \{C_1, C_2, ..., C_k\}$ is a cover of the graph G, if all graph edges $E(G)$ are covered by edges of cliques C_i and:

$$E(G) = \sum_{i=1}^{k} E(C_i) \tag{1}$$

Where: k is the number of cliques (clusters).

If there is a set S of clusters in which each edge belongs to at least one cluster C_i, $1 \le i \le k$, then S is a graph covered by clusters. The cardinality number of S is called the cluster covering number of graph G and is marked by cc(G).

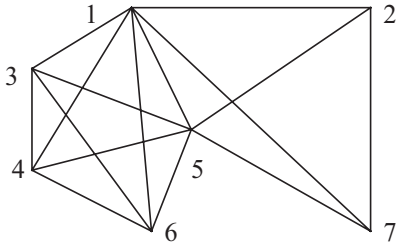

Fig. 5. A graph for two problems

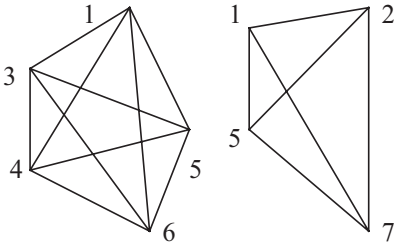

Fig. 6. A solution to the clique covering problem

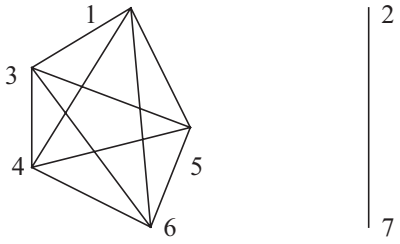

Fig. 7. A solution to the vertex clique partitioning problem

A set S of clusters in which each vertex belongs to exactly one element of S means that for two different cliques C_i and C_j, $1 \le i \le k$, $1 \le j \le k$, $i \ne j$, there are no common vertices. $V(C_i) \cap V(C_j) = \varnothing$ is called a graph vertex partitioning on clusters. The cardinality number of S is called the clique vertex partitioning number of graph G and is marked by cwp(G).

Definition: minimum clique (cluster) covering problem in a graph

A minimum number of maximal cliques is searched for in graph $G = (V, E)$ such that each graph edge has two of its ends in at least one clique; in other words, a set S with a minimum number cc(G) is searched for.

Definition: minimum clique (cluster) vertex partitioning problem in graph

A minimum number of maximal cliques is searched for in graph $G = (V, E)$ such that each graph vertex belongs to exactly one clique; in other words, a set S with minimum number cwp(G) is searched for.

Differences between these two problems were shown in Figs. 6 and 7 as a different kind of solution for an example of the graph presented in Fig. 5. This means a solution to the clique covering problem and a solution to the vertex clique partitioning problem, respectively.

4. Ant algorithms for cluster problems

The solution to the minimum cluster covering problem is a list of maximal cliques. There should be no repetitions within the list. Each of these maximal cliques is constructed by ants for each graph edge. The pseudo-code of the ant algorithm is shown as algorithm 1.

The solution for the minimum vertex partitioning problem is a list of maximal cliques. Each of these maximal cliques is constructed by ants in a current graph structure, which is obtained from the preceding graph structure by removing its edges and vertices. The pseudo-code of the ant algorithm was shown as algorithm 2.

Maximal cliques were determined in both algorithms. In general, a maximal clique is created by vertices which are neighbours, this means that each pair of vertices from a clique are connected by an edge. A maximal clique is created so that at first, one and then another vertex is selected from the neighbours of the first selected vertex; then the third vertex is selected from the neighbours of the first two selected vertices and so on, until there is no vertex among the neighbours of the vertices already selected. The order in which these vertices are selected influences the size of the created maximal clique. The size of the clique depends on the vertex selection sequence; thus it is important to know which vertices should be selected, and in which sequence, in order to obtain the maximal clique. Since there are so many possible selection sequences, there is no way to check them all out; this is why ant algorithms are used to select vertices and to create the maximal clique.

Algorithm 1

Cluster covering procedure

```
Repeat for each graph edge
  while (there is a cycle to repeat)
        while (there is an ant which has not yet worked)
       while (the clique has not been completed)
            include one of the vertices next to the maximal clique with probability p
    remember the best solution which was found by all ants in one cycle
  remember the best solution which has been found so far in all cycles
  update pheromone trails
  include the clique on the list
```

Algorithm 2

Cluster vertex partitioning procedure

```
while (there is a cycle to repeat)
      while (there is an ant which has not yet worked)
         while (the clique has not been completed)
          include one of the vertices next to the maximal clique with probability p
      remember the best solution which was found by all ants in one cycle
  remember the best solution which has been found so far in all cycles
  update pheromone trails
  remove vertices and edges which participated in the last clique from the graph
  include the clique on the list
```

Algorithm 3

Maximum clique (cluster) procedure

$C = \varnothing$
$N = \varnothing$
select a first vertex $v_i \in V$
$C = C + \{v_i\}$
$N = N +$ all $\{v_j : (v_j, v_i) \in E\}$
while N $\neq \varnothing$ **do**
  select a vertex $v_i \in N$ with probability $p(i)$
  $C = C + \{v_i\}$
  $N = N \cap$ all $\{v_j : (v_i, v_j) \in E\}$
end while
return C

In the ant algorithm, which would be called the ALG algorithm later in this paper and which is presented in paper [4], a vertex is included in the maximal cluster with probability:

$$p(j) = \frac{[t_j]^x}{\sum_{v_j \varepsilon \text{Candidates}} [t_j]^x} \tag{2}$$

In the ant algorithm, which would be called the NALG algorithm later in this paper and which is presented in this paper, a vertex is included in the maximal cluster with probability:

$$p_j = \begin{cases} \dfrac{\tau_j^{\alpha} n_j^{\beta}}{\sum_{j \in N_i} \tau_j^{\alpha} n_j^{\beta}}, & \text{for } \; j \in N_i \\ 0, & \text{for } \; j \text{ which not} \notin N_i \end{cases} \tag{3}$$

This probability depends on the pheromone trail and on the desire for vertex selection expressed by the formula:

$$n_j = \frac{d_j d_j}{\sum_{j=1}^{m} (d_j d_j)}, \; j \in N_i \tag{4}$$

where: d_j is a vertex degree, i.e., the number of edges adjacent to vertex j.

After all ants have worked in each cycle, some of the pheromone evaporates at rate r according to expression $\tau = r\tau$. On all vertices which, taken together, constitute the maximal clique, a pheromone is deposited – this pheromone quantity is expressed as

$$\Delta\tau = \frac{1}{1 + \dfrac{cs_{best} - cs}{cs_{best}}} \tag{4.4}$$

where: cs is the size of the maximal clique.

5. Experiments

Tests were conducted for both of the ant algorithms, ALG and NALG, mentioned above and for two problems which have been taken into consideration: the minimum clique covering problem and the minimum vertex partitioning problem.

The first test was conducted for the minimum clique covering problem for a constant number of vertices $n = 100$ and for different graph densities $q = \{0.1, 0.3, 0.5, 0.7, 0.9\}$ with a constant evaporation rate $r = 0.99$, a constant number of cycles $lc = 50$ and a constant

number of ants m = 30. Average values from 100 measurements are shown in Table 1. It is advantageous to use the NALG algorithm rather than the ALG, since there is a positive difference over the entire range of graph density {0.1, 0.3, 0.5, 0.7, 0.9}.

Table 1

Covering: number of cliques as a function of graph density

q	0.1	0.3	0.5	0.7	0.9
ALG	306.2	948.9	2184.8	3402.8	4450.0
NALG	306.0	761.4	1975.1	3369.3	4449.7
ALG – NALG	0.2	187.5	209.7	33.5	0.3

The second test was also conducted for the minimum clique covering problem, but for a different number of vertices n = {50, 100, 150, 200, 250} and for a constant graph density q = 0.5 with a constant evaporation rate r = 0.99, a constant number of cycles lc = 50 and a constant number of ants m = 30. Average values from 100 measurements are shown in Table 2. It is advantageous to use the NALG algorithm rather than the ALG, since there is a positive difference over the entire range of graph vertices {50, 100, 150, 200, 250}.

Table 2

Covering: number of cliques as a function of the number of vertices

n	50	100	150	200	250
ALG	390.9	1975.1	5150.5	9446.8	15489
NALG	457.5	2184.8	5397.5	9688.2	15654.8
ALG – NALG	66.6	209.7	247	241.4	165.8

The third test was conducted for the minimum vertex clique partitioning problem, for a constant number of vertices n = 100 and for a different graph density q = {0.1, 0.3, 0.5, 0.7, 0.9} with a constant evaporation rate r = 0.99, a constant number of cycles lc = 50 and a constant number of ants m = 30. Average values from 100 measurements are shown in Table 3. It is advantageous to use the NALG algorithm rather than the ALG, since there is a positive difference over the entire range of graph density {0.1, 0.3, 0.5, 0.7, 0.9}.

Table 3

Vertex partitioning: number of cliques as a function of graph density

q	0.1	0.3	0.5	0.7	0.9
ALG	42.58	27.1	17.78	12.53	6.8
NALG	42.47	26.8	17.61	12.45	6.69
ALG – NALG	0.11	0.3	0.17	0.08	0.11

The last test was conducted for the minimum vertex clique partitioning problem, but for a different number of vertices $n = \{50, 100, 150, 200, 250\}$ and for a constant graph density $q = 0.5$ with a constant evaporation rate $r = 0.99$, a constant number of cycles $lc = 50$ and a constant number of ants $m = 30$. Average values from 100 measurements are shown in Table 4. It is advantageous to use the NALG algorithm rather than the ALG, since there is a positive difference over the entire range of graph vertices {50, 100, 150, 200, 250}.

Table 4

Vertex partitioning: number of cliques as a function of the number of vertices

n	50	100	150	200	250
ALG	11.95	17.78	24.44	30.2	35.33
NALG	11.91	17.61	24.11	29.76	34.52
ALG – NALG	0.04	0.17	0.33	0.44	0.81

6. Conclusion

The NALG algorithm described in this paper was compared with the previously elaborated ALG algorithm. It has been shown that the NALG algorithm has a permanent advantage over the ALG, since it obtained lists of clusters in both problems taken into consideration shorter than those obtained by the already elaborated ALG for a broad range of graph density and number of graph vertices.

References

[1] Garey M., Johnson D., *Computers and Intractability: A Guide to the Theory of NP-Completeness*, Freeman, New York 1979.

[2] Ponce J., Ponce E., Padilla F., Padilla A., *Ant Colony Algorithm for Clustering through of Cliques*, 2006.

[3] Ponce J., Ponce E., Padilla F., Padilla A., Ochoa A., *Algoritmo De Colonia De Hormigas Para El Problema Del Clique Máximo Con Un Optimizador Local K-Opt*, Hífen, Uruguaiana, Vol. 30, No. 58, Brasil 2006.

[4] Fenet S., Solnon C., *Searching for Maximum Cliques with Ant Colony Optimization EvoWorkshops 2003*, LNCS 2611, 2003, 236-245.

[5] Xu X., Ma J., Lee J., *An Improved Ant Colony Optimization for the Maximum Clique Problem*, IEEE Third International Conference on Natural Computation, 2007.

[6] Bron C., Kerbosh J., *Finding all cliques of an undirected graph*, Association of Computer Machine, 16, 1973, 575-577.

[7] Wang J., Xu X., Tsng Z., Bi W., Chen X., Li Y., *A New Neural Network Algorithm for Clique Vertex-Partition Problem*, Yin F., Wang J., Guo C. (Ed.): ISNN 2004, LNCS 3173, Springer-Verlag, Berlin 2004, 425-29.

[8] Kellerman E., *Determination of keyword conflict*, IBM Technical Disclosure Bulletin 16, 1973, 544-546.

[9] Behrisch M., Taraz A., *Efficiently covering complex networks with cliques of similar vertices*, TCS: Theory of Computer Science 355, 2006, 37-47.

[10] Cazals F., Karnade C., *An algorithm for reporting maximal c-cliques*, Theory of Computer Science 349, 2005, 484-490.

[11] Gramm J., Guo J., Huffner F., Niedermeier R., *Data reduction and exact algorithms for clique cover*, ACM J. Exp. Algorith. 13, 2.2–2.15, 2009.

[12] Tseng C., Siewiorek D., IEEE Trans. CAD 5, 379 (1986).

[13] Harmanani H., *A Parallel Neural Networks Algorithm for the Clique Partitioning Problem*, IJCA, Vol. 9, No. 2, 2002

[14] Little P., Rylander B., *Problem Partitioning in Hybrid Genetic Algorithms*, Proceedings of the 5th WSEAS Int. Conf. on circuits, systems, electronics, control and signal processing, Dallas, USA, November 1–3, 2006

[15] Rizzo J., *An Ant System Algorithm for Maximum Clique*, The Pennsylvania State University, Master Thesis, 2003.

[16] Orlin J., *Contentment in graph theory: covering graphs with cliques*, Mathematics 39, 1977, 406-424.

[17] Snyers D., Clique Partitioning Problem and Genetic Algorithms in Albrecht R.F.(eds.), Artificial Neural Nets and Genetic Algorithms Springer-Verlag, 1993.

[18] Tseng C., Sieworek D., *Facet: A Procedure for the Automated Synthesis of Digital Systems*, Proceeding of the 20th ACM/IEEE Design Automated Conf., 1983, 490-496.

TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE
AUTOMATIC CONTROL | AUTOMATYKA
4-AC/2013

ADRIAN NAKONECHNYY, ZENOVIY VERES*

ADAPTIVE TRAINED FILTERS WITH SSIM OPTIMIZATION FOR VIDEO UPSCALING WITH SOFT PIXEL VALUE TRUNCATION

ADAPTACYJNE FILTRY UCZĄCE SIĘ Z OPTYMALIZACJĄ SSIM DLA ZWIĘKSZENIA ROZDZIELCZOŚCI OBRAZU Z MIĘKKIM PROGOWANIEM WARTOŚCI PIKSELI

Abstract

The paper presents the modified model for video upscaling based on the preservation of structural information from the input sequence in the adaptive trained filters. Additionally, it is proposed to make soft truncation of the result pixel value during the interpolation process. The evaluation of the proposed algorithm has shown good resulting for a variety of test sequences. The best results are obtained for a sequence with average bitrate and movement of the scene. Received results prove the algorithm is valuable for upscaling tasks.

Keywords: *image and video interpolation, up-scaling, image quality*

Streszczenie

W artykule przedstawiono zmodyfikowany model zwiększania rozdzielczości, oparty na przechowywaniu informacji strukturalnej z wejściowej sekwencji w adaptacyjnych filtrach uczących się. Dodatkowo zaproponowano miękkie progowanie wartości pikseli podczas procesu interpolacji. Weryfikacja proponowanego algorytmu wykazała korzystne rezultaty dla różnych sekwencji testowych. Najlepsze wyniki uzyskano w przypadku sekwencji o średniej szybkości transmisji i przy ruchomym obrazie. Wskazują one na korzyści wynikłe ze stosowania algorytmu w zadaniach zwiększania rozdzielczości obrazu.

Słowa kluczowe: *interpolacja zdjęć i video, zwiększanie rozdzielczości, jakość obrazu*

* Prof. D.Sc. Ph.D. Adrian Nakonechnyy, e-mail: , M.Sc. Zenoviy Veres, Department of Computerized Automatic Systems, Lviv Polytechnic National University.

1. Introduction

High definition television (HDTV) is becoming a standard appliance in every modern household. Also, with the introduction of HDTV-capable TV receivers, the transmission of standard definition television (SDTV) material will not stop immediately. In general, the price of high resolution screens has come down to a level that is affordable, even for TVs that have no HDTV reception. This raises the problem where low-resolution video materials have to be upscaled to fit the resolution of HDTV.

The set of linear techniques, such as nearest neighbor, bi-linear, and bi-cubic [8] interpolations have been popular in many applications. However, these linear methods usually result in blurred images, because the scaling process does not add new frequency components to interpolated images.

Several advanced non-linear image and video upscaling algorithms have been developed recently to deal with this problem [10]. Those video enhancement algorithms include sharpness enhancement, noise/coding artifacts reduction, resolution up-conversion, contrast enhancement, etc. This approaches provide good result image quality but requires much time in order to process the input image. This time consumption is acceptable for plain images, but they become a bottleneck for the video frames that are required to be transformed in the real time mode.

A family of trained filters [1] was developed that are used in real time mode for upscaling the video frames. The existing filter modifications uses different approaches in order to receive better output image quality [3]. In order to receive better quality of the output frames, in this paper, a novel interpolation technique based on optimizing filter coefficients based on structural information is presented. The proposed algorithm is evaluated using Mean Square Error (MSE) and Structural SIMilarity (SSIM) metrics.

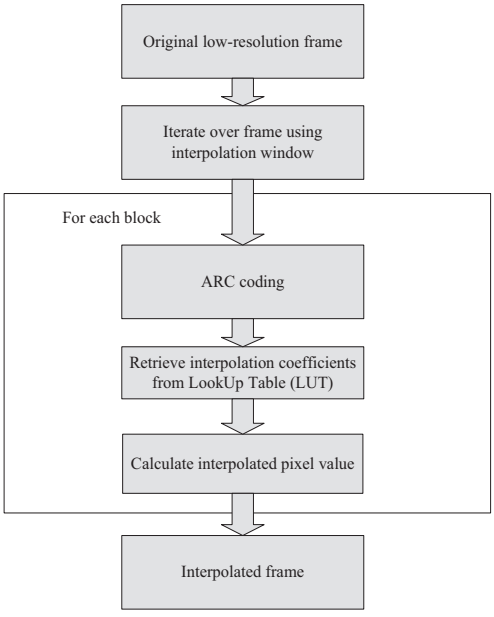

Fig. 1. Interpolation process with usage of trained filters

2. Trained Filters

Trained filters [1, 3, 6, 10] are widely used to solve image and video up-scaling tasks due to its simplicity. The interpolation process for trained filter is depicted at Fig. 1.

Low-resolution video frames are passed to the filter's input. Interpolation is performed in sliding window mode – one interpolated pixel value is evaluated based on all pixel values that belong to the interpolation window. The interpolation window represents a square block of 3×3 pixels. For each pixel to be interpolated, the pixels from the interpolation window

are classified using a classification method. The classification output provides the key for picking up filtering coefficients in the filter's look-up table (LUT) that are used for the pixel value calculation.

To obtain filtering coefficients of the filter the training process should be executed in advance. Fig. 2 depicts the training process of the trained filters. Original images are first downscaled by a factor of two according to the specification of the application. The training process employs the original video sequences and corresponding downscaled video sequences as the training material and uses the Least Mean Squares (LMS) criterion to get the optimal coefficients, which is computationally intensive due to the large number of classes. Fortunately, it needs to be performed only once.

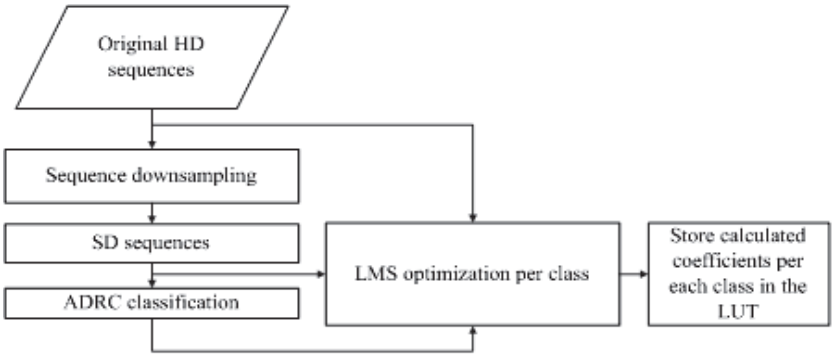

Fig. 2. Training process of the trained filter

Images after downsampling are referred to as downscaled images. In the downscaled images, each pixel and the pixels in its vicinity are characterized using a specific classification method. All the pixels and their neighbours belonging to a specific class and their corresponding pixels in the target (original) images are accumulated, and the optimal coefficients are obtained by making the MSE minimized statistically.

Let F_{SD}, F_{HD} represent pixels of the degraded images and the reference images for a particular class c, respectively. Then the interpolated pixel F_{HI} can be obtained by the desired optimal coefficients as follows:

$$F_{HI} = \sum_{k=0}^{n} w_c(k) \cdot F_{SD}(k) \tag{1}$$

where:

$w_c(i)$, $i \in [1 \ldots n]$ – are the desired coefficients,
n – is the number of pixels in the aperture,
j – is the indicator of the particular aperture that represents the class c.

The summed square error between the filtered pixels and the reference pixels is:

$$e^2 = \sum_{j=1}^{N_c} \left(F_{R,c} - F_{F,c} \right)^2 = \sum_{j=1}^{N_c} \left[F_{R,c}(j) - \sum_{i=0}^{n} w_c(i) F_{D,c}(i,j) \right]^2 \tag{2}$$

where:

N_c – represents the number of training samples belonging to class c.

To minimize e^2, the first derivative of e^2 to $w_c(k)$, $k \in [1 \ldots n]$ should be equal to zero.

$$\frac{\partial e^2}{\partial w_c(k)} = \sum_{j=1}^{N_c} 2 \cdot F_{D,c}(i,j) \cdot \left[F_{R,c}(j) - \sum_{i=0}^{n} \omega_c(i) F_{D,c}(i,j) \right] \tag{3}$$

The optimal coefficients $w_c(k)$ are obtained by making the MSE minimized statistically. The calculated coefficients are then stored in a look-up table (LUT) for future use.

The local block content of the image can be classified based on the pattern of the image region and structure. Adaptive Dynamic Range Coding (ADRC) [1] is proposed as a powerful method for representing the structure of the region because of its high efficiency and simplicity. Let $x_1, x_2, \ldots, x_n$ be pixel values, AV – is the average of all the pixel values in the aperture. The ADRC code per pixel is defined as follows:

$$\mathrm{ADRC}(x_i) = \begin{cases} 1, & x_i > AV \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

ADRC coding diagram is shown in Fig. 3.

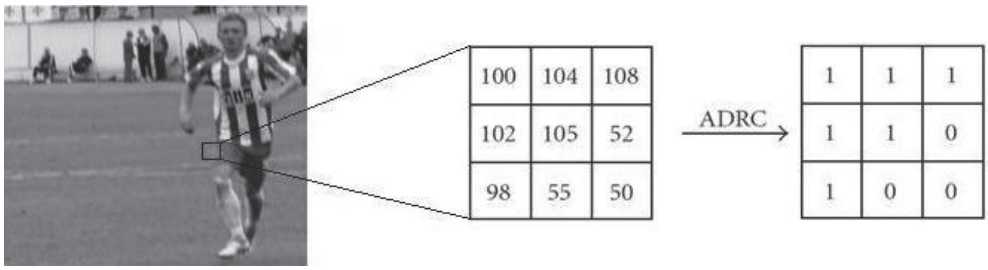

Fig. 3. ADRC coding of a 3 × 3 block

The main advantage of ADRC is its simple implementation. Classes count is decreased from 256^9 to 2^9 for apertures containing 3 × 3 SD pixels using Equation 1. It has been shown in [1] that if the image data is inverted, the coefficients in the LUT should remain the same.

By combining the two complementary classes, the size of the LUT is reduced to 2^8 without loss of image quality.

3. Main challenges of the trained filters

The optimal filtering coefficients are calculated during 'offline' training process. Filters incorporate the MSE metric for the coefficient's calculation that shows good results for the variety of tasks. However, this metric has several issues [7, 11]:

1. Digital pixel values, on which the calculation of this metric is based, may not accurately represent the light stimulation which is perceived by the human eye.
2. Human visual system's (HVS) sensitivity to errors is different for different types of errors and may also vary depending on image content. This difference is not counted correctly in the MSE calculation.
3. Two distorted images with the same errors energy may have different types of errors.
4. A simple summation of errors, which is implemented in the MSE calculation, is different from how the HVS and brain perceive the quality of the received image.

As a consequence, trained filters are sensitive to the same issues. The filter coefficients are calculated based on a predefined set of test sequences. The filters show good results for high-quality sequences if they were trained on high-quality ones but can fail when the sequence is distorted by noise. To address this issue, it is possible to create separate coefficients set for each input sequence type, but it will require much memory for their storage. To improve the quality of the result sequences, MSE metric is proposed to be replaced with another one. The structural similarity index – SSIM [4, 5] has been shown good results for objective evaluation of image and video quality. Usage of SSIM allows taking into account the structural information during coefficient's evaluation.

4. Structural Similarity Index [4]

The most fundamental principle underlying structural approaches to the image and video quality assessment (QA) is that the human visual system (HVS) is highly adapted to extract structural information from the visual scene. Therefore, a measurement of structural similarity or distortion should provide a good approximation to perceptual image quality.

The main idea that underlies the structural similarity (SSIM) index is a comparison of the distortion of three image components:

– Luminance
– Contrast
– Structure

Depending on how structural information and structural distortion are defined, there may be different ways to develop image QA algorithms. The SSIM index is a specific implementation from the perspective of image formation. The luminance of the surface of an object being observed is the product of illumination and the reflectance, but the structures of the objects in the scene are independent of the illumination. Consequently, we wish to

separate the influence of illumination from the remaining information that represents object structures. Intuitively, the major impact of illumination change in the image is a variation of the average local luminance and contrast, and such variation should not have a strong effect on perceived image quality.

Consider two image patches, $\tilde{f}$ and $\tilde{g}$, obtained from the reference and test images. $\tilde{f}$ and $\tilde{g}$ consist of the two vectors of dimension N, where $\tilde{f}$ is composed of N elements of $f(n)$ spanned by a window B and similarly for $\tilde{g}$. To index each element of $\tilde{f}$, we use the notation $\tilde{f} = \left[\tilde{f}_1, \tilde{f}_2, \ldots, \tilde{f}_N\right]^t$.

First, the luminance of each signal is estimated as the mean intensity:

$$\mu_{\tilde{f}} = \frac{1}{N}\sum_{i=1}^{N}\tilde{f}_i, \quad \mu_{\tilde{g}} = \frac{1}{N}\sum_{i=1}^{N}\tilde{g}_i \tag{5}$$

A luminance comparison function $l(\tilde{f}, \tilde{g})$ is then defined as a function of $\mu_{\tilde{f}}$ and $\mu_{\tilde{g}}$:

$$l(\tilde{f}, \tilde{g}) = \frac{2\mu_{\tilde{f}}\mu_{\tilde{g}} + C_1}{\mu_{\tilde{f}}^2 + \mu_{\tilde{g}}^2 + C_1} \tag{6}$$

where:

C_1 – the constant that is included to avoid instability when $\mu_{\tilde{f}}^2 + \mu_{\tilde{g}}^2$ is very close to zero.

C_1 is taken as follows:

$$C_1 = (K_1 L)^2 \tag{7}$$

where:

L – is the dynamic range of the pixel values (255 for 8-bit grayscale images),
$K_1 << 1$ – is a small constant.

Similar considerations also apply to contrast comparison and structure comparison terms.

The contrast of each image patch is defined as an unbiased estimate of the standard deviation of the patch:

$$\sigma_{\tilde{f}}^2 = \frac{1}{N-1}\sum_{i=1}^{N}(\tilde{f}_i - \mu_{\tilde{f}})^2, \qquad \sigma_{\tilde{g}}^2 = \frac{1}{N-1}\sum_{i=1}^{N}(\tilde{g}_i - \mu_{\tilde{g}})^2 \tag{8}$$

The contrast comparison $c(\tilde{f}, \tilde{g})$ takes a similar form as the luminance comparison function and is defined as a function of $\sigma_{\tilde{f}}$ and $\sigma_{\tilde{g}}$:

$$c(\tilde{f},\tilde{g}) = \frac{2\sigma_{\tilde{f}}\sigma_{\tilde{g}} + C_2}{\sigma_{\tilde{f}}^2 + \sigma_{\tilde{g}}^2 + C_2} \tag{9}$$

where:

C_2 – is a nonnegative constant:

$$C_2 = (K_2 L)^2 \tag{10}$$

where:

K_2 – satisfies $K_2 << 1$.

The signal is normalized (divided) by its own standard deviation so that the two signals being compared have unit standard deviation. The structure comparison $s(\tilde{f},\tilde{g})$ is conducted on these normalized signals. The SSIM framework uses a geometric interpretation, and the structures of the two images are associated with the direction of the two unit vectors $\tilde{f} - \mu_{\tilde{f}}/\sigma_{\tilde{f}}$ and $\tilde{g} - \mu_{\tilde{g}}/\sigma_{\tilde{g}}$. The angle between the two vectors provides a simple and effective measure to quantify SSIM. In particular, the correlation coefficient between $\tilde{f}$ and $\tilde{g}$ corresponds to the cosine of the angle between them and is used as the structure comparison function:

$$s(\tilde{f},\tilde{g}) = \frac{\sigma_{\tilde{f}\tilde{g}} + C_3}{\sigma_{\tilde{f}}\sigma_{\tilde{g}} + C_3} \tag{11}$$

The covariance function $\sigma_{\tilde{f}\tilde{g}}$ between $\tilde{f}$ and $\tilde{g}$ is estimated as:

$$\sigma_{\tilde{f}\tilde{g}} = \frac{1}{N-1}\sum_{i=1}^{N}(\tilde{f}_i - \mu_{\tilde{f}})(\tilde{g}_i - \mu_{\tilde{g}}) \tag{12}$$

Finally, the SSIM index between image patches $\tilde{f}$ and $\tilde{g}$ is defined as:

$$\mathrm{SSIM}(\tilde{f},\tilde{g}) = l(\tilde{f},\tilde{g})^{\alpha} \cdot c(\tilde{f},\tilde{g})^{\beta} \cdot s(\tilde{f},\tilde{g})^{\gamma} \tag{13}$$

where:

α, β and γ – are parameters used to adjust the relative importance of the three components.

In order to simplify the expression, in [2] these values were taken as $\alpha = \beta = \gamma = 1$ and $C_3 = C_2/2$. This results in a specific form of the SSIM index [2]:

$$\mathrm{SSIM}(\tilde{f},\tilde{g}) = \frac{(2\mu_{\tilde{f}}\mu_{\tilde{g}} + C_1)(2\sigma_{\tilde{f}\tilde{g}} + C_2)}{(\mu_{\tilde{f}}^2 + \mu_{\tilde{g}}^2 + C_1)(\sigma_{\tilde{f}}^2 + \sigma_{\tilde{g}}^2 + C_2)} \tag{14}$$

Values of K_1 and K_2 in (3) are defined as $K_2 = 0.01$ and $K_2 = 0.03$.

The SSIM index and the three comparison functions – luminance, contrast, and structure – satisfy the following desirable properties:

1. Symmetry: $\mathrm{SSIM}(\tilde{f}, \tilde{g}) = \mathrm{SSIM}(\tilde{g}, \tilde{f})$. When quantifying the similarity between two signals, exchanging the order of the input signals should not affect the resulting measurement.
2. Boundedness: $\mathrm{SSIM}(\tilde{f}, \tilde{g}) \leq 1$. An upper bound can serve as an indication of how close the two signals are to being perfectly identical.
3. Unique maximum: $\mathrm{SSIM}(\tilde{f}, \tilde{g}) = 1$ if and only if $\tilde{f} = \tilde{g}$. The perfect score is achieved when the signals being compared are identical. In other words, the similarity measure should quantify any variations that may exist between the input signals.

The structure term of the SSIM index is independent of the luminance and contrast of the local patches, which is physically sensible because the change of luminance and/or contrast has little impact on the structures of the objects in the scene. Although the SSIM index is defined by three terms, the structure term in the SSIM index is regarded as the most important since variations in luminance and contrast of an image do not affect visual quality as much as structural distortions [7].

5. SSIM based trained filter

SSIM has been proven as a good alternative to MSE for objective image quality assessment [2, 11]. SSIM optimization will not change either the performance or complexity of the filtering process, but will be employed during the filter training process as depicted at Fig. 4.

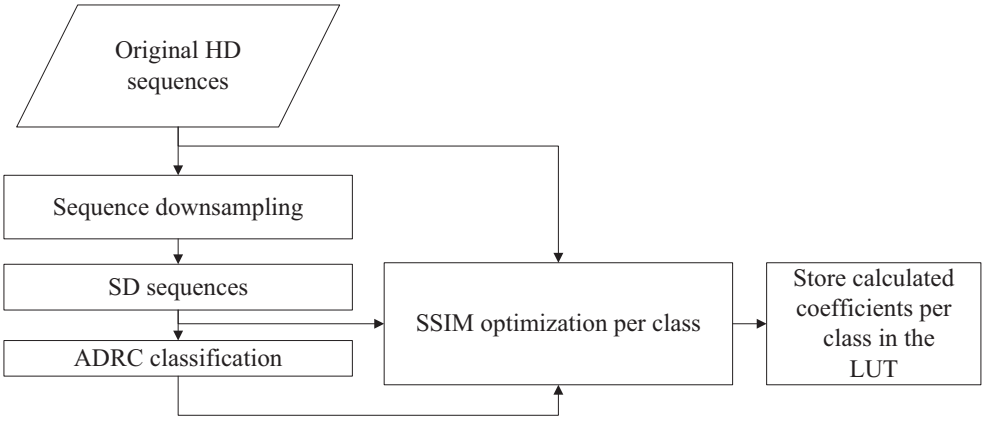

Fig. 4. Training process with SSIM optimization

Let F_{HD} – represents original HD data and F_{HI} – corresponding interpolated one from the set of SD pixels. The class c contains N samples during training. As a consequence (5), (6), (8), (12) are rewritten as:

$$\mu_{F_{\text{HI}}} = \overline{F_{\text{HI}}} = \frac{1}{N}\sum_{i=1}^{N} F_{\text{HI}i}, \quad \mu_{F_{\text{HD}}} = \overline{F_{\text{HD}}} = \frac{1}{N}\sum_{i=1}^{N} F_{\text{HD}i} \tag{15}$$

$$\sigma^2_{F_{\text{HI}}} = \frac{1}{N-1}\sum_{i=1}^{N}(F_{\text{HI}i} - \overline{F_{\text{HI}}})^2, \quad \sigma^2_{F_{\text{HD}}} = \frac{1}{N-1}\sum_{i=1}^{N}(F_{\text{HD}i} - \overline{F_{\text{HD}}})^2 \tag{16}$$

$$\sigma_{F_{\text{HI}}F_{\text{HD}}} = \frac{1}{N-1}\sum_{i=1}^{N}(F_{\text{HI}i} - \overline{F_{\text{HI}}})(F_{\text{HD}i} - \overline{F_{\text{HD}}}) \tag{17}$$

Finally, (14) becomes:

$$\text{SSIM}_c = \frac{(2\mu_{F_{\text{HI}}}\mu_{F_{\text{HD}}} + C_1)(2\sigma_{F_{\text{HI}}F_{\text{HD}}} + C_2)}{(\mu^2_{F_{\text{HI}}} + \mu^2_{F_{\text{FD}}} + C_1)(\sigma^2_{F_{\text{HI}}} + \sigma^2_{F_{\text{FD}}} + C_2)} \tag{18}$$

Instead of SSIM calculation over apertures from original and interpolated frames, the similarity index is calculated and optimized based on pixel values that belong to specified class c over the all training data set.

To maximize SSIM_c, the first derivative of $\frac{\partial(\text{SSIM}_c)}{\partial w_c(k)}$, $k \in [1 \ldots n]$ should be equal to zero:

$$\begin{cases} \dfrac{\partial(\text{SSIM}_c)}{\partial w_c(1)} = 0 \\ \dfrac{\partial(\text{SSIM}_c)}{\partial w_c(2)} = 0 \\ \ldots \\ \dfrac{\partial(\text{SSIM}_c)}{\partial w_c(n)} = 0 \end{cases} \tag{19}$$

Equation (19) represents the system of non-linear equations since it contains squared values $\mu^2_{F_{\text{HI}}}$ and $\sigma^2_{F_{\text{HI}}}$. To figure out optimal $w_c(k)$ values one of the numerical methods should be employed. The Newton method for nonlinear systems is a proven solution [9] that has common usage for solving systems of nonlinear equations and provides a fast convergence.

6. Result pixel truncation

The usage of the SSIM faced with a problem of a pixel value overshot as described in [12]. In order to address this issue, it is proposed to introduce truncation into (1) during result pixel value calculation:

$$F_{\mathrm{HI}}(i,j)_{\mathrm{trunk}} = \begin{cases} F_{\mathrm{SD\,MIN}}, & F_{\mathrm{HI}}(i,j) \le F_{\mathrm{SD\,MIN}} \\ F_{\mathrm{HI}}(i,j), & F_{\mathrm{SD\,MIN}} < F_{\mathrm{HI}}(i,j) < F_{\mathrm{SD\,MAX}} \\ F_{\mathrm{SD\,MAX}}, & F_{\mathrm{SD\,MAX}} \ge F_{\mathrm{SD\,MAX}} \end{cases} \tag{20}$$

where:

$F_{\mathrm{SD\,MIN}}$ and $F_{\mathrm{SD\,MAX}}$ – the lowest and the highest pixel values from the interpolation window accordingly.

The experiments showed that the interpolated pixel value $F_{\mathrm{HI}}(i, j)$ is never less than $F_{\mathrm{SD\,MIN}}$.

Originally, the pixel truncation allows a small improvement in the image quality – less than 3% [12]. The proposed truncation is hard – the pixel overshot is not allowed at all. In order to receive better quality, it is proposed to make this truncation soft – 1% of pixel overshot is allowed; in case of the maximum pixel value in the block equals to the 240, the maximum allowed pixel value is 242. As a consequence, (20) is changed to the following:

$$F_{\mathrm{HI}}(i,j)_{\mathrm{trunk}} = \begin{cases} F_{\mathrm{HI}}(i,j), & F_{\mathrm{HI}}(i,j) < 1.01 \cdot F_{\mathrm{SD\,MAX}} \\ F_{\mathrm{SD\,MAX}}, & F_{\mathrm{SD\,MAX}} \ge 1.01 \cdot F_{\mathrm{SD\,MAX}} \end{cases} \tag{21}$$

The original MSE-based interpolation filter requires the following memory to be allocated for the one calculation of the interpolated value within the 3 × 3 interpolation window:

– 512 bytes – for the LUT;
– 4 bytes – for the average of all the pixel values in the aperture (*AV* from (4));
– 9 bytes – to store pixel values ADRC code.

In total, the filter requires 525 bytes of memory except the one required to store input and output frames. The proposed algorithm requires two extra bytes of memory; it is less than 0.4% (527 vs. 525 bytes).

In order to receive one interpolated value, the original interpolation filter executes the following operations:

– 8 additions,
– 1 division,
– 9 comparisons,
– 9 multiplications.

The latency for the addition and comparison operations are equal for most count of the processors [13] and could be taken equal to 1. This parameter differs from processor to processor when the division operation is executed. The experiments were conducted on the Intel Core 2 processor (32 bit) with a latency value equal to 40 [13]. The result processor latency of the calculation equals 66. In order to fulfill (20), the filter additionaly requires 9 comparisons. As a consequence, the latency is increased by 13% – to 75. However, this change did not affect the processing time of the frame of the input sequence of HD format. This is explained by the processor's nature – the additional operations are executed in parallel. The majority of time is spent on the division operation.

7. Results

For objective evaluation, the MSE and SSIM between the original sequences and the result sequences processed on the down-scaled versions of the original sequences are calculated. The five sequences from the VQEG database were chosen for objective evaluation. The first four sequences have different compression levels, and the fifth is a raw sequence without compression. Table 1 shows the sequence characteristics.

Table 1

Test sequences and their characteristics

Sequence	Characteristics	Bitrate
New York 2	movement	1.5 Mb/s
Mobile & Calendar	color, movement	768 Kb/s
Football	color, movement	3 Mb/s
Sailboat	almost still	4.5 Mb/s
Suzie	skin color	–

Table 2 shows the MSE results of upscaling between the original sequences (in raw format) and the upscaled ones. A trained filter performed the upscaling with the original coefficients (MSE optimized) and coefficients obtained using the proposed approach (SSIM optimized with truncation). The upscaled sequence was processed on the down sampled versions of the original sequences at different bitrates.

Table 2

MSE scores for resolution upscaling using MSE and SSIM based ADRC

Sequence	ADRC (MSE-based)	ADRC (SSIM-based)
NewYork 2	174.54	179.50
Mobile & Calendar	819.15	846.78
Football	1415.31	1414.33
Sailboat	191.23	234.43
Suzie	13.79	14.07

Table 3 shows the SSIM results evaluation. The results of the proposed algorithm are shown in the right column. All results are captured for the first 60 frames of the test sequences. The perfect SSIM score is achieved when the signals being compared are identical (SSIM = 1).

According to Table 2, the proposed algorithm has shown the worst results for MSE calculations, except in one case. This degradation is expected the since MSE-optimization approach was replaced by the SSIM based version.

Table 3

SSIM scores for resolution upscaling using MSE and SSIM based ADRC

Sequence	ADRC (MSE-based)	ADRC (SSIM-based)
NewYork 2	0.912	0.924
Mobile & Calendar	0.293	0.293
Football	0.287	0.293
Sailboat	0.892	0.884
Suzie	0.946	0.958

The results of the SSIM metric calculation have shown better results for the proposed algorithm for three sequences (NewYork 2, Football, Suzie). Results are equals and worse in one case (Mobile & Calendar and Sailboat sequences respectively).

The SSIM and MSE results for the proposed algorithm are better for one sequence – Football. Usage of the SSIM optimization is beneficial when the filter is running against the sequence with average bitrates and movement of the scene. The usage of this approach over the sequences with small or no movement (Suzie and Sailboat) provides small benefits – the SSIM value is better for Suzie only while MSE values increased.

8. Conclusions

In this paper, the SSIM based coefficients calculation algorithm for the trained filter is presented. In the previous paper [12], it was proposed to introduce the truncation of the result pixel value. During the experiments, it was observed that the resulting pixel value is greater than the minimum pixel value from the interpolation window. Taking these observations into account allows for the elimination of the need to store the minimal pixel value in the memory. Additionally, it is proposed to make soft truncation and allow pixel overshot limited by 1% of the maximum pixel value.

The presented method has shown good results for various test video sequences. Since the simplest SSIM implementation was chosen for optimization and evaluation tasks, the observed results are similar for both algorithms. The received results allow for the focusing of further research on combining the SSIM and wavelet transformation to obtain better results for sequences with lower bitrates.

The proposed algorithm shows better results for the sequence with average bitrate and movement of the scene. This behavior is expected since the SSIM allows for focusing on the structural information from the scene rather than raw pixel values. The results prove that the algorithm is valuable for upscaling tasks and could be used in TV receivers as a replacement for the original MSE based version.

References

[1] Kondo T. et al., *Picture conversion apparatus, picture conversion method, learning apparatus and learning method*, US-patent 6, 323, 905, Nov. 2001.

[2] Zhao M. et al., *Content adaptive image de-blocking*, Proc. IEEE International Symposium on Consumer Electronics, Sep. 2004.

[3] Wang Z., Bovik A.C., Sheikh H.R., Simoncelli E.P., *Image Quality Assessment: From Error Visibility to Structural Similarity*, IEEE Transactions on Image Processing, Vol. 13, No. 4, Apr. 2004.

[4] Bovik A.C., *The Essential Guide to Image Processin*g, Academic Press, 2009.

[5] Wang Z., Lu L., Bovik A.C., *Video quality assessment based on structural distortion measurement*, Signal processing: Image communication, Vol. 19, No. 2, Feb. 2004, 121-132.

[6] Shao L., An *Overview and Performance Evaluation of Least Mean Square Trained Filters*, Technical Note PR-TN 2007/00230 Philips Research Europe.

[7] Wang Z., Bovik A.C., *A universal image quality index*, IEEE Signal Process. Lett., 9(3), 2002, 81-84.

[8] Lehmann T.M., Gonner C., Spitzer K., *Survey: Interpolation Methods in Medical Image Processing*, IEEE Transactions On Medical Imaging, Vol. 18, No. 11, November 1999.

[9] Ortega M., Rheinboldt W.C., *Iterative Solution of Nonlinear Equations in Several Variables*, Classics in Applied Mathematics, SIAM, 2000.

[10] Veres Z., Nakonechnyy A., *Classifications of methods for video up-scaling*, Methods and instruments of quality control, No. 22, 2009, 76-80.

[11] Veres Z., Nakonechnyy A., *Image quality metrics and ways for their improvement*, Computer Technology In Printing, No. 20, 2008, 69-81.

[12] Veres Z., Nakonechnyy A., *Adaptive Trained Filters With SSIM Optimization and Pixel Value Truncation for Video Upscaling*, 7th International Conference Intelligent Data Acquisition and Advanced Computing Systems, Berlin, 12–14 September 2013, 404-407.

[13] Torbj ¨orn Granlund, *Instruction latencies and throughput for AMD and Intel x86 processors*, (online) homepage: https://gmplib.org/~tege/x86-timing.pdf (date of access: 2014-04-27).

TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE
AUTOMATIC CONTROL | AUTOMATYKA
4-AC/2013

ALEX TORMÁSI*, LÁSZLÓ T. KÓCZY**

IDENTIFICATION OF THE INITIAL RULE-BASE OF A MULTI-STROKE FUZZY-BASED CHARACTER RECOGNITION METHOD WITH META-HEURISTIC TECHNIQUES

IDENTYFIKACJA POCZĄTKOWEJ BAZY REGUŁ W METODZIE ROZPOZNAWANIA WIELOLINIOWEGO PISMA ODRĘCZNEGO OPARTEJ NA LOGICE ROZMYTEJ Z WYKORZYSTANIEM TECHNIK METAHEURYSTYCZNYCH

A b s t r a c t

This paper summarizes the basic concept of the designed a fuzzy-based character recognition algorithm family and the results of the optimization of its rule-base with two various meta-heuristic methods, the Imperialist Competitive Algorithm and the bacterial evolutionary algorithm. The results are presented and compared with two other methods from literature after a short overview of the recognition algorithm.

Keywords: *fuzzy systems, character recognition, mate-heuristic optimization*

S t r e s z c z e n i e

W niniejszym artykule podsumowano podstawową koncepcję projektowania rodziny algorytmów rozpoznawania pisma odręcznego opartej na logice rozmytej oraz wyniki optymalizacji bazy reguł, z wykorzystaniem dwóch różnych metod metaheurystycznych: algorytmu ewolucyjnego ICA (*Imperialist Competitive Algorithm*) oraz ewolucyjnego algorytmu bakteryjnego. Przedstawiono krótkie podsumowanie algorytmu rozpoznawania pisma, a także wyniki porównawcze z dwoma innych metodami dostępnymi w literaturze.

Słowa kluczowe: *systemy rozmyte, rozpoznawanie pisma, optymalizacja metaheurystyczna*

* M.Sc. Alex Tormási, e-mail: tormasi@sze.hu, Department of Information Technology, Faculty of Engineering Sciences, Széchenyi István University, Györ.

** Prof. D.Sc. Ph.D. László T. Kóczy, Department of Automation, Faculty of Engineering Sciences, Széchenyi István University Györ; Department of Telecommunications and Media Informatics, Budapest University of Technology and Economics.

1. Introduction

At present, there are several handwriting solutions used which are based on various techniques. Most of these techniques are also implemented on smart phones and tablet PCs, but the default text entry methods are still physical or virtual keyboards and only a few devices come with a stylus to support handwriting recognition.

There are several reasons why handwriting recognition techniques are still not used as main text entry methods, such as usability issues caused by the latency of the recognition as a result of the use of complex mathematical transformations. The other reason is the lack of flexibility of recognition methods; there are many alphabets and various writing styles which make recognition more difficult, and most of the methods are able to recognize only a few of these.

The most important property of handwriting recognizers is accuracy; the user-acceptance threshold is a 97% recognition rate, determined by LaLomia [1]. Recognition methods usually apply various complex geometric transformation methods on the input to reach this level of accuracy; even recent devices do not have the necessary hardware capacity to provide near real-time recognition with such complexity of these methods.

Researchers must deal with all these problems to design an acceptable, wide-spread handwriting recognizer. According to previous results, the properties of fuzzy logic [2] may provide an acceptable solution for the mentioned problems. Meta-heuristic methods might be able to optimize the fuzzy rule-bases [3, 4] to reach the user-acceptance threshold of the recognition system.

After the introduction, the basic features and concept of the Fuzzy-Based Character Recognition (FUBAR) algorithm-family are overviewed [5, 6, 7]. In section 3, concepts of the investigated meta-heuristics are summarized. In section 4, results are presented and analyzed for the optimization of a multi-stroke FUBAR rule-base. Results are compared to the accuracy of other recognition methods and the possible orientations of future work are also investigated in the last section.

2. Basic Concept of Multi-Stroke Fuzzy-Based Character Recognition

2.1. Features and Limitations

The following important key features were kept in mind during the development of the recognition engine:

1. Accuracy: The algorithm has to reach an acceptable recognition rate or at least the same as other accepted methods.
2. Efficiency: The designed methods must fit user's requirements in response time and in resources of the currently used hardware. This means that complex geometrical transformations and other mathematical functions should be avoided.
3. Flexibility of the alphabet: The model of the alphabet must be easily modifiable to support various alphabets and context-sensitive recognition.
4. Learning: The designed system should be able to learn user-specific writing styles.

The characteristics and properties of fuzzy systems are able to satisfy all the considered features, which led to the use of the fuzzy inference method for the recognition method. Fuzzy-Based Character Recognizer (FUBAR) is a family of algorithms of various single-stroke and multi-stroke hand printed (handwritten, non-cursive, capital letters) character recognition engines. The basic concept of the designed method is shown in Fig. 1 [5].

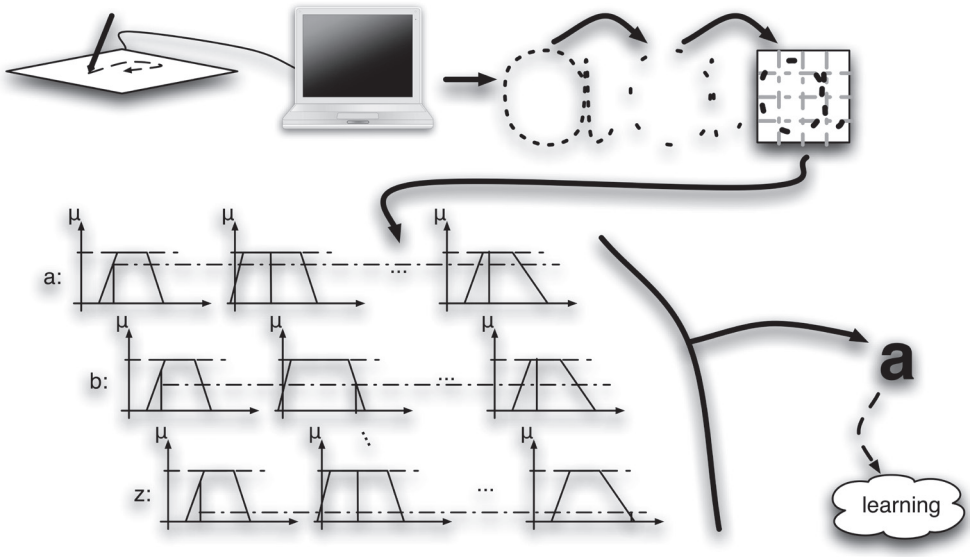

Fig. 1. The concept of FUBAR algorithms

The designed system is an online and personalized recognizer, which means that it processes digital ink and the recognition uses user-specific information.

2.2. Input Handling

The input signal of the algorithm consists of two-dimensional (x, y) coordinates in chronological order, representing the pen-movement (called (pen)stroke) as seen in Fig. 2.

In unistroke (or single-stroke) recognizers, letters are represented by one single stroke; in multi-stroke recognizers each symbol is represented by various numbers of strokes (sub-strokes). The FUBAR algorithms handle multi-strokes as one stroke by merging all the sub-strokes.

Usually, the received signal is non-continuous as a result of the bottlenecks of hardware, such as the bandwidth of interfaces and the available CPU resources. This causes information loss in recording the pen-movement; this information-loss causes difficulties in the processing because the positions of the missing coordinates are non-deterministic as seen in Fig. 3.

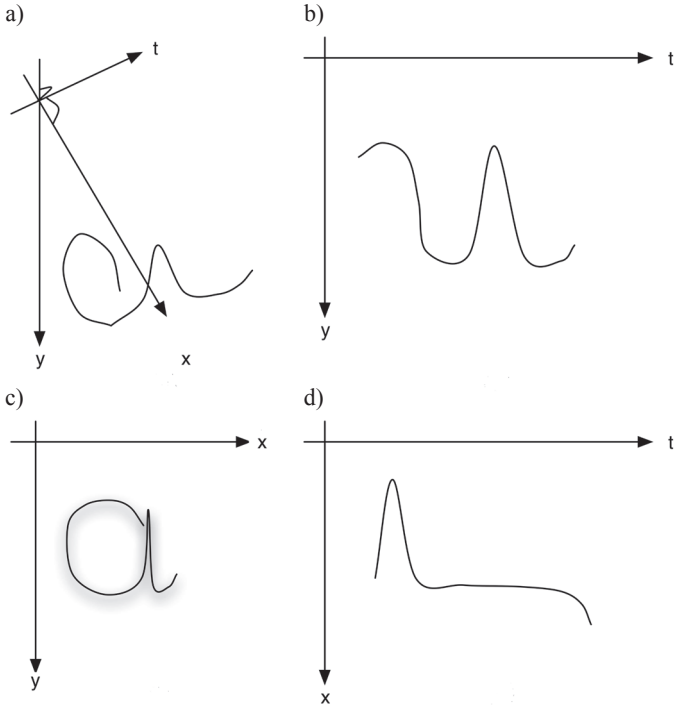

Fig. 2. A received 3-D stroke from various views: a) in 3-D, b) changes in the value of *y* by time, c) the 2-D projection, d) changes in the value of *x* by time

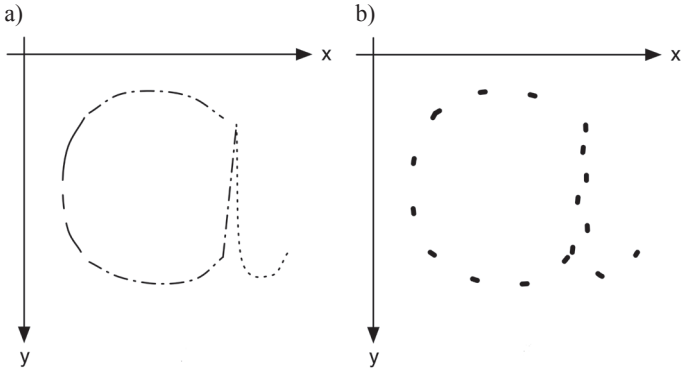

Fig. 3. Received stroke (a) and re-sampled stroke (b)

The received signal must be normalized for further processing and better recognition rate as seen in Fig. 3. In the FUBAR algorithm family, the points of the received signal are re-sampled, the points between a given (Euclidean-)distance from the reference point (the very first point of the stroke) are filtered out, the first and last points are always kept for reference. The following formula describes the re-sampling (filtering):

$$\underline{l'} = \{l_1\} \cup \left\{ l_j \middle| l \underset{\substack{j \in \{N_{n-1}-N_1\} \\ j>k}}{\arg\min_{j}} \left[\left| d\left(l_{k-1}, l_j\right) - \gamma \right| \right], k = \underset{i \in \{N_{n-1}-N_1\}}{\arg\min_{p}} \left[\left| d\left(l'_{i-1}, l_p\right) - \gamma \right| \right], i = \dim \underline{l'} \right\} \cup \{l_n\} \quad (1)$$

where:

- l – the received stroke,
- l' – the re-sampled stroke,
- l_j – a point in the stroke,
- γ – the threshold of distance between the kept points.

The re-sampling of the strokes also has an anti-aliasing property, which increases the degree of recognition as a result of the normalized stroke as seen in Fig. 4.

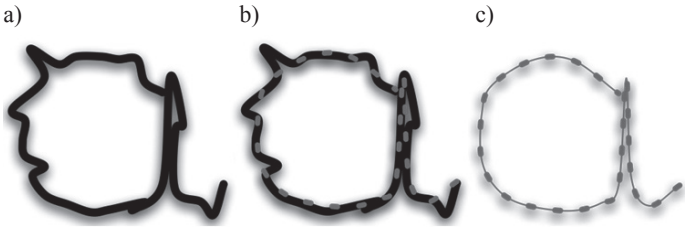

Fig. 4. Anti-aliasing properties of re-sampling: a) received stroke, b) received signal with selected points, c) selected points of the stroke

2.3. Character-Feature Extraction

FUBAR uses two kinds of stroke features for the recognition: 1) the width/height ratio of the stroke and 2) the average number of points in the rows and columns of the grid [5] drawn around the stroke.

The first member of the FUBAR family used crisp grid (classical grid with sharp borders) for the feature extraction, but the system reached a low average recognition rate as a result of the changing writing style of the test subjects. Some of the users started to write faster and use an italic writing style after creating a few samples, according to the collected data as seen in Fig. 5.

The sampled points of the strokes of oblique and normal characters could be located in completely different rows and columns of the grid. This caused huge overlap between the features of various letters. Other methods (like [18]) are rotating the input characters to avoid the negative effects of the italic writing style, but those methods are complex mathematical transformations, which would dramatically increase the computational complexity of the method.

Fuzzy grids were designed to resolve the problems caused by the italic writing style by the increased information provided by it. In fuzzy grids, the rows and columns of the grid are defined by fuzzy sets. It can be also considered as a transformation of the stroke into a fuzzy space.

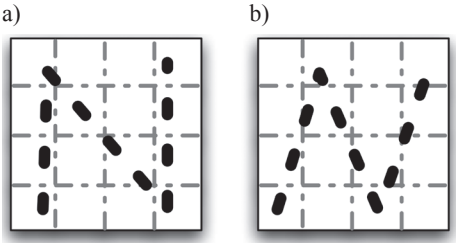

Fig. 5. Straight (a) and italic letters (b) in a grid

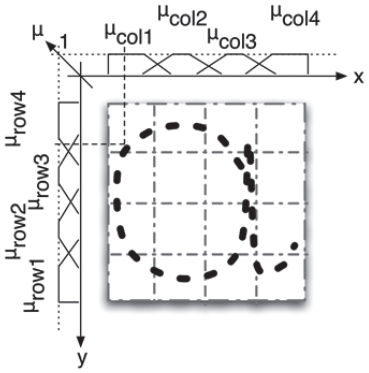

Fig. 6. Concept of the fuzzy grid

The optimal sizes of fuzzy grids were determined in [8, 9], which is 6 × 6 for single-stroke and 4 × 3 for multi-stroke letters. If the computational cost and recognition rate are both considered, then the optimal size of fuzzy grid for single-stroke letters is 6 × 4, while for the multi-strokes it does not change. The points in a fuzzy grid may belong to two different columns or rows at the same time with various membership values as seen in Fig. 6.

2.4. Inference

In the designed recognition engine a Takagi-Sugeno method [4] is used for inference. Each symbol in the alphabet is represented by a single rule associated to it. The input parameters of the rules are the features described in the previous sub-section. The output parameter of the rules is the degree of matching between the features of the input stroke and the stored rules as seen in Fig. 7, where R_i is the *i*-th rule, si is the degree of matching between the candidate stroke and the one represented by the *i*-th rule.

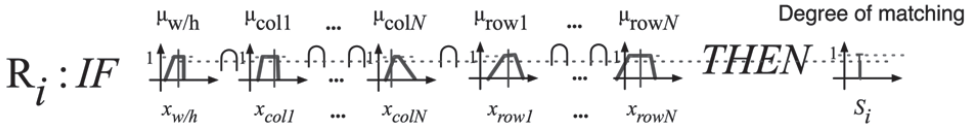

Fig. 7. A rule describing a letter

FUBAR returns with the character associated to the best matching rule (the one with the highest si value) after the rule evaluation phase. The initial rule-base was determined statistically from 60 samples per character, collected previously from test subjects.

There are two main ways to reduce the complexity of a rule-based fuzzy system: reducing the number of input parameters (antecedent) or reducing the number of rules evaluated. Hierarchical rule-bases are used to reduce the number of rules evaluated during the inference by partitioning the problem domain. Each partition contains a subset of the rules. Meta-rules are added to select which subset of the rules should be evaluated in a particular case; the input

parameters of these rules might be completely independent from the ones used in the original rules. Hierarchical rule-bases [10, 11] were used in some members of the FUBAR algorithm to decrease the computational complexity [12]. The average (fuzzified) number of points in the third row of a stroke was used as meta-rules in the single-stroke, while the number of strokes in the multi-stroke FUBAR.

3. The Applied Optimization Techniques

The initial rule-base for multi-stroke characters was determined the same way as described in [5] for single-stroke characters. The multi-stroke FUBAR algorithm reached 100% recognition rate for the 60 training sample per characters with the initial rule-base.

The fitness-function of each optimization algorithm was defined to maximize the average recognition rate for the training set. The optimization algorithms could not modify the initial rule-base because it had already reached the 100% accuracy. It does not mean that it can not be improved at all; it just reflects that the used initial rule-base is already optimal for the training set, so it can not be used during the meta-heuristic optimization and it must be changed to reach a lower recognition rate in order to start the optimization. The fuzzy sets in the initial rule-base for multi-stroke characters were modified with random values before the start of the optimization algorithms; the modified initial rule-base reached 79.17% recognition rate for the same training set and 74.7% for the validation sample set (120 samples per character) before the optimizations. The same randomly modified rule-base was used during each test.

Each entity (bacterium or country) in the algorithms represents a rule from the rule base. Each entity is coded into a vector, where the elements are the breakpoints of the fuzzy sets describing the antecedent of the rule. However, this technique requires validating the correctness of the new entities after any changes. The vector should be partially ordered for each represented membership function (it should maintain a trapezoid shape). Each algorithm had to be extended to use multiple populations (without migration), a further loop to say. These modified algorithms represent each letter by a single population, this was required to avoid the mixture between the data of various types of symbols.

Two different meta-heuristic algorithms were selected to optimize the modified initial rule-base, the Imperialist Competitive Algorithm (ICA) and the Bacterial Evolutionary Algorithm (BEA). Detailed description of Imperialist Competitive Algorithm can be found in [13] and of the Bacterial Evolutionary Algorithm in [14]. Maximizing the average recognition rate for the whole training set was used as a fitness function of the optimization algorithms, which means that the results were evaluated for all the strokes in the training set.

The ICA was inspired by politic and strategic processes instead of biological analogy. In the algorithm, countries are points in the search domain, the countries are determined randomly in the first step. Each country can be a conqueror or a colony, the strongest countries (those with the highest fitness values) are the conquerors.

The colonies are approaching the conqueror countries in the assimilation phase. The conqueror countries are fighting for the colonies, which are providing the possibility to increase their strength which is calculated by the aggregated fitness values of the empires and colonies. This assimilation phase is repeated until only one empire stands. The colonies can uprise against the conqueror countries (revolution phase), this step is included to prevent the algorithm from sticking in local optimums.

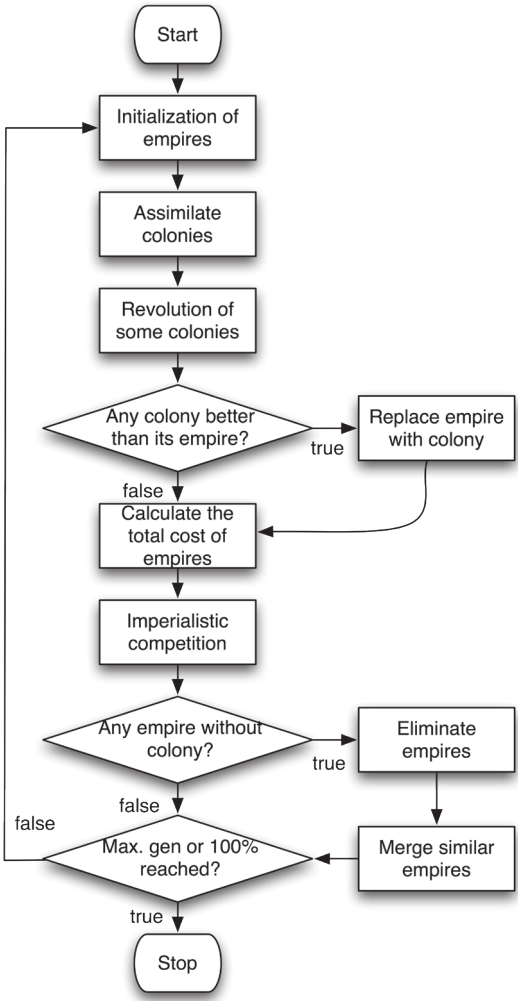

Fig. 8. The diagram of the Imperialist Competitive Algorithm

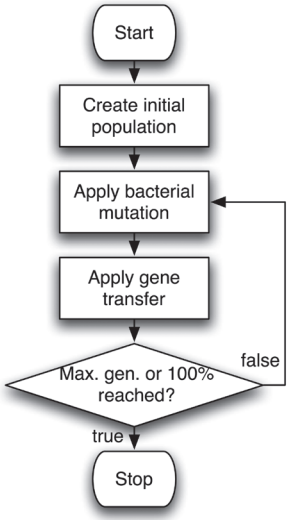

Fig. 9. The diagram of the Bacterial Evolutionary Algorithm

The BEA is based on the evolution process of bacteria. Each bacterium represents a point in the search domain. The first step of the algorithm is the bacterial mutation, which is applied to each bacterium individually. Each selected bacterium is cloned in this phase, then each allele of the clones are changed (only one at a time). The clone (or the original bacterium) with the best fitness value transfers its allele to the other clones.

The mutation phase is followed by the gene transfer (or infection) step. In this step, the bacteria population is divided into the groups of good and bad bacteria by their fitness values. A randomly selected good bacterium transfers a randomly selected allele to a randomly selected bad bacterium. All steps are repeated until the algorithm reaches the maximal number of generations.

4. Results of the Applied Optimization Techniques

The parameters of the bacterial evolutionary algorithm are the number of clones, the number of infections and the number of generations. The parameters of the imperialist competitive algorithm are the number of countries, the imperialist and the revolution factor.

The parameters of optimization algorithms were set according to preliminary results from formerly published works (where the single-stroke FUBAR was optimized), the ones with the best results are presented in this paper. The number of countries was constantly 120 and the number of imperialists was constantly 48 for the ICA, while the revolution factor was varying between 10 and 14 (steps were 0.2). All the parameters, number of clones, maximum generations and the number of infections were set to 10 in each BEA experiments.

The two best recognition rates achieved with the ICA for the validation set are 93.48% and 94.1%, as you may see in Fig. 10, the first average recognition rate was achieved with 120 countries, 48 imperialists and 12.4 revolution factors in 2 generations. The second result was reached by 120 countries, 48 imperialists and a revolution factor of 12.6 in 1 generation.

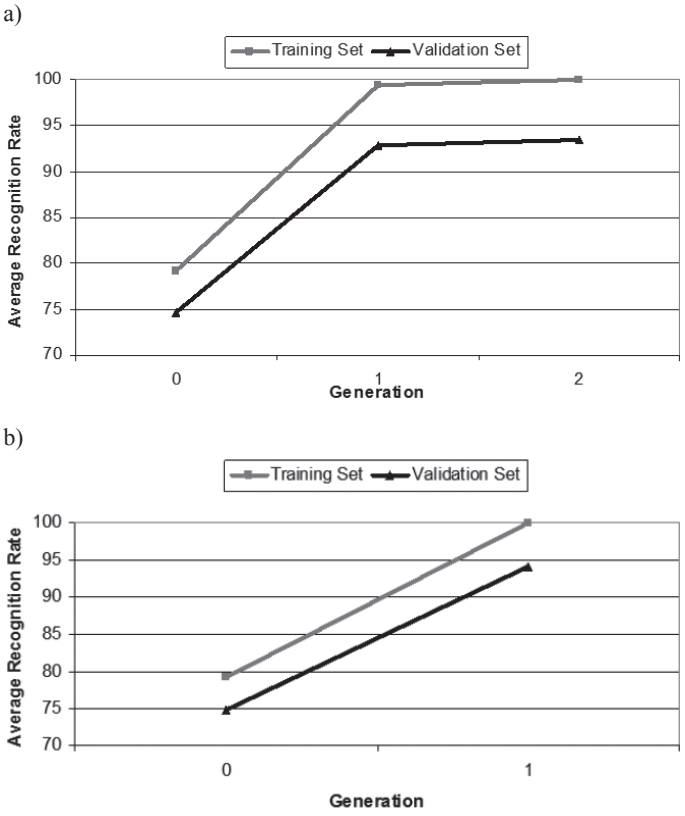

Fig. 10a) the second highest average recognition rate after the imperial competitive optimization, b) the highest average recognition rate after the imperial competitive optimization

The best two results of BEA are average recognition rates of 96.65% and 97.54% in 3 and 4 generations, with the same parameter settings, which are 10 clones, 10 infections and 10 generations as seen in Fig. 11.

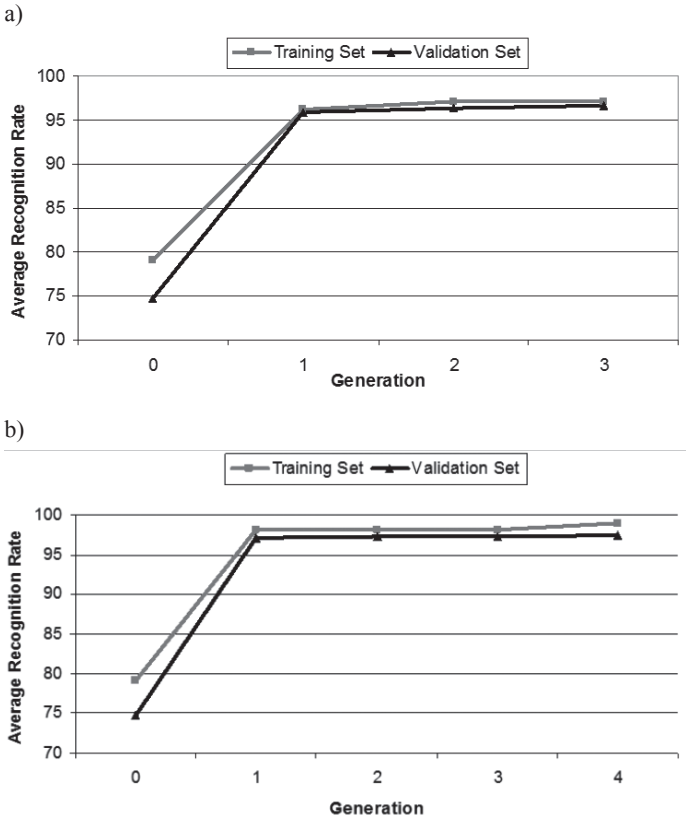

Fig. 11a) the second highest average recognition rate after the bacterial evolutionary optimization, b) the highest average recognition rate after the bacterial evolutionary optimization

The best accuracy reached by the ICA decreased the average error rate by only 10.6%. The highest average recognition rate of 97.54% was achieved by the BEA, which is a 62.72% decrease in the average error rate.

The average of the results from the experiments with the same parameters for BEA is 96.145%. There is no significant variance between the results for ICA with the previously described parameters; the average of the results is 93.67%.

It is also important to highlight the fact that only the ICA could reach 100% average recognition rate for the training set during the optimization in the first few (1–5) generations, while the bacterial evolutionary algorithm could not, the BEA finished only after the number of the generations reached the maximum. It is important, because it reflects that ICA is able to find (local)optimum solutions faster.

Despite the better convergence and lower resource consumption (computational cost), the imperialist competitive algorithm reached a lower increase in the recognition rate compared

to the bacterial evolutionary algorithm, which has a moderate convergence (compared to the imperialist competitive algorithm), a higher computational cost and a much higher increase in the accuracy.

The BEA may give better results, but it takes more time to compute the results. While BEA can process 0.0667 generations, ICA can evaluate 0.667 generations in a second. Both algorithms used 99% of 7 CPU cores from 8 during the experiments.

5. Conclusion and Future Work

The imperialist competitive algorithm had a better convergence during the optimization of the recognizer's rule-base, while the BEA reached the highest average accuracy (97.54%) for the validation sample set. The achieved average recognition rate reaches the 97% user acceptance threshold.

The most similar recognizer to the FUBAR is the Palm's Graffiti2 [15] and the $N recognizers [16]. The Graffiti2 is the multi-stroke version of the Palm's Graffiti unistroke recognizer [17]. The accuracy of the algorithm was determined at 86.03% by Költringer and Grechenig.

The $N recognizer is the multi-stroke successor of the $1 single-stroke recognizer [18], which was developed by J.O. Wobbrock, A.D. Wilson and Y. Li. The $N was developed by L. Anthony and J.O. Wobbrock.

The FUBAR algorithm with multi-stroke support reached a higher average recognition rate with 26 different characters after the optimization, compared to the Graffiti2 algorithm, which reached 86.03% accuracy and supports only 3 multi-stroke symbols, and compared to the $N algorithm, which achieved a 96.7% average recognition rate for 16 single-stroke symbols, as seen in Fig. 12.

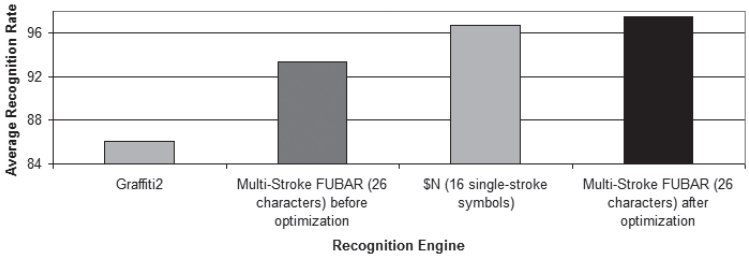

Fig. 12. Average recognition rates of various multi-stroke recognition engines

The in-depth analysis of these and other meta-heuristic techniques in model identification is important. Much more detail and further directions should be investigated such as a wider range of parameter values and as other type of rule-bases.

Future research includes the investigation of modeling symbols with multiple rules in the rule-base, this might increase the recognition rate by supporting various types of the same symbols. The increased number of the rules in the rule-base causes higher resource

requirements by the algorithm. The hierarchical structure of the fuzzy rule-base might reduce the resource requirements without any significant reduction in the average accuracy of the FUBAR algorithms.

The support of off-line character recognition will be included in the FUBAR algorithm family to extend the application areas of the method such as form processing.

Cursive handwriting is more general (especially in everyday uses), which makes its support a high priority task in the near future. The segmentation of characters must be also included in the FUBAR algorithms before the support of cursive writing recognition.

The FUBAR algorithm will be integrated into the HandSpy system [19], which is a collaborative environment for managing experiments in the cognitive processes in writing.

This paper was supported by the Hungarian Scientific Research Fund (Hungarian abbreviation: OTKA) K105529, K108405 and TÁMOP-4.2.2.A-11/1/KONV-2012-0012.
The used Meta-Heuristic algorithms were implemented by Márton Hevér as a part of his Master thesis (thesis advisor: Alex Tormási).

References

[1] LaLomia M.J., *User acceptance of handwritten recognition accuracy*, Companion Proc. CHI ’94, New York 1994, 107.
[2] Zadeh L.A., *Fuzzy sets*, Inf. Control, 83, 1965, 338-35.
[3] Mamdani E.H., Assilian S., *An experiment in linguistic synthesis with a fuzzy logic controller*, International Journal of Man-Machine Studies, Vol. 7, 1975, 1-13.
[4] Takagi T., Sugeno M., *Fuzzy identification of systems and its applications to modeling and control*, IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-15, 1985, 116-132.
[5] Tormási A., Botzheim J., *Single-stroke character recognition with fuzzy method*, New Concepts and Applications in Soft Computing SCI, Vol. 417, V.E. Balas et al. (eds.), 2012, 27-46.
[6] Tormási A., Kóczy T.L., *Improving the Accuracy of a Fuzzy-Based Single-Stroke Character Recognizer by Antecedent Weighting*, Proc. 2nd World Conference on Soft Computing, Baku 2012, 172-178.
[7] Tormási A., Kóczy T.L., *Fuzzy-Based Multi-Stroke Character Recognizer*, Preprints of the Federated Conference on Computer Science and Information Systems, Kraków 2013, 675-678.
[8] Tormási A., Kóczy T.L., *Comparing the efficiency of a fuzzy single-stroke character recognizer with various parameter values*, Proc. IPMU 2012, Part I. CCIS, Vol. 297, S. Greco et al. (eds.), 2012, 260–269.
[9] A. Tormasi, and Kóczy T.L., *Efficiency and accuracy analysis of a fuzzy single-stroke character recognizer with various rectangle fuzzy grids*, Proc. CSCS ’12, Szeged 2012, 54-55.
[10] Sugeno M., Griffin F.M., Bastian A., *Fuzzy hierarchical control of an unmanned helicopter*, Proc. IFSA ’93, Seoul 1993, 1262-1265.
[11] Kóczy T.L., Hirota K., *Approximate inference in hierarchical structured rule-bases*, Proc. IFSA ’93, Seoul 1993, 1262-1265.

[12] Tormási A., Kóczy T.L., *Improving the Efficiency of a Fuzzy-Based Single-Stroke Character Recognizer with Hierarchical Rule-Base*, Proc. 13th IEEE International Symposium on Computational Intelligence and Informatics, Óbuda 2012, 421-426.
[13] Atashpaz-Gargari E., Lucas C., *Imperialist Competitive Algorithm: An algorithm for optimization inspired by imperialistic competition*, Proc. 2007 IEEE Congress on Evolutionary Computation, 7, Singapore 2007, 4661-4666.
[14] Nawa N.E., Furuhashi T., *Fuzzy system parameters discovery by bacterial evolutionary algorithm*, IEEE Transactions on Fuzzy Systems, 7(5), 1999, 608-616.
[15] Költringer T., Grechenig T., *Comparing the Immediate Usability of Graffiti 2 and Virtual Keyboard*, Proc. CHI EA'04, New York 2004, 1175-1178.
[16] Anthony L., Wobbrock J.O., *A Lightweight Multistroke Recognizer for User Interface Prototypes*, Proc. GI 2010, Ottawa 2010, 245-252.
[17] Fleetwood M.D. et al., *An evaluation of text-entry in Palm OS – Graffiti and the virtual keyboard*, Proc. HFES'02, Santa Monica, CA, 2002, 617-621.
[18] Wobbrock J.O., Wilson A.D., Li Y., *Gestures without libraries, toolkits or training: A $1 recognizer for user interface prototypes*, Proc. UIST '07. ACM Press, New York 2007, 159-168.
[19] Monteiro C., Leal J.P., *Managing experiments on cognitive processes in writing with HandSpy*, Computer Science and Information Systems, Vol. 10, No. 4, Novi Sad 2013, 1747-1773.

TECHNICAL TRANSACTIONS | CZASOPISMO TECHNICZNE
AUTOMATIC CONTROL | AUTOMATYKA
4-AC/2013

MATEUSZ DZIEDZIC*, JANUSZ KACPRZYK**, SŁAWOMIR ZADROŻNY***

CONTEXTUAL BIPOLARITY AND ITS QUALITY CRITERIA IN BIPOLAR LINGUISTIC SUMMARIES

BIPOLARNOŚĆ KONTEKSTOWA I KRYTERIA JEJ JAKOŚCI W BIPOLARNYCH PODSUMOWANIACH LINGWISTYCZNYCH

Abstract

Bipolar linguistic summaries of data are assumed to be an extension of the 'classical' linguistic summarization, a data mining technique revealing complex patterns present in data in a human-consistent form. The extension proposal is based on the possibilistic interpretation of the 'and possibly' operator and introduced notion of context, which results in the introduction of the new 'contextual and possibly' operator. As the end user is expecting the most relevant summaries, ways of determining the quality of summary propositions (quality measures) needs to be developed. Here we focus on specific insights into the quality measures of proposed bipolar linguistic summaries of data and present some basic examples of their correctness and necessity of introduction.

Keywords: *bipolarity, context, linguistic summaries, quality criteria*

Streszczenie

Bipolarne podsumowania lingwistyczne są rozwinięciem „klasycznego" podejścia do lingwistycznego podsumowania danych, techniki ich eksploracji, której celem jest odszukanie obecnych w nich wzorców oraz zaprezentowanie ich w przystępnej dla człowieka formie. Propozycja rozwinięcia oparta jest na posybilistycznej interpretacji operatora „and possibly" oraz wprowadzonym pojęciu kontekstu, w wyniku czego zaproponowano nowy operator „contextual and possibly". Ponieważ użytkownik oczekuje prezentacji najbardziej trafnych podsumowań, konieczne jest zaproponowanie sposobów określania ich poprawności, zwanych dalej wskaźnikami jakości podsumowań. W niniejszym artykule skupiono się na szczegółowym spojrzeniu na wskaźniki jakości bipolarnych podsumowań danych i przedstawiono proste przykłady świadczące o ich poprawności oraz konieczności ich wprowadzenia.

Słowa kluczowe: *bipolarność, kontekst, kryteria jakości, podsumowania lingwistyczne*

* M.Sc. Mateusz Dziedzic, e-mail: mdziedzic@pk.edu.pl, Department of Automatic Control and Information Technology, Faculty of Electrical and Computer Engineering, Cracow University of Technology; Doctoral Studies, Systems Research Institute, Polish Academy of Sciences, Warsaw.

** Prof. D.Sc. Ph.D. Janusz Kacprzyk, Systems Research Institute, Polish Academy of Sciences, Warsaw; Department of Automatic Control and Information Technology, Faculty of Electrical and Computer Engineering, Cracow University of Technology.

*** D.Sc. Ph.D. Sławomir Zadrożny, Systems Research Institute, Polish Academy of Sciences, Warsaw School of Information Technology.

1. Introduction

The aim of data mining is to discover patterns in data in a form interesting and clear to the end user. A promising way to achieve this is to use (quasi) natural language. This has been a motivation for the linguistic data summaries introduced by Yager [11] and further developed by him [12] and other contributors, notably Kacprzyk and Zadrożny [8, 9].

Recently, an important role of bipolarity of user preferences, in particular in fuzzy linguistic querying [17], has been noticed. Its essence is in considering both positive and negative evaluations of objects in question which are not necessarily complements of each other. This entails the need to introduce logical connectives other than simple conjunction and disjunction.

An important and most interesting line of research focuses on the treatment of negative evaluations as obligatory while the positive evaluations as somehow secondary. This results in the introduction and study of the 'and possibly' logical connective [1]. Moreover, the concept of bipolar queries involving such a connective has been proposed [2] to better model user preferences as exemplified by the query 'Find an apartment, cheap and possibly located close to a station'.

In our previous papers [4, 5] we began to study if the relationship between fuzzy linguistic queries and linguistic data summaries may be adopted for bipolar queries. The results were positive and led us to the concept of bipolar linguistic summaries of data. In this paper, we focus on two quality criteria of such new type of linguistic summaries, introduced in [5] and referring to the notion of the context of a summary.

The structure of the paper is as follows. In Section 2, we briefly review the basics of fuzzy linguistic queries and 'classical' linguistic summaries, and introduce the notation to be used in the rest of the paper. In Section 3, we discuss the concepts of bipolar queries and bipolar linguistic summaries. Section 4 reports on the computational experiments focused on comparing different summary contexts and discusses the results obtained.

The preliminary version of this paper was presented at the FedCSIS'2013 conference [3].

2. Fuzzy linguistic queries and linguistic data summaries

2.1. Fuzzy linguistic queries

In classical query languages, such as SQL, preferences of users must be expressed precisely. However, due to the fact that their original form is a natural language expression, they are very often imprecise. For example, one may be concerned primarily with the cost while looking for an apartment to rent and express his or her preference as:

Find cheap apartments in Kraków (1)

In an approach, referred to here as fuzzy linguistic queries, such imprecise terms (e.g. cheap) are represented by fuzzy sets defined in the domains of respective attributes.

Usually, a dictionary of linguistic terms is assumed as a part of an implementation which contains predefined linguistic terms and corresponding fuzzy sets as well as terms defined by the users. Linguistic terms collected in a dictionary are a starting point to derive meaningful linguistic summaries of a database.

2.2. Linguistic summaries of data

As linguistic summaries we understand a (quasi) natural language sentences that grasp some characteristic features of data collected in a database. We use Zadeh's calculus of linguistically quantified propositions as the underlying formalism. The statement representing a linguistic summary points out some properties shared by a number of data items and the proportion of these data items is expressed using a linguistic quantifier. Yager [11, 12] first proposed the use of linguistically quantified propositions to summarise data in a user consistent way. That idea has been further developed, cf., e.g., Kacprzyk and Yager [6], and Kacprzyk, Yager and Zadrożny [7, 8].

Assuming $R = \{t_1, ..., t_n\}$ is a set of tuples (a relation) in a database, representing, for example, a set of employees; $A = \{A_1, ..., A_m\}$ is a set of attributes defining schema of the relation R, for example, Comfort, Price, No. of rooms etc., in a real estate database $A_j(t_i)$ denotes a value of attribute A_j for a tuple t_i), the linguistic summary of a set R is a linguistically quantified proposition which is an instantiation of one of the following abstract protoforms [16] of type I and type II, respectively:

$$Q_{t \in R} S(t) \tag{2}$$

$$Q_{t \in R}(U(t), S(t)) \tag{3}$$

(also denoted 'Q of R are S' and 'Q of U are S', respectively) then a linguistic summary is composed of the following elements:

- Summariser S which is a fuzzy predicate representing, for example, an expression an apartment is comfortable, formed using attributes of the set A;
- Qualifier U (optional) which is another fuzzy predicate representing, for example, a set of cheap apartments;
- Linguistic quantifier Q, for example, most expressing the proportion of tuples satisfying the summariser (optionally, among those satisfying a qualifier);
- Truth (validity) T of the summary, i.e. a number from [0, 1] expressing the truth of a respective linguistically quantified proposition in the scope of summarised data.

In Yager's original approach [11] the linguistic quantifiers are represented using Zadeh's definition [15]. A proportional, non-decreasing linguistic quantifier Q is represented by a fuzzy set in [0, 1] and $\mu_{Q(x)}$ states the degree to which the proportion of $100 \cdot x\%$ of elements of the universe match the proportion expressed by the quantifier Q. Thus, the formulas for the truth degree of type I and type II linguistic summaries, are respectively:

$$T\left(Q_{\{t \in R\}} S(t)\right) = \mu_Q \left[\frac{1}{n} \sum_{i=1}^{n} \mu_S(t_i)\right] \tag{4}$$

$$T\left(Q_{\{t\in R\}}\left(U(t),S(t)\right)\right)=\mu_Q\left(\frac{\sum_{i=1}^{n}\left(\mu_U(t_i)\wedge\mu_S(t_i)\right)}{\sum_{i=1}^{n}\mu_U(t_i)}\right) \tag{5}$$

3. Bipolar queries and bipolar linguistic summaries of data

3.1. Bipolar queries

In classical approaches to preferences modelling, notably in database querying, it is usually assumed that an alternative (tuple) is either accepted or rejected. However, the results of many studies, cf. [2], seem to suggest that the decision maker often comes up with somehow independent evaluations of the positive and negative features of the alternatives in question. This leads to a general concept of *bipolar query* against the database, which evaluation results in two degrees corresponding to the satisfaction of the positive and negative condition.

Most of the research on bipolar queries is focused on a special case where the positive and negative conditions are interpreted in an asymmetric way, cf. [2]. Namely, the latter is treated as a *constraint*, denoted *C*, which has to be satisfied, while the former plays the role of a mere *preference*, denoted ~*P*.

We follow the approach of Lacroix and Lavency [10], Yager [13, 14] and Bordogna and Pasi [1], adapted for database querying by Zadrożny and Kacprzyk [18], which combine both conditions using the 'and possibly' operator which aggregates their satisfaction degrees depending on the possibility of a simultaneous matching of both conditions.

Thus, the bipolar query's condition may be formally written as:

$$C \text{ and possibly } P \tag{6}$$

Such a bipolar query would be denoted (*C*, *P*) and interpreted as follows.

If there is a tuple which satisfies both conditions, then and only then is it actually *possible* to satisfy both of them and each tuple of data has to do so, which turns (*C*, *P*) into the conjunction of both conditions, $C \wedge P$. On the other hand, if there is no such a tuple, then condition *P* is ignored.

As an example, consider the query:

Find apartments that are comfortable *and possibly* cheap (7)

to a databases shown in Tab. 1–2. Let us assume that apartments priced below 250k PLN (250 000 of Polish zlotys) are in general considered as *cheap* (to a high degree, whenever we refer to satisfying a fuzzy condition or matching a summary, we mean to a high degree) and Comfort higher than 7.5 means a comfortable apartment. Then, it is possible to find in the *Sample real estate database 1* (Tab. 1) an apartment that is both *comfortable* and *cheap*, e.g. apartments No. 1 or 2, which, as stated earlier, turns the example query (7) into:

Find apartments that are *comfortable* and *cheap*

and returns tuples No. 1 and 2.

Let us contrast this to the result against the *Sample real estate database 2* (Tab. 2), where it is not possible to satisfy both conditions as there is no *cheap* apartment, which, as a result, ignores condition *P* and returns all four tuples.

The matching degree of the (*C*, *P*) query against a tuple *t* may be formalised as [10]:

$$T\left(C\left(t\right)\textit{and possibly}\,P\left(t\right)\right)=C\left(t\right)\wedge\left(\exists s\left(C\left(s\right)\wedge P\left(s\right)\right)\Rightarrow P\left(t\right)\right) \tag{8}$$

Table 1

Sample real estate database 1

Apt. No.	Comfort [1–10]	Price [k PLN]	Rooms
1	9.0	250	4
2	8.6	229	3
3	9.3	895	8
4	9.1	830	9

Table 2

Sample real estate database 2

Apt. No.	Comfort [1–10]	Price [k PLN]	Rooms
1	9.3	455	6
2	8.9	429	5
3	9.3	895	8
4	9.1	830	9

3.2. Bipolar linguistic summaries

Let us start with a brief remainder of the point of departure of our work. In [4] we proposed the concept of a bipolar linguistic summary using, as a starting point the concept of a bipolar query and a link between fuzzy linguistic queries and 'classical' linguistic summaries pointed out earlier in our works. We to follow the same concept with bipolar queries and bipolar linguistic summaries.

The earlier proposed interpretation of '*C* and possibly *P*' expressed by (8) makes this proposition true for a tuple *t* only if either of two conditions hold:

1) *t* satisfies both conditions *C* and *P*, or

2) *t* satisfies *C* and there is no tuple in the whole database which satisfies both conditions.

Thus, the the straightforward use of the formula (6) to instantiate the summariser in the linguistic summary protoforms (2)–(3) and its interpretation via (8) does not make much sense. Namely, the expression (6) may be appropriate to represent preferences of the user (as it is exemplified by query (7)) who does not know if there is an interference between conditions *C* and *P* with respect to the content of the queried database. However, e.g., the following proposition:

Most apartments (in the database) are *comfortable* and possibly *cheap*

is rather meaningless in the role of a summary as 'the system' knows if there is or isn't such an interference and should incorporate this information into results.

The main idea behind the interpretation of the bipolar linguistic summaries proposed by us is to relate the '*C* and possibly *P*' to *a part* the database of instead of the whole database. Let us consider the following example summary:

Most apartments (in the database) are *confortable* and possibly,
with respect to aparments of *similar size* (assuming ± 1 room), *cheap*

An apartment matches such a summary if:
1) it is comfortable and cheap, or
2) it is comfortable and there is no other apartment of similar size that is both comfortable and cheap.

Taking this into consideration and assuming that 430–460 k PLN priced apartments this time could be considered *cheap* (to much lover degree than ≤ 250 k PLN obviously), the above summary is true (still to a high degree) for both *Sample real estate databases 1* and 2 (Tab. 1–2).

A characteristic feature of such a summary is the use of a summariser employing an extended version of the 'and possibly' operator, which we will refer to as the 'contextual and possibly' operator. This operator may be expressed as:

$$C \text{ and possibly } P \text{ with respect to } W \tag{9}$$

For the purposes of contextual bipolar queries (and, thus, bipolar linguistic summaries) the predicates C and P should be interpreted as the required and desired conditions, respectively, while the predicate W denotes the *context* in which the possibility of satisfying both C and P will be assessed, separately for each tuple. Then, the formula (9) is interpreted as:

$$\begin{aligned} &T\left(C(t) \textit{ and possibly } P(t) \textit{ with respect to } W(t)\right) \\ &\quad = C(t) \wedge \left(\exists s\left(W(t,s) \wedge C(s) \wedge P(s)\right) \Rightarrow P(t)\right) \end{aligned} \tag{10}$$

Our preliminary computational experiments show that usage of standard De Morgan triples $(\wedge_{\min}, \vee_{\max}, \neg)$, $(\wedge_{\Pi}, \vee_{\Pi}, \neg)$ and $(\wedge_{L}, \vee_{L}, \neg)$ with t- and s-norms: Minimum and Maximum; Product and Probabilistic sum; and Łukasiewicz's t- and s-norm, respectively), both with the S- and R-implication, in (10) may lead to somehow counter-intuitive results in terms of bipolar queries evaluation. Thus we use the $(\wedge_{\min}, \vee_{\max}, \neg)$ De Morgan triple and Goguen R-implication which turns (10) into:

$$\begin{aligned} &T\left(C(t) \textit{ and possibly } P(t) \textit{ with respect to } W(t)\right) = \\ &\begin{cases} \min\left(C(t), 1\right) & \textit{for } \exists WCP(t) = 1 \\ \min\left(C(t), \min\left(1, \dfrac{P(t)}{\exists WCP(t)}\right)\right) & \textit{otherwise} \end{cases} \end{aligned} \tag{11}$$

where $\exists WCP(t)$ denotes $\max_{s \in R} \min(W(t, s), C(s), P(s))$ in this context.

3.3. Summary context quality criteria

In [5] we stated that the quality of the summary context W itself and the whole implication premise in (10) (i.e. $\exists WCP(t)$) have to be considered when measuring the quality of the bipolar linguistic summaries.

If P and/or W are such that the $\exists WCP(t)$ is true to a very low or a very high degree for *most* of tuples, then the summarizer (9) does not make much sense even if the truth value of the whole summary is high. This is due to the behaviour of the bipolar query 'C and possibly P' which turns into C and $C \wedge P$, respectively, when the truth degree of $\exists WCP(t)$ (i.e. $\exists_{s \in R} C(s) \wedge P(s)$) is close to 0 and close to 1.

The introduction of the context W partially alleviates this problem, but W has to be chosen carefully. If for most t's there does not exist $s \in R/\{t\}$ such that $W(t, s)$, then the premise of the implication is most often false and the summary is true for any P.

We propose a solution to those problems in a form of quality measures incorporating following linguistically quantified propositions:

$$Q_{t \in R} \exists_{s \in R \setminus \{t\}} W(t, s) \tag{12}$$

$$Q_{t \in R} \exists_{s \in R \setminus \{t\}} C(s) \wedge P(s) \wedge W(t, s) \tag{13}$$

Namely, if the truth of (12) for a summary is too small (lower than some threshold value), then such a summary should be discarded. Also, if the truth of (13) is too small (too close to 0; lower than the third threshold value) or too high (too close to 1; larger than the second threshold value) then the summary also shouldn't be taken into account. Obviously, if the first threshold is violated, then the third one also is. On the other hand, even if the first threshold is satisfied, the summary may still fail to satisfy thresholds two or three and should be discarded.

Tuple t is excluded from the range of the existential quantifiers in (12)–(13) as if the only tuple related via W with t is only t itself, then, naturally, the resulting summary is of no interest.

4. Computational examples and discussion

As a confirmation of the need to introduce quality criteria (12) and (13), let us consider sample databases similar to those used in section 3.2, shown in Tab. 3–6 (Tab. 3–5 presents data to which bipolar linguistic queries should not be used and need to be filtered out, whereas Tab. 6 shows an example in a favour of them – the proposed quality criteria support this).

For simplicity we limited components of a summary to:

- One quantifier *most* defined as unitary quantifier: $most(x) = x$;
- Two fuzzy predicates based on Comfort and Price attributes, instantiated for clarity by only one linguistic value each: *comfortable* and *cheap*, respectively, shown on Fig. 1;
- (Crisp) context W = *similar size* true iff $|Rooms(t) - Rooms(s)| \leq 1$.

Table 3

Sample real estate database 3

Apt. No.	Comfort [1]	Price [k PLN]	Rooms
1	7.8	213	3
2	8.1	349	5
3	7.6	629	7
4	8.0	712	9

Table 4

Sample real estate database 4

Apt. No.	Comfort [1–10]	Price [k PLN]	Rooms
1	8.2	475	5
2	7.8	489	5
3	8.9	629	7
4	7.5	655	8

Table 5

Sample real estate database 5

Apt. No.	Comfort [1–10]	Price [k PLN]	Rooms
1	8.0	205	3
2	7.5	213	3
3	7.8	245	5
4	8.3	249	6

Table 6

Sample real estate database 6

Apt. No.	Comfort [1–10]	Price [k PLN]	Rooms
1	8.0	205	3
2	7.5	213	3
3	7.8	345	5
4	8.3	359	6

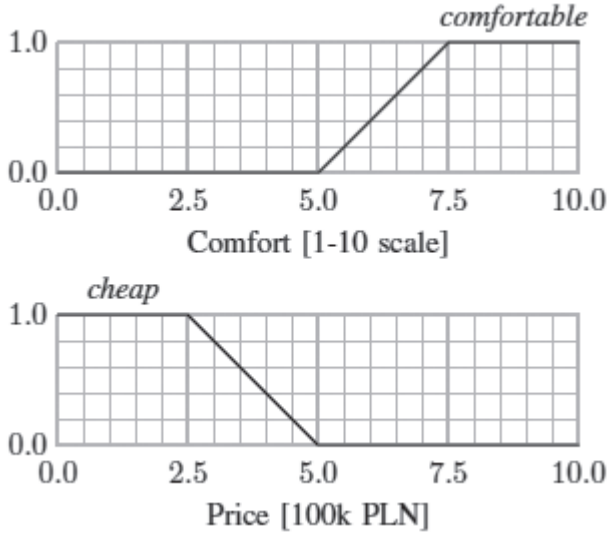

Fig. 1. Membership functions of *comfortable* (condition *C* – upper plot) and *cheap* (condition *P* – lower plot) predicates based on Comfort and Price attributes, respectively

All of which taken together result in an example summary considered:

Most apartments (in the sample database) are comfortable and possibly, with respect to apartments of similar size, cheap (14)

4.1. Discussion of results

A brief analysis of results obtained for individual tables shows that:

- Tab. 3 – truth value of the summary (14) equals 1.00 because of the wrongly chosen context *W*, where the majority of tuples (in this example all of them) are in its own 'neighborhood'. This context should not be taken into consideration during summarisation.
  Situation identified by a low value of criterion (12): 0.00.
- Tab. 4 – truth value of the summary (14) equals 0.86 making it a feasible component of the returned set of summaries despite that apartments No. 3 and 4 are not *cheap* and the remaining two are considered *cheap* in a very low degree (0.10 and 0.04). This summary should be replaced by: '*Most* apartments (in the database) are *comfortable*', '*Most* apartments (in the database) are *comfortable* and *NOT cheap*' or even '*Almost none of* apartments (in the database) are *cheap*'.
  Identified by a low value of criterion (13): 0.0.
- Tab. 5 – truth value of the summary (14) equals 1.00 regardless to the context *W* (all apartments are *cheap* to the same, or at least similar, degree), therefore the summary itself should be replaced by a 'classical' one such as '*Most* apartments (in the sample database) are *comfortable* and *cheap*'.
  Identified by a high value of criterion (13): 1.00.
- Tab. 6 – truth value of the summary (14) equals 0.98, which suggest that this summary could be considered as a component of the resulting set of summaries. Value 1.00 of criterion (12) indicates the properly chosen context and value 0.80 of criterion (13) (rather high) suggests caution in selecting this summary, however competitive 'classical' summary receives lover truth value: 0.80.

We focused here on showing the benefits of using 'contextual and possibly' operator in the scope of linguistic data summarization, presenting both a theoretical and semantic justification of this concept, and intuitively appealing examples of correctness of the proposed criteria.

Contextual bipolarity employed in the summaries manifests itself by determining dynamically for each tuple, a context *W*, in which the possibility of matching the conditions *C* and *P* simultaneously is checked. This property of bipolar summaries offers the possibility to discover more interesting patterns in data utilising a very human-specific bipolar approach to preferences.

Examples clearly argue in favour of introduced additional quality criteria (measures) and confirm that their help to distinguish interesting summaries from among all with high truth values. Additional studies are needed in order to clearly determine the best summaries, yet already the results are promising.

5. Concluding remarks

Preliminary computational results of bipolar linguistic summaries proposed in [4], demonstrated the need for new quality criteria to determine the true quality of the summary. In [5] we introduced two of them which have been studied deeper here. The results presented in section 4 show that proposed criteria fulfil their role and help select bipolar linguistic

summaries that are valuable and interesting for an end user. Due to a conceptual character of the paper and for simplicity, we have considered here only qualifier-free bipolar linguistic summaries, however, they may be extended to include qualifiers in an obvious way.

Future works in this subject will mainly cover combining introduced criteria with other known quality measures, in order to determine a single value of the quality of the linguistic summary on one hand, and for evaluating and selecting linguistic summaries by means of heuristic methods on the other hand.

Mateusz Dziedzic contribution is supported by the Foundation for Polish Science under International PhD Projects in Intelligent Computing. Project financed from The European Union within the Innovative Economy Operational Programme (2007–2013) and European Regional Development Fund.

References

[1] Bordogna G., Pasi G., *Linguistic aggregation operators of selection criteria in fuzzy information retrieval*, International Journal of Intelligent Systems, Vol. 10(2), 1995, 233-248.

[2] Dubois D., Prade H., *Bipolarity in flexible querying*, [in:] Andreasen T., Motro A., Christiansen H., Larsen H.L. (Eds.), FQAS 2002, volume 2522 of LNAI, Springer-Verlag, Berlin, Heidelberg, 2002, 174-182.

[3] Dziedzic M., Kacprzyk J., Zadrożny S., *On some quality criteria of bipolar linguistic summaries*, Proceedings of 2013 Federated Conference on Computer Science and Information Systems (FedCSIS), Kraków (Poland), IEEE, 2013, 643-646.

[4] Dziedzic M., Kacprzyk J., Zadrożny S., *Towards bipolar linguistic summaries: a novel fuzzy bipolar querying based approach*, Proceedings of 2012 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Brisbane (Australia), IEEE, 2012, 1-8.

[5] Dziedzic M., Kacprzyk J., Zadrożny S., *Bipolar linguistic summaries: a novel fuzzy querying driven approach*, Proceedings of 2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), Edmonton (Canada), IEEE, 2013, 1279-1284.

[6] Kacprzyk J., Yager R.R., *Linguistic summaries of data using fuzzy logic*, International Journal of General Systems, Vol. 30, (2), 2001, 133-154.

[7] Kacprzyk J., Yager R.R., Zadrożny S., *A fuzzy logic based approach to linguistic summaries of databases*, International Journal of Applied Mathematics and Computer Science, Vol. 10, (4), 2000, 813-834.

[8] Kacprzyk J., Zadrożny S., *On a fuzzy querying and data mining interface*, Kybernetika, Vol. 36, (6), 2000, 657-670.

[9] Kacprzyk J., Zadrożny S., *Linguistic database summaries and their protoforms: towards natural language based knowledge discovery tools*, Information Sciences, Vol. 173, (4), 2005, 281-304.

[10] Lacroix M., Lavency P., *Preferences: Putting more knowledge into queries*, Proceedings of the 13 International Conference on Very Large Databases, Brighton (UK), 1987, 217-225.

[11] Yager R.R., *A new approach to the summarization of data*, Information Sciences, Vol. 28, 1982, 69-86.
[12] Yager R.R., *On linguistic summaries of data*, [in:] Frawley W., Piatetsky-Shapiro G., (Eds.), Knowledge Discovery in Databases, AAAI/MIT Press, 1991, 347-363.
[13] Yager R.R., *Higher structures in multi-criteria decision making*, International Journal of Man-Machine Studies, Vol. 36, 1992, 553-570.
[14] Yager R.R., *Fuzzy logic in the formulation of decision functions from linguistic specifications*, Kybernetes, Vol. 25, (4), 1996, 119-130.
[15] Zadeh L., *A computational approach to fuzzy quantifiers in natural languages*, Computers and Mathematics with Applications, Vol. 9, 1983, 149-184.
[16] Zadeh L., *From search engines to question answering systems – the problems of world knowledge relevance deduction and precisiation*, [in:] E. Sanchez, (Ed.), Fuzzy Logic and the Semantic Web, Elsevier, 2006, 163-210.
[17] Zadrożny S., Kacprzyk J., *Bipolar queries: An approach and its various interpretations*, [in:] Carvalho J.P., Dubois D., Kaymak U., Sousa J.M.C. (Eds.), Proceedings of the Joint 2009 IFSA/EUSFLAT Conference, Lisbon (Portugal), 2009, 1288-1293.
[18] Zadrożny S., Kacprzyk J., *Bipolar queries: An aggregation operator focused perspective*, Fuzzy Sets and Systems, Vol. 196, 2012, 69-81.

CONTENTS

TREŚĆ

Architektura
Gulnara Abdrasilova
Jacek Borowski
Wojciech Brzezowski
Talgat Maramowic Bajtasow
Martin Baloga
Boris Bielek
Milian Bielek
Andrew Borgart
Wacław Celadyn
Ewa Cichy-Pazder
Jan Cremers
Peter Čuka
Alina Drapella-Hermansdorfer
Peter Durica
Ursula Eicker
Raimund Fein
Halina Garbalińska
Elżbieta Greiner-Wrona
Frank Grümmer
Elżbieta Horszczaruk
Artur Jasiński
Elżbieta Kaczmarska
Tomasz Kisilewicz
Sabina Kuc
Jan Kurek
Teresa Kusionowicz
Wojciech Leśnikowski
Adam Lisik
Madelyn Marrero
Elena Meldjieva
Hanna Michalak
Anna Mitkowska
Yelena Morozowa
Zbigniew Myczkowski
Irena Niedźwiecka-Filipiak
Lucyna Nyka
Bonawentura Pawlicki
Halina Petryszyn
Katarzyna Pluta
Elżbieta Przesmycka
Zbigniew Radziewanowski
Janusz Rębielak
Liudmila Ruban
Zbigniew Rykiel
Zoltan Sadovsky
Sarah Shangereevna Sadykova
Grażyna Schneider-Skalska
Anna Sedlakova
Ingrid Senitkova
Alena Sičáková
Jolanta Tamošaitienė
Joanna Tymkiewicz
Zuzanna Vranayova
Ewa Węcławowicz-Gyurkovich

Budownictwo
Jan Adamczyk
Toth Balazs
Renata Bášková
Miram Bell
Jan Benčat
Zinoviy Blikharskyy
Istvan Bodi
Marek Bomberg
Claudio Bori
Włodzimierz Brząkała
Ugis Cabulis
Marek Cała
Dorota Chwieduk
Felix Darve
Janusz Datta
Andrzej Dobrucki
Dan Dubina
Pavol Ďurica
Grzegorz Dzierżanowski
Andrzej Flaga
Jean Francois-Destrebeg
Josef Eberhardsteiner
Maria Fiertak
Sławomir Francik
Anders Gade
Christoph Gehlen
Jacek Gołaszewski
Bronisław Gosowski
Marian Giżejowski
Gunnar Grün
Lajos Gyorgy
Petr Hájek
Stephen Hall
Dariusz Heim
Markku Heinisuo
Zdzisław Hejducki
Rüdiger Höffer
Jens Holger
Bożena Hoła
Elżbieta Horszczaruk
Irena Ickiewicz
Roman Jankowiak
Karol Jarmai
Jerzy Jasieńko
Eugeniusz Kałuża
Oleg Kapliński
Tadeusz Kasprowicz
Dušan Katunský
Davorin Kereković
Hyunsun A. Kim
Agnieszka Klemm
Katarzyna Klemm
Michał Knauff
Andrzej Kosecki
Mária Kozlovská
Ryszard Kowalczyk
Andrzej Kowalski
Jan Kubica
Bernd Kugelmann
Stanisław Kuś
Valeriy Kuznetsov
Krystyna Kuźniar
Vincent Kvočak
Lesław Kwaśniewski
Lech Lichołaj
Martin Lopušniak
Andrzej Łapko
Dariusz Łydźba
Josef Machaček
Arkadiusz Madaj
Anton Antonowicz Malinowski
Piotr Małkowski
Herbert Mang
Mariusz Maślak
Marco Menegotto
Peter Mesároš
Pavel Mišek
Leszek Mikulski
Vít Motyčka
Piotr Noakowski
Henryk Nowak
Zygmunt Orłowski
Wiesław Ostachowicz
Jerzy Pasłowski
Hartmud Pasternak
Elżbieta Pilecka
Edyta Plebankiewicz
Marek Piekarczyk
Vyacheslav Pisarev
Marianna Polak
Mieczysław Połoński
Elżbieta Radziszewska-Zielina
Charles Rodrigues
Guido de Roeck
Janusz Rymsza
Zoltan Sadovsky
Iouri N. Semenov
Johann Sienes
Zbigniew Sikora
Tomasz Siwowski
Dariusz Skorupka
Marek Słoński
Leslie Smith
Anna Sobótka
Giovanni Solari
Marcela Spišáková
Zuzana Struková
Jiri Strasky
Janusz Szwabowski

Lucjan Ślęczka
Jan Ślusarek
Magdalena Rucka
Barbara Tal-Figiel
Alena Tažiková
Jolanta Tamošaitienė
Yannis Tripanagnostopoulos
Dieter Ungermann
Ivan Vaníček
Ioanis Vayas
Martins Vilnitis
Tatjana Vilutiene
Frantisek Wald
Jurij Warecki
Jerzy Wekezer
Krzysztof Wilde
Andrzej Winnicki
Wojciech Włodarczyk
Bogdan Wolski
Walter Wuwer
Raul Zaharia
Jacek Zimny
Izabela Zimoch
Alphose Zingoni
Henryk Zobel
Adam Zybura
Jan Żmuda
Krzysztof Żółtkowski

Chemia
Elcio Balota
Paolo Battaini
Marek Berezowski
Karlis Bormanis
Marek Czernicki
Wojciech Fabianowski
Elżbieta Filipek
Michael Fuentes
Aneta Głuszek
Muneaki Ishijima
Janusz Konstanty
Teresa Kowalska
Kazimierz Kozłowski
Dharam Persaud-Sharma
Roman Petrus
Anne Marie Pourcher
Daniela Plachá
Jeffrwet E. Rosen
Mirosław Szukiewicz
Małgorzata Iwona Szynkowska
Akinola Tolueke
Anish Tuteja
Julian Vysochanskii
Krystyna Wieczorek-Ciurowa
Marcin Wilk
Valery Zazhigalov

Mechanika
Dalibor Barta
Jan Chłopek
Leszek Dobrzański
Jan Dusza
Jan Dutkiewicz
Krzysztof Fitzner
Klaus Friedrich
Stanisław Guzowski
Andrzej Grzybek
Hakan Hallberg
Herbert Ipser
Jacek Kaczmarek
Janusz Kowal
Marek Kozień
Marek Książek
Jan Łuczko
Andrzej Mruk
Antoni Neyman
Józef Nizioł
Paweł Piec
Tadeusz Piechowiak
Ladislaw Plist
Etienne Savary
Andrzej Seweryn
Bohumir Strandel
Jan Szybka
Antoni Świć
Wiktor Taranenko
Krzysztof Zima
Lidia Żakowska

Nauki Podstawowe
Andrzej Bargieła
Jan Błachut
Roman Bobryk
Marian Bubak
Alwarez Castillo
Antoni Dawidowicz
Leszek Demkowicz
Stanisław Drożdż
Dawid Edwin
Heiner Gonska
Robertas Grigalaitis
Frank Gruemmer
Aleksander Iwanow
Navnit Jha
Andrzej Kucharski
Tadeusz Lesiak
Jakub Lewicki
Barbara Majcher-Iwanow
Marian Nowak
Jerzy Pamin
Vera Pavlovna-Plaksina
Ioan Rasa
Lucyna Rempulska
Bolesław Szafirski
Marcin Szpyrka
Natalja Tropina
Yulian Vysochansky
Eugeniusz Wachnicki
Edward Wantuch
Karol Życzkowski

Środowisko
Armando Carlos
Witold Chmielnicki
Krzysztof Gaska
Rejesh Kanna
Piotr Kowalik
Andrzej Królikowski
Hans Kupfersberger
Hans-Peter Nachtnebel
Jan Pawełek
Helena Raclavska
Jan Rak
Henryk Sobczuk
Józef Trela
Stanisław Witczak
Eugeniusz Zawisza