Advances in Intelligent Systems and Computing 285

Radek Silhavy
Roman Senkerik
Zuzana Kominkova Oplatkova
Petr Silhavy
Zdenka Prokopova *Editors*

Modern Trends and Techniques in Computer Science

3rd Computer Science On-line Conference 2014 (CSOC 2014)

Springer

Advances in Intelligent Systems and Computing

Volume 285

Series editor

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

For further volumes:
http://www.springer.com/series/11156

About this Series

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

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

Radek Silhavy · Roman Senkerik
Zuzana Kominkova Oplatkova
Petr Silhavy · Zdenka Prokopova
Editors

Modern Trends and Techniques in Computer Science

3rd Computer Science On-line Conference 2014 (CSOC 2014)

Springer

Editors
Radek Silhavy
Roman Senkerik
Zuzana Kominkova Oplatkova
Petr Silhavy
Zdenka Prokopova
Faculty of Applied Informatics
Tomas Bata University in Zlín
Zlín
Czech Republic

ISSN 2194-5357 ISSN 2194-5365 (electronic)
ISBN 978-3-319-06739-1 ISBN 978-3-319-06740-7 (eBook)
DOI 10.1007/978-3-319-06740-7
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2014937958

Printed on acid-free paper

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

Preface

This book constitutes the refereed proceedings of the *3rd Computer Science On-line Conference 2014 (CSOC 2014)*, held in April 2014.

We are promoting new scientific conference concepts by organizing the CSOC conference. Modern online communication technology improves the traditional concept of scientific conferences. It brings equal opportunity to participate to all researchers around the world. Therefore, CSOC 2014 uses innovative methodology to allow scientists, postdocs, and doctoral students to share their knowledge and ideas online.

The conference intends to provide an international forum for the discussion of the latest high-quality research results in all areas related to Computer Science. The topics addressed are the theoretical aspects and applications of Artificial Intelligences, Computer Science, and Software Engineering. The authors present new approaches and methods to real-world problems, and particularly, exploratory research that describes novel approaches in their field.

The 53 papers presented in the proceedings were carefully reviewed and selected from 95 paper submissions. At least two respected reviewers reviewed each submission. 58 % of all submissions were received from Europe, 27 % from Asia, 7 % from America, and 5 % from Africa.

The editors believe that readers will find the proceedings interesting and useful for their own research work.

March 2014

Radek Silhavy
Roman Senkerik
Zuzana Kominkova Oplatkova
Petr Silhavy
Zdenka Prokopova

Program Committee

Program Committee Chairs

Zdenka Prokopova, Ph.D., Associate Professor, Tomas Bata University in Zlín, Faculty of Applied Informatics, e-mail: prokopova@fai.utb.cz
Zuzana Kominkova Oplatkova, Ph.D., Associate Professor, Tomas Bata University in Zlín, Faculty of Applied Informatics, e-mail: kominkovaoplatkova@fai.utb.cz
Roman Senkerik, Ph.D., Associate Professor, Tomas Bata University in Zlín, Faculty of Applied Informatics, e-mail: senkerik@fai.utb.cz
Petr Silhavy, Ph.D., Senior Lecturer, Tomas Bata University in Zlín, Faculty of Applied Informatics, e-mail: psilhavy@fai.utb.cz
Radek Silhavy, Ph.D., Senior Lecturer, Tomas Bata University in Zlín, Faculty of Applied Informatics, e-mail: rsilhavy@fai.utb.cz

Program Committee Members

Dr. Luis Alberto Morales Rosales, Head of the Master Program in Computer Science, Superior Technological Institute of Misantla, Mexico.
Mariana Lobato Baes, M.Sc., Research Professor, Superior Technological of Libres, Mexico.
Abdessattar Chaâri, Professor, Laboratory of Sciences and Techniques of Automatic control and Computer engineering, University of Sfax, Tunisian Republic.
Gopal Sakarkar, Shri Ramdeobaba College of Engineering and Management, Republic of India.
V. V. Krishna Maddinala, Assistant Professor, GD Rungta College of Engineering and Technology, Republic of India.
Anand N. Khobragade, Scientist, Maharashtra Remote Sensing Applications Centre, Republic of India.
Abdallah Handoura, Assistant Professor, Computer and Communication Laboratory, Telecom Bretagne—France.

Technical Program Committee Members

Eric Afful Dazie
Michal Bliznak
Donald Davendra
Radim Farana
Zuzana Kominkova Oplatkova
Martin Kotyrba
Erik Kral
David Malanik
Michal Pluhacek
Zdenka Prokopova
Martin Sysel
Roman Senkerik
Petr Silhavy
Radek Silhavy
Jiri Vojtesek
Eva Volna

Organizing Committee Chair

Radek Silhavy, Ph.D., Senior Lecturer, Tomas Bata University in Zlín, Faculty of Applied Informatics, e-mail: rsilhavy@fai.utb.cz

Conference Organizer (Production)

OpenPublish.eu s.r.o.
Svornosti 1908
755 01 Vsetin
Czech Republic
Web: www.openpublish.eu
E-mail: csoc@openpublish.eu

Contents

Part I
Artificial Inteligence

Intelligence Digital Image Watermark Algorithm Based on Artificial Neural Networks Classifier

Cong Jin and Shu-Wei Jin

Abstract An intelligence robust digital image watermarking algorithm using artificial neural network (ANN) is proposed. In new algorithm, for embedding watermark, the original image first is divided into some $N_1 \times N_2$ small blocks, different embedding strengths are determined by RBFNN classifier according to different textural features of every block after DCT. The experimental results show that the proposed algorithm are robust against common image processing attacks, such as JPEG compression, Gaussian noise, cropping, mean filtering, wiener filtering, and histogram equalization etc. The proposed algorithm achieves a good compromise between the robustness and invisibility, too.

Keywords Digital watermarking · ANN · Classification · Textural feature · Invisibility · Robustness

1 Introduction

Multimedia data is easily copied and modified, so necessity for copyright protection is increasing. Digital watermarking [1, 2] has been proposed as the technique for copyright protection of multimedia data. A watermarking algorithm requires both invisibility and robustness, which exist in a trade-off relation. Many watermarking systems based on artificial neural network (ANN) have been already proposed [3–5]. In [3], watermarking is viewed as a form of communications. A blind watermarking algorithm is presented using Hopfield neural network to calculate the capacity of watermarking. Zhang et al. [4] used a back-propagation

C. Jin (✉) · S.-W. Jin
School of Computer Science, Central China Normal University,
Wuhan 430079, People's Republic of China
e-mail: jincong@mail.ccnu.edu.cn

S.-W. Jin
e-mail: dede91@mail.ccnu.edu.cn

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_1,

(BP) ANN to learn the characteristics of relationship between the watermark and the watermarked image. The false decoded rate of the watermark can be greatly reduced by the trained ANN. In [5], ANN is used to model human visual system (HVS) [6] and an image-adaptive scheme of watermarking strength decision method is presented for the watermarking on DCT coefficients. Robust digital image watermarking schemes based on ANN are proposed in [5], too. We know that the robustness of watermarks depends on the watermark embedding strength. For transform domain watermark, if selecting higher watermark embedding strength, it has good robustness and bad invisibility; and if selecting lower watermark embedding strength, it has bad robustness and good invisibility. Therefore, when we choose a watermark embedding strength, we should consider a trade-off between invisibility and robustness.

Different from existing methods, in this paper, we don't discuss the capacity of watermarking (topic of [3]). In [4], the properties of HVS don't be considered. Although the good watermark performance was obtained by using ANN in [5], the watermark extraction processes are not blind with referring to the original image. In this paper, we propose a new watermarking algorithm based on ANN. After dividing the original image into some non-overlapping small blocks of size $N_1 \times N_2$, ANN is used to determine different watermarking embed strengths according to different textural features of every block.

2 ANN Classifier

Multi-layer perceptron has many advantages such as simple structure [7], rapid training process and good extend ability etc. So it can be applied to many fields, especially, in the aspects of pattern classification and function approach.

ANN-based classifiers can incorporate both statistical and structural information and achieve better performance than the simple minimum distance classifiers [8]. An ANN possesses the following characteristics [9]: (1) It is capable of inferring complex non-linear input–output transformations. (2) It learns from experience, so, has no need for any a priory modeling assumptions. Therefore, based on the advantages of ANN, multi-layered ANN, usually employing the back propagation (BP) algorithm, is also widely used in digital watermarking [10]. In this paper, an effective classification approach using ANN according to the image textural features is proposed.

In this paper, we let ANN have four layers including an input layer with five neurons, first hidden layer with six neurons, second hidden layer with eight neurons and one output neuron. Where, the number of the input neuron is determined by dimension number of textural feature vector. The output of ANN is the maximum watermarking strength. All the input features and output have to be normalized so that they always fall within a specified range before training. The inputs and the output are normalized to fall in the interval $[-1, 1]$ and $[0, 1]$, respectively.

3 Select Textural Features

Let I be a gray-level original image of size $M_1 \times M_2$, and we divide I into non-overlapping $N_1 \times N_2$ blocks, J,

$$J = \bigcup_{i=1}^{M_1/N_1} \bigcup_{j=1}^{M_2/N_2} J_{(i,j)} \tag{1}$$

where, $M_1 > N_1$ and $M_2 > N_2$.

For one non-overlapping block $J_{(i,j)}$ in I, we will extract five-dimensional vectors of features. We use five features: one from the image histogram (mean gray level) and four from the gray level co-occurrence matrix (contrast, entropy, correlation, and angular second moment). Specific definitions of these features are given below.

1. First-order gray level parameter

   In this category, the feature is derived from the gray level histogram. It describes the first-order gray level distribution without considering spatial interdependence. As a result, it can only describe the echogenicity of texture as well as the diffuse variation characteristics within the every $N_1 \times N_2$ block. The feature selected from this category is:

(a) The mean gray level (*MGL*)

$$MGL = \frac{1}{N_1 \times N_2} \sum_{m=1}^{N_1} \sum_{n=1}^{N_2} J_{(i,j)}(m,n),\ 1 \le i \le \frac{M_1}{N_1},\ \ 1 \le j \le \frac{M_2}{N_2} \tag{2}$$

where $J_{(i,j)}(m, n)$ is the gray level of $N_1 \times N_2$ block $J_{(i,j)}$ at pixel (m, n).

2. Second-order gray level features

   This category of features describes the gray level spatial inter-relationships and hence, they represent efficient measures of the gray level texture homogeneity. These features can be derived using several approaches such as first-order gradient distribution, gray level co-occurrence matrix, edge co-occurrence matrix, or run-length matrix. Because the gray level co-occurrence matrix seems to be a well-known statistical technique for feature extraction, we will generate these features from the gray level co-occurrence matrix. We generated a gray-level co-occurrence matrix from every $N_1 \times N_2$ block. The formal definition of this matrix is as follows:

$$Co(s,t) = \frac{1}{N_1 \times N_2} cardinality\{((k_1,k_2),\ (l_1,l_2)) \in J_{(i,j)} : \quad |k_1 - l_1| = dx, \quad |k_2 - l_2| = dy \tag{3}$$

$$sign((k_1 - l_1) \cdot (k_2 - l_2)) = sign(dx \cdot dy),\ J_{(i,j)}(k_1, k_2) = s,\ J_{(i,j)}(l_1, l_2) = t\Big\} \tag{4}$$

where $Co(s, t)$ is the gray level co-occurrence matrix entry at gray levels s and t, and (dx, dy) is a prescribed neighborhood definition taken in case to be (0, 12) representing an axial neighborhood definition. In other words, the entry (s, t) of this matrix describes how often the two gray levels s and t are neighbors under the given neighborhood definition. Note that this definition does not discriminate between negative and positive shifts and hence, the co-occurrence matrix is expected to be symmetric using this definition. The four features are defined as follows

(b) The contrast (*CON*)

$$CON = \sum_{s,t} (s - t)^2 \cdot Co(s, t) \tag{5}$$

The contrast feature is a difference moment of the $Co(s, t)$ matrix and is a standard measurement of the amount of local variations presented in an image. The higher the values of contrast are, the sharper the structural variations in the image are.

(c) The entropy (*ENT*)

$$ENT = -\sum_{s,t} Co(s,t) \cdot \log(Co(s,t)) \tag{6}$$

(d) The correlation (*COR*)

$$COR = \frac{\sum_{s,t} stCo(s, t) - m_x \cdot m_y}{S_x \cdot S_y} \tag{7}$$

where,

$$m_x = \sum_s s \sum_t Co(s, t),\ m_y = \sum_t t \sum_s Co(s, t), \quad S_x^2 = \sum_s s^2 \sum_t Co(s, t) - m_x^2,$$
$$S_y^2 = \sum_t t^2 \sum_s Co(s, t) - m_y^2$$

(e) The angular second moment (*ASM*)

$$ASM = \sum_{s,t} (Co(s, t))^2 \tag{8}$$

Angular second moment gives a strong measurement of uniformity. Higher non-uniformity values provide evidence of higher structural variations.

4 Embedding and Extraction of Watermark

4.1 Select DCT Coefficients for Inserting Watermark

Our goal is to embed the roust watermark into to the DCT frequency bands of I. Before the embedding procedure, we need to transform the spatial domain pixels into DCT domain frequency bands. After we perform the $N_1 \times N_2$ block DCT on I, we get the coefficients in the frequency bands, F,

$$F = DCT(I), \quad \text{and} \quad F = \bigcup_{i=1}^{M_1/N_1} \bigcup_{j=1}^{M_2/N_2} F_{(i,j)} \tag{9}$$

For one non-overlapping block (i, j) in I, the resulting $N_1 \times N_2$ DCT bands $J_{(i,j)}$ can be represented by

$$F_{(i,j)} = \bigcup_{k=1}^{N_1 \times N_2} \{F_{(i,j)}(k)\}, \quad 1 \le i \le \frac{M_1}{N_1}, \quad 1 \le j \le \frac{M_2}{N_2} \tag{10}$$

$F_{(i,j)}(k)$ are zigzag ordered DCT coefficients. Afterwards, we are able to embed the watermark in the DCT domain. Assuming that the binary-valued watermark to be embedded is W, having size $S_1 \times S_2$. A pseudo-random number traversing method is applied to permute the watermark to disperse its spatial relationship. With a pre-determined key, key_0, in the pseudo-random number generating system, we have the permuted watermark W^P,

$$W^P = permute(W, key_0). \tag{11}$$

And we use W^P for embedding the watermark bits into the selected DCT frequency bands. The human eye is more sensitive to noise in lower frequency components than in higher frequency ones. However, the energy of most natural images is concentrated in the lower frequency range, and watermark data in the higher frequency components might be discarded after quantization operation of lossy compression. In order to invisibly embed the watermark that can survive lossy data compressions, a reasonable trade-off is to embed the watermark into the middle-frequency range of the image. In this paper, the middle band of the DCT domain is chosen. But, with regard to JPEG, casting watermarks in the middle band of the $N_1 \times N_2$ block-based DCT domain is more robust.

In the case of embedding a watermark based on $N_1 \times N_2$ block DCT, only $k = \left\lfloor N_1 \times N_2 \times \frac{S_1 \times S_2}{M_1 \times M_2} \right\rfloor$ coefficients for each $N_1 \times N_2$ block $F_{(i,j)}$ will be used for the watermark embedding. First, the DCT coefficients of each block are reordered in zig-zag scan. Then, the coefficients from the $(L + 1)$th to the $(L + k)$th, *i.e.* a sequence of values $F_{(i,j)} = \{F_{(i,j)}(L+1), F_{(i,j)}(L+2), \ldots, F_{(i,j)}(L+k)\}$ are taken according to the zig-zag ordering of the DCT spectrum, where the first L coefficients are skipped for embedding the middle band.

4.2 Embedding the Watermark

The amount of modification each coefficient undergoes is proportional to the magnitude of the coefficients itself as expressed by

$$\hat{F}_{(i,j)}(L+u) = \begin{cases} F_{(i,j)}(L+u) + \alpha_{(i,j)}(u) \cdot \left|F_{(i,j)}(L+u)\right| \cdot w^P_{(i,j)}(u), & \text{if } w^P_{(i,j)}(u) = 1 \\ F_{(i,j)}(L+u), & \text{otherwise} \end{cases} \tag{12}$$

$$u = 1, 2, \ldots, k; \quad 1 \le i \le \frac{M_1}{N_1}, \quad 1 \le j \le \frac{M_2}{N_2}$$

where, $\alpha_{(i,j)}(u)$ indicates uth parameter controlling the watermarking strength of block (i, j) given by the ANN. $\hat{F}_{(i,j)}$ is then inserted back into the image in place of $F_{(i,j)}$, and we obtain the watermarked DCT coefficients, F',

$$F' = \bigcup_{i=1}^{M_1/N_1} \bigcup^{M_2/N_2} \hat{F}_{(i,j)} \tag{13}$$

After performing inverse DCT on F', we get the watermarked image, $\hat{I}$,

$$\hat{I} = inverse\_DCT(F') \tag{14}$$

4.3 Extracting the Watermark

To extract a watermark in a possibly watermarked image I_w, firstly, I_w is decomposed into non-overlapping $N_1 \times N_2$ blocks and the DCT is computed for each blocks. k DCT coefficients are selected by using the same method with

embedding watermark for each block. Another necessary procedure is to calculate the suggested strength $\alpha_{(i,j)}(u)$ of the corresponding block (i,j). Moreover, we let

$$Th_{(i,j)} = \frac{1}{k}\sum_{u=1}^{k} F_{(i,j)}(L+u) \tag{15}$$

we are able to extract the permuted watermark according to the following equation

$$W'^{P}_{(i,j)}(u) = \begin{cases} 1, & if \ \left|\hat{F}_{(i,j)}(L+u) - F_{(i,j)}(L+u)\right| \geq \alpha_{(i,j)}(u) \cdot Th_{(i,j)} \\ 0, & otherwise \end{cases} \tag{16}$$

$$W'^{P} = \bigcup_{i=1}^{M_1/N_1} \bigcup_{j=1}^{M_2/N_2} W'^{P}_{(i,j)}(u), \quad u = 1, 2, \ldots, k \tag{17}$$

We use key_0 in Eq. (11) to acquire the extracted watermark W' from W'^{P},

$$W' = inverse\_permute\,(W'^{P}, key_0) \tag{18}$$

5 Experimental Results

5.1 Experiments

To evaluate the performance of the proposed watermarking algorithm, a set of experiments is performed under the following conditions.

The "*Flower*" image with size $M_1 \times M_2 = 512 \times 512$, is used the original image, which is shown in Fig. 1. We have the embedded watermark with size $S_1 \times S_2 = 128 \times 128$, shown in Fig. 2. Size $N_1 \times N_2$ of small block is 8×8. Hence, the number of bits to be embedded in one 8×8 non-overlapping block is $k = 128^2/512^2 \cdot 64 = 4$, $L = 6$, and $F_{(i,j)} = \{F_{(i,j)}(7),\ F_{(i,j)}(8),\ F_{(i,j)}(9),\ F_{(i,j)}(10)\}$. Which is shown in Fig. 3. After watermark embedding in the DCT domain, we take the inverse DCT, and obtain the watermarked image. We measure the invisibility of the watermarked images and the robustness of the extracted watermarks against various attacks. The Peak Signal to Noise Ratio (PSNR) and the Normal Correlation (NC) shown in Eqs. (19) and (20) are used to measure the invisibility and the robustness, respectively.

$$PSNR = 10\log_{10}\frac{M_1 \times M_2 \times \sum_{i=1}^{M_1}\sum_{j=1}^{M_2}\left(\hat{I}(i,j)\right)^2}{\sum_{i=1}^{M_1}\sum_{j=1}^{M_2}\left(I(i,j) - \hat{I}(i,j)\right)^2} \tag{19}$$

Fig. 1 Original image with size 512 × 512

Fig. 2 Watermark image with size 128 × 128

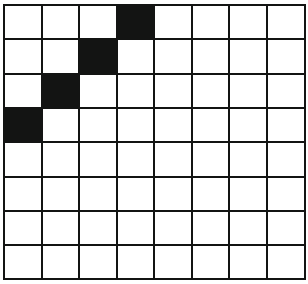


Fig. 3 4 bits are embedded in every 8 × 8 block

$$NC = \frac{\sum_{i=1}^{S_1} \sum_{j=1}^{S_2} W(i,j)W'(i,j)}{S_1 \times S_2} \times 100\ \% \tag{20}$$

where W and W' are original watermark and extracted watermark, respectively.

The ANN training procedure is terminated either when the training error is less than 10^{-4} or 2,000 iterations. The training error is the mean square error. The learning rate and momentum term were chosen as 0.1–0.15 and 0.8–0.9, respectively. The initial weight values, momentum term, and learning rate are the parameters of BP algorithm. The most commonly used winner-takes-all method was used for selecting the ANN output. The hidden and output neuron functions were defined by the logistic sigmoid function $f(x) = 1/(1 + \exp(-x))$.

The watermarked image using proposed algorithm is depicted in Fig. 4.

Fig. 4 The watermarked image, PSNR = 44.63

Table 1 PSNR of watermarked images

Image	Flower	Baboon	Lena	Goldhill	Peppers
PSNR	44.63	45.31	46.82	43.94	45.58

5.2 Results

5.2.1 Invisibility

The goal is to measure the invisibility of the watermarked images. PSNR is used to measure the invisibility of the watermarked image, where the higher PSNR, the more transparency of the watermark. Basics image "*Flower*", another four images are chosen as the tested images. The four images of size 512 × 512 are *Baboon*, *Lena*, *Goldhill*, and *Peppers*. By using the proposed algorithm, we see that the watermark is almost invisibility to the human eyes. Table 1 shows PSNR of tested images after embedding watermark.

5.2.2 Robustness

1. Wiener filtering Wiener filtering is a kind of the common signal attack for digital watermark. Table 2 collects the results in terms of the average PSNR and NC about above mentioning five images, and Wiener filtering with different windows for the watermarked image.

To illustrate the effect of the Wiener filtering to an individual tested image, Fig. 5a shows the watermarked images of "*Flower*" under 7 × 7 window and Fig. 5b shows the corresponding extracted watermark. The experiment results demonstrated that the proposed algorithm has great robustness against Wiener filtering.

Table 2 PSNR and NC for Wiener filtering attack

Windows	PSNR	NC
3 × 3	40.07	0.9261
5 × 5	35.98	0.8492
7 × 7	30.62	0.8019

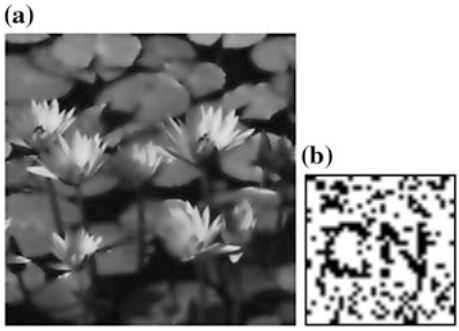


Fig. 5 **a** Watermarked image after Wiener filtering with a 7 × 7 window. **b** The watermark extracted from (**a**), NC = 0.8219

2. Mean filtering Mean filtering is another kind of common watermark attacks method. Table 3 collects the results in terms of the average PSNR, NC about above mentioning five images, and mean filtering with different windows for the watermarked image, respectively.

To illustrate the effect of the mean filtering to an individual tested image, Fig. 6a shows the watermarked images of "*Flower*" under 5 × 5 window and Fig. 6b shows the corresponding extracted watermark. Experiment results show that the proposed algorithm has very robustness against mean filtering.

3. Histogram equalization Histogram equalization is a kind of the common signal processing operation. Figure 7a shows the watermarked images of "*Flower*" after attacked by histogram equalization and Fig. 7b shows the corresponding extracted watermark.

Above experiment results confirm that proposed algorithm is robustness to histogram equalization attack, but also has more invisibility.

4. Gaussian noise The addition Gaussian noise is also a very common signal attack. We introduce Gaussian noise to watermarked image, Gaussian noise is zeros mean, and variance are 0.0005, 0.001, 0.002, 0.003 respectively. Figure 8a shows the watermarked images of "*Flower*" after attacked by Gaussian noise, and Fig. 8b–e shows the corresponding extracted watermark. The experiment results confirmed that the proposed algorithm has great robustness against Gaussian noise.
5. JPEG Compression We demonstrate the robustness of new algorithm. Since most digital images on networks are compressed, resistance against lossy

Table 3 PSNR and NC for mean filtering attack

Windows	PSNR	NC
3 × 3	38.25	0.8564
5 × 5	33.16	0.8091
7 × 7	29.67	0.7023

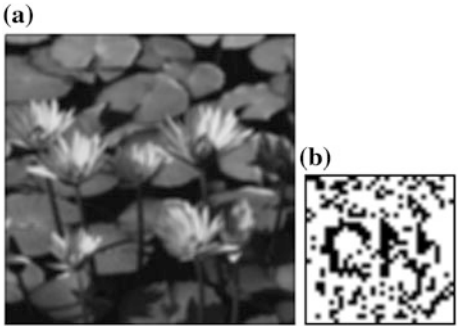


Fig. 6 **a** Watermarked image after mean filtering with a 5 × 5 window. **b** The watermark extracted from (**a**), NC = 0.7952

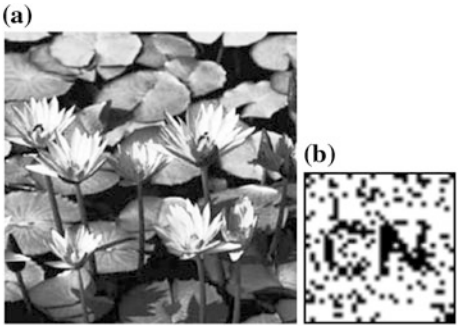


Fig. 7 **a** Watermarked image attacked by histogram equalization. **b** The watermark extracted from (**a**), NC = 0.8584

compression is essential. We obtain the watermark extraction results for the JPEG compression of watermarked '*Flower*' image with different quality factors (QF) 45, 40, 20, 15 to Fig. 4 respectively. The extraction results are shown in the Fig. 9a–d.

The experiment results demonstrated sufficiently that the proposed algorithm has great robustness against JPEG compression.

6. Cropping We cropped a part of the watermarked image, 1/4 cropped and filled with pixels valued 0 and 1/2 cropped and filled with pixels valued 255 respectively, and the extraction results are shown in Fig. 10. These experiments show that proposed algorithm is robustness to cropping attack.

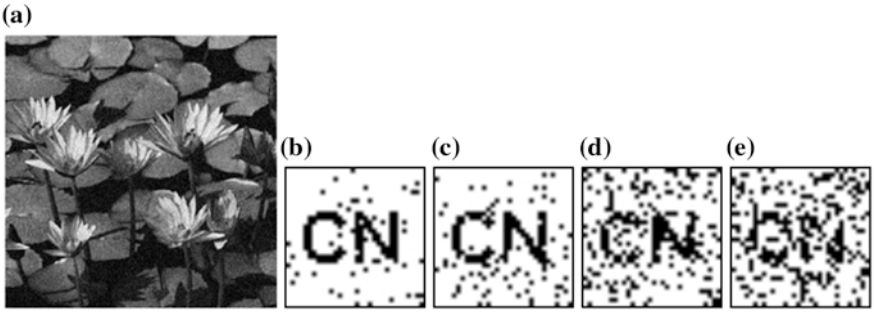


Fig. 8 **a** Watermarked image attacked by Gaussian noise with zero-mean and variance = 0.003, **b** extracted watermark when variance = 0.0005, **c** extracted watermark when variance = 0.001, **d** extracted watermark when variance = 0.002, and **e** extracted watermark when variance = 0.003

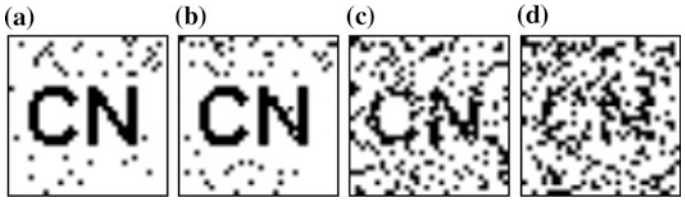


Fig. 9 Extracted images with a quality factors of 45, 40, 20, 15 respectively

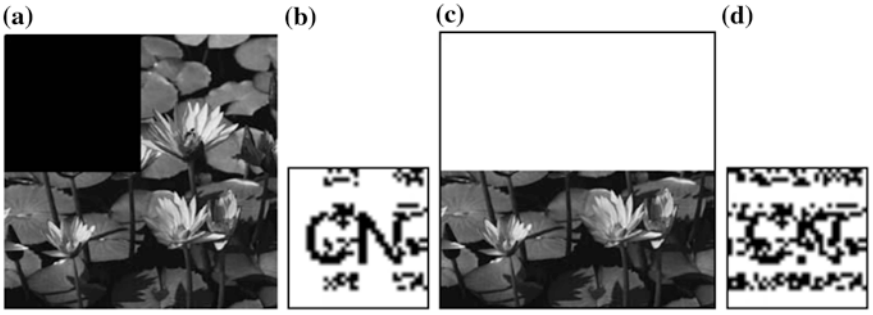


Fig. 10 Cropped the watermarked image 1/4 shown in (**a**) or 1/2 in (**c**) by different methods and the extracted watermarks are shown in (**b**) and (**d**) respectively

5.2.3 Comparison with Other Methods

We compare proposed algorithm to [11] method to *Lena*, *Goldhill* and *Peppers* images. The experiment results after various attacked are shown in Table 4. In Table 4, "Proposed" is average NC value to three images *Lena*, *Peppers*, and *Goldhill*.

Table 4 NC after attacked by JPEG compression with different QF of [11]'s methods

QF (%)	90	80	70	60	50	40	30	20	10
Lena[a]	0.99	0.99	0.97	0.95	0.96	0.90	0.82	0.67	0.34
Goldhill[a]	0.98	0.97	0.96	0.95	0.90	0.88	0.81	0.69	0.30
Peppers[a]	0.99	0.99	0.97	0.96	0.96	0.93	0.86	0.72	0.35
Proposed	1.000	1.000	0.991	0.989	0.985	0.941	0.847	0.768	0.626

[a] is proposed by [11]

Table 5 PSNR and NC after various attacks of [12]'s methods

Various attacks	[12][b]	[13][b]	[14][b]	[15][b]	Proposed
PSNR	38.92	40.89	40.08	26.77	40.86
LPF 3 × 3	0.86	0.92	0.91	1.00	0.9406
Median 3 × 3	1.00	0.87	0.84	1.00	1.0000
JPEG (80)	0.86	0.84	0.89	1.00	1.0000
JPEG (75)	0.82	0.81	0.85	1.00	1.0000
JPEG (50)	0.55	0.69	0.79	1.00	0.9259
Cropping 1/4	1.00	0.89	0.93	1.00	0.9772
Scaling 256 × 256	1.00	0.67	1.00	0.99	0.9898

[b] are proposed by [12–15], respectively

Numerical values in Table 4 show that new algorithm has shown the best performance for robustness against JPEG compression attacks. The performance of new algorithm is also compared with the results reported in [12–15] respectively. "Proposed" is average PSNR or NC value to five tested images *Baboon*, *Flower*, *Lena*, *Peppers*, and *Goldhill*. Numerical values in Table 5 show that the proposed algorithm has shown the very good performance for invisibility and robustness against various attacks. It is also seen that NC values for the method in [12] is little better compared to the proposed algorithm in case of lowpass filtering (LPF), JPEG compression (50 %), Cropping 1/4, and Scaling 256 × 256, but PSNR is much lower than the latter, did not achieve a good trade-off between invisibility and robustness.

6 Conclusion

Proposed new watermarking algorithm has the following advantages: (1) It can resist to common image processing attacks. Furthermore, watermark has good invisibility, and which makes that the proposed algorithm solves the conflict between invisibility and robustness better. (2) It is very difficult only to use a threshold determined by the experiments. By the proposed algorithm, this problem is a good solution. (3) The proposed algorithm shows the complex texture information of the image can be classified by ANN, and which makes that the proposed algorithm is very suitable to design digital image watermark algorithm.

Acknowledgments This work was supported by the science and technology research program of Wuhan of China (Grant No. 201210121023).

References

1. Jin, C.: Adaptive robust image watermark scheme based on fuzzy comprehensive evaluation and analytic hierarchy process. SIViP **6**, 317–324 (2012)
2. Li, L.D., Yuan, X.P., Lu, Z.L., Pan, J.S.: Rotation invariant watermark embedding based on scale-adapted characteristic regions. Inf. Sci. **180**, 2875–2888 (2010)
3. Zhang, F., Zhang, H.B.: Applications of a neural network to watermarking capacity of digital image. Neurocomputing **67**, 345–349 (2005)
4. Zhang, J., Wang, N.C., Xiong, F.: Hiding a logo watermark into the multi-wavelet domain using neural networks. In: 14th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'02), pp. 477–482 (2002)
5. Frank, H.L.L., Chen, Z.Y., Tang, L.: Novel perceptual modeling watermarking with MLF neural networks. Int. J. Inf. Technol. **10**, 82–85 (2004)
6. Tsai, H.H., Liu, C.C.: Wavelet-based image watermarking with visibility range estimation based on HVS and neural networks. Pattern Recogn. **44**, 751–763 (2011)
7. Haykin, S.: Neural Networks, A Comprehensive Foundation. Macmillan, New York (1994)
8. Ohlsson, N., Helander, M., Wohlin, C.: Quality improvement by identification of fault-prone modules using software design metrics. In: International Conference on Software Quality, pp. 1–13 (1996)
9. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press, New York (1995)
10. Haykin, S.: Neural Networks. Prentice Hall, New Jersey (1999)
11. Lin, W.H., Wang, Y.R., et al.: A blind watermarking method using maximum wavelet coefficient quantization. Expert Syst. Appl. **36**, 11509–11516 (2009)
12. Nasir, I., Weng, Y., Jiang, J.: A new robust watermarking scheme for color image in spatial domain. In: IEEE Conference on Signal Image Technologies and Internet-Based System, pp. 942–947. China (2007)
13. Zhang, D., Wu, B., Sun, J., Huang, H.: A new robust watermarking algorithm based on DWT. In: The Second International Congress on Image and Signal Processing, pp. 1–6. Tianjin, China (2009)
14. Luo, K., Tian, X.: A new robust watermarking scheme based on wavelet transform. In: Congress Image Signal Processing, pp. 312–316. Hainan, China (2008)
15. Verma, B., Jain, S.: A spatial domain robust non-oblivious watermarking scheme for image database. In: The Second Indian International Conference on Artificial Intelligence, pp. 3071–3085. Puna, India (2005)

PPSA: A Tool for Suboptimal Control of Time Delay Systems: Revision and Open Tasks

Libor Pekař and Pavel Navrátil

Abstract During the development of algebraic controller design in a special ring for time delay systems (TDSs) a problem of a suitable free controller parameters setting appeared. The first author of this contribution recently suggested a natural idea of placing the dominant characteristic numbers (poles) and zeros of the infinite-dimensional feedback control system on the basis of the desired overshoot for a simple finite-dimensional matching model and shifting of the rest of the spectrum. However, the original procedure called the Pole-Placement Shifting based controller tuning Algorithm (PPSA) was not developed and described entirely well. The aim of this paper is to revise the idea of the PPSA and suggest a possible ways how to improve or extend the algorithm. A concise illustrative example is attached to clarify the procedure for the reader as well.

Keywords Time delay systems · Pole placement controller tuning · Optimization · Direct-search algorithms · Evolutionary algorithms · SOMA · Nelder-Mead algorithm · Gradient sampling algorithm · Model matching

1 Introduction

Time delay systems (TDSs) constitute a huge class of processes and systems that are affected by any form of delay or latency, either in the input–output relation (as it is known in classical engineering problems) or inside the system dynamics (in this case notions of internal or state delays are introduced). The latter models and

L. Pekař (✉) · P. Navrátil
Faculty of Applied Informatics, Tomas Bata University in Zlín, Zlín, Czech Republic
e-mail: pekar@fai.utb.cz

P. Navrátil
e-mail: pnavratil@fai.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_2,

processes those are much more involved for analysis and control can be found in many theoretical and practical applications covering various fields of human activity, such as technology, informatics, biology, economy, etc., see e.g. [1–4].

A typical feature of TDSs is their infinite spectrum, due to transcendental nature of the characteristic equation, i.e. they have an infinite number of solution modes and corresponding system poles. This unpleasant attribute makes them difficult to analyze and design a control law as well. Linear time-invariant TDSs can be modeled and described by transfer functions by means of the Laplace transform. In most cases, roots of the transfer function denominator coincide with system poles.

The ring of quasipolynomial meromorphic functions (R_{MS}), originally developed and introduced in [5] and revised and extended in [6], represents a possible tool for description and control design of TDSs. However, in many cases, namely, for unstable TDSs, the control algorithm must deal with also infinitely many feedback characteristic poles the positions of which depend on the selectable controller parameters. The use of pole-placement (pole-assignment, root-locus) tuning algorithms can be a possible way how to solve the setting problem, see e.g. [7–9]. However, these algorithms deal with poles only ignoring closed-loop zeros and/or they have been derived for state-space controllers.

The idea of the Pole-Placement Shifting based controller tuning Algorithm (PPSA) provides slightly different approach [10]. It is based on the analysis of a simple finite-dimensional model where the relative maximum overshoot, relative dumping and relative time-to-overshoot of the reference-to-output step response are calculated and serve as a control performance indicators. Then, according to the selected values, the desired positions of dominant (i.e. the rightmost) poles and zeros are calculated, and poles and zeros of the infinite-dimensional feedback system are shifted to the prescribed positions while the rest of the spectrum is pushed to the left (i.e. to the "stable" region). In some sense, it represents a matching problem. The initial solution (i.e. controller parameter setting) is obtained using the Quasi-Continuous Shifting Algorithm (QCSA) [7, 8] which is followed by the use of an advanced numerical optimization algorithm. The method was independently developed in [9]; however, there are some essential differences—the reader is referred e.g. to [10] for details.

However, the original algorithm was described neither precisely nor in details and it contains some shortcomings and errors. Thus, the aim of this contribution is to revise and consolidate the PPSA and raise some open tasks how to improve and accelerate the algorithm. In this connection, the reader is kindly asked to participate on the solution of these problems in the future if he or she is interested in them.

To make the procedure clearer (to the reader) a short illustrative example on the control of an unstable time delay system by means of Matlab-Simulink environment is provided.

2 Time Delay Systems: Introductory Description

Since the reader is supposed to be a non-expert in system and control theory and the description and control design of TDSs is not the primary topic of this contribution, only a very concise overview of TDS models is provided such that all necessary information are given him or her.

A possible formulation of a TDS model (either a plant or a delayed control feedback loop) can be done using the transfer function in a complex variable s as the direct consequence of the use of the Laplace transform as follows

$$G(s) = \frac{b(s)}{a(s)} \tag{1}$$

where $a(s)$, $b(s)$ are quasipolynomials of a general form

$$x(s) = s^n + \sum_{i=0}^{n} \sum_{j=1}^{h_i} x_{ij} s^i \exp\left(-s\eta_{ij}\right); \quad \eta_{ij} \geq 0, x_{ij} \in \mathbb{R} \tag{2}$$

where η_{ij} express delays and $\mathbb{R}$ means the set of real numbers. If delays are included only in the numerator $b(s)$, they influence the input-output relation; in the contrary, the system contain internal delays and equation $a(s) = 0$ has infinitely many solution. These solution values constitute (in overwhelming majority of cases) system poles, more precisely, poles $s_i, i = 1, 2, \ldots$ are singularities of $G(s)$ satisfying

$$\lim_{s \to s_i} G(s) = \pm\infty; \quad \exists n_0, \forall n \geq n_0 : \lim_{s \to s_i} (s - s_0)^n G(s) < \infty \tag{3}$$

Zeros have the same meaning as in (3) yet for $1/G(s)$ instead of $G(s)$, i.e. they coincide with the roots of $b(s)$ (in most cases).

3 Problem Formulation

Now consider that $G(s)$ means the control feedback transfer function. Some control design approaches yield this function with the denominator containing delays along with free real controller parameters from the set $\mathbf{K} = \{k_1, k_2, \ldots, k_r\} \neq \varnothing \in \mathbb{R}^n$. This results in the infinite-dimensional (delayed) control feedback. Naturally, the numerator can own delays (and controller parameters) as well.

The idea of the PPSA is to match some number of the rightmost (i.e. the dominant) poles and zeros of $G(s)$ with all poles and zeros of a finite-dimensional model $G_m(s)$. Thus the selected poles and zeros of $G(s)$ are quasi-continuously

shifted to the desired positions by small steps and the rest of both spectra (of poles and zeros) try to push to the left (i.e. to the stable complex semiplane) as far as possible. The shifting can be done e.g. using the QCSA or via an advanced algorithm, [11–13], minimizing a suitable cost function reflecting the distance of dominant poles from prescribed positions and the spectral abscissa (i.e. the value of the real part of the rightmost pole/zero). By doing this, the values of $\mathbf{K}$ are being adjusted and hence the controller parameters are being tuned.

A crucial problem is to choose a suitable number of prescribed poles and zeros, i.e. degrees of the numerator, $N(s)$, and denominator, $D(s)$, of $G_m(s)$. Let us denote the numerator and the denominator as $N(s, \mathbf{K}_N)$ and $D(s, \mathbf{K}_D)$, respectively, where $\mathbf{K}_N$ and $\mathbf{K}_D$ mean free real parameters of the numerator and denominator, respectively, with $r_N = |\mathbf{K}_N| \geq 0$, $r_D = |\mathbf{K}_D| > 0$. It is initially assumed that equations $N(s_i, \mathbf{K}_N) = 0$, $D(s_j, \mathbf{K}_D) = 0$ are independent for arbitrary yet fixed s_i, s_j with $i = 1, 2, \ldots n_N \leq r_N$, $j = 1, 2, \ldots n_D \leq r_D$, that is

$$\begin{aligned} \operatorname{rank}\left[\frac{\partial}{\partial k_{N,l}} N(s_i, \mathbf{K}_N)\right]_{\substack{i=1,2,\ldots n_N \\ l=1,2,\ldots r_N}} &= n_N \\ \operatorname{rank}\left[\frac{\partial}{\partial k_{D,l}} D(s_j, \mathbf{K}_D)\right]_{\substack{j=1,2,\ldots n_D \\ l=1,2,\ldots r_D}} &= n_D \end{aligned} \tag{4}$$

Then the following conditions must hold: As indicated above, the number of prescribed poles, n_D, and zeros, n_N, must be less or equal to the number of corresponding free parameters to obtain a solvable matching problem. Moreover, if one needs to enable shifting the rest of the spectrum to the left, some parameters might not be bounded with desired position of roots, hence

$$0 \leq n_D < r_D, \quad 0 < n_N < r_N \tag{5}$$

where $\Delta n_D = r_D - n_D, \Delta n_N = r_N - n_N$ serve for adjusting the rightmost real parts of the rest of spectra. Naturally, the number of all desired solutions can not exceed the number of all free parameters, which gives rise to

$$n_D + n_N < r \tag{6}$$

In addition, the model has to be strictly proper, i.e.

$$n_D < n_N \tag{7}$$

Conditions (5)–(7) ought to be taken into account when designing the finite-dimensional model.

4 PPSA Strategies

Three possible revised modifications of the PPSA follows. A thorough algorithm description is consequently supported by its vague explanation and discussion in all three cases. Let is use these notations in the algorithms: $\mathbf{K} = \mathbf{K}_N \cup \mathbf{K}_D$ where numerator coefficients of $G(s)$ read $\mathbf{K}_N = \mathbf{K}_{N\setminus D} \cup \mathbf{K}_{\overline{ND}}$ with $r_{\overline{ND}} = \left|\mathbf{K}_{\overline{ND}}\right| = |\mathbf{K}_N \cap \mathbf{K}_D|$, $r_{N\setminus D} = \left|\mathbf{K}_{N\setminus D}\right| = \left|\mathbf{K}_N \setminus \mathbf{K}_{\overline{ND}}\right|$, whereas denominator ones analogously are $\mathbf{K}_D = \mathbf{K}_{D\setminus N} \cup \mathbf{K}_{\overline{ND}}$ with $r_{D\setminus N} = \left|\mathbf{K}_{D\setminus N}\right| = \left|\mathbf{K}_D \setminus \mathbf{K}_{\overline{ND}}\right|$. Simply, $r_N = r_{N\setminus D} + r_{\overline{ND}}$, $r_D = r_{D\setminus N} + r_{\overline{ND}}$.

Algorithm 1 (*PPSA strategy 1: "Poles First Independently"*)
Input. Closed-loop reference-to-output transfer function $G(s)$ with $r_{N\setminus D} > 0$.

Step 1. Set $n_D = r_D - 1$, thus $\Delta n_D = r_D - n_D = 1$. (Or just select $n_D < r_D$ as high as desirable).

Step 2. Verify that there can exist a non-negative number n_N satisfying

$$0 \le n_N < \min\left\{n_D, r_{N\setminus D}\right\} \tag{8}$$

If (8) holds, fix n_N and go to Step 3; otherwise, set $n_D = n_D + 1$. If $n_D < \min\{r_D, r_{N\setminus D} + 1\}$, i.e. $n_D < r_D$ and $n_D \le r_{N\setminus D}$, go to Step 2, else terminate the procedure (a solution does not exist).

Step 3. Choose a simple matching model of a stable finite-dimensional system with the numerator of degree n_N, the denominator of degree n_D and the unit static gain governed by the transfer function $G_m(s)$. The model can be prescribed e.g. according to the desired dynamic behavior of the feedback loop. Its poles and zeros are referred as "prescribed" below.

Step 4. Set a part of the spectrum of poles via the number n_D of coefficients from the set $\mathbf{K}_D$ into the prescribed positions while the rest of denominator parameters are chosen arbitrarily. If these poles are dominant, initialize the counter of currently shifted poles as $n_{sp} = n_{sp,m} + n_{sp,opt} = n_D + 1$ where $n_{sp,m} = n_D$ and $n_{sp,opt} = 1$. If not, then $n_{sp} = n_{sp,m} + n_{sp,opt} = n_D$, $n_{sp,m} = n_D$, $n_{sp,opt} = 0$.

Step 5. Check that (4) holds for the number n_{sp} of the rightmost poles and $\mathbf{K}_D$. If not, go to Step 4 and reset the initial assignment; otherwise, shift the number $n_{sp,m}$ of the rightmost feedback system poles towards the prescribed locations (i.e., keep in the close proximity of them), e.g. using the QCSA, whereas the number $n_{sp,opt}$ of poles is pushed to the left. If necessary, increase $n_{sp,opt} \Rightarrow n_{sp}$. If $n_{sp} = r_D$ and/or the shifting is no more successful, go to Step 6.

Step 6. If all $n_{sp,m}$ poles are dominant, go to Step 7. Otherwise, select a suitable cost function $\Phi_P(\mathbf{K}_D)$ reflecting the distance of dominant poles of $G(s)$ from prescribed positions and the spectral abscissa. Minimize $\Phi_P(\mathbf{K}_D)$ starting with results from Step 5 (using e.g. an advanced iterative algorithm, [11–13]). Fix $\mathbf{K}_D$.

Step 7. Place a part of the spectrum of zeros of $G(s)$ using the number of n_N coefficients from the set $\mathbf{K}_{N\setminus D}$ into the prescribed positions and the remaining

parameters in $\mathbf{K}_{N\setminus D}$ are chosen arbitrarily. If these zeros are dominant, initialize the counter of currently shifted zeros as $n_{sz} = n_{sz,m} + n_{sz,opt} = n_N + 1$ where $n_{sz,m} = n_N$ and $n_{sz,opt} = 1$; otherwise, set $n_{sz} = n_{sz,m} + n_{sz,opt} = n_N$, $n_{sz,m} = n_N$, $n_{sz,opt} = 0$.

Step 8. Check that (4) holds for the number n_{sz} of the rightmost zeros of $G(s)$ and for current values of $\mathbf{K}_{N\setminus D}$. If it is approved, $n_{sz,m}$ zeros are to be incessantly moved to the prescribed positions whereas $n_{sz,opt}$ zeros are pushed to the left. If necessary, increase $n_{sz,opt} \Rightarrow n_{sz}$. If $n_{sz} = r_{N\setminus D}$ and/or the shifting is no more successful, go to Step 9.

Step 9. If all $n_{sz,m}$ zeros are dominant, the algorithm is finished. Otherwise, select a suitable cost function $\Phi_Z(\mathbf{K}_{N\setminus D})$ reflecting the distance of dominant zeros of $G(s)$ from prescribed positions and the spectral abscissa. Minimize $\Phi_Z(\mathbf{K}_{N\setminus D})$ with initial setting of $\mathbf{K}_{N\setminus D}$ obtained from Step 8.

Output. The vector of controller parameters $\mathbf{K} = \mathbf{K}_{N\setminus D} \cup \mathbf{K}_D$, positions of the rightmost poles and zeros and the spectral abscissae.

The above presented strategy of the PPSA places the feedback poles to the desired positions first, and consequently, transfer function numerator parameters not included in the numerator serve as tuning tool for inserting zeros to the desired loci. Thus, zeros are placed independently from poles by means of $\mathbf{K}_{N\setminus D}$. In both the cases, the rest of the spectrum is pushed to the left as far as possible to minimize the spectral abscissa. If this quasi-continuous shifting is not successful, a trade-off between the zeros/poles matching task and the spectral abscissa is optimized. Note that condition (8) stem from (5) and (7) while (6) always holds for this strategy.

In fact, the QCSA or a shifting technique presented in [14] enables to shift a conjugate pair of roots along the real axis using a single controller parameter, i.e. it is possible to write $n_{sp,m} + n_{sp,opt,R} + n_{sp,opt,C} \leq r_D$ and $n_{sz,m} + n_{sz,opt,R} + n_{sz,opt,C} \leq r_{N\setminus D}$ where a subscript R denotes real roots whereas C means complex conjugate pairs.

If $r_{D\setminus N} > 0$, it is possible to apply the strategy reversely, i.e. to set zeros first and, afterwards, to place poles. However, the presented variant prefers poles since they affect the system dynamics more significantly.

Let us present now another (a simpler) strategy combining both, the poles and zeros matching, under one procedure.

Algorithm 2 (*PPSA strategy 2: "Poles and Zeros Together"*)
Input. Closed-loop reference-to-output transfer function $G(s)$.

Step 1. Set $n_D = r_D - 1$, or just select $n_D < r_D$ as high as desirable.
Step 2. Verify that there exists a non-negative number n_N satisfying

$$0 \leq n_N < \min(n_D, r - n_D, r_N) \tag{9}$$

If (9) holds, fix n_N and go to Step 3; otherwise, set $n_D = n_D - 1$. If $r_D > n_D \geq \max\{r - n_D, r_N\}$, go to Step 2; contrariwise, a solution does not exist.

Step 3. Choose a simple model $G_m(s)$ of a stable finite-dimensional system with the numerator of degree n_N, the denominator of degree n_D, the unit static gain and prescribed (desired) zeros and poles.

Step 4. Set finite subsets of both the spectra, poles and zeros, via the number n_D of coefficients from the set $\mathbf{K}_D$ and by means the number n_N of coefficients from the set $\mathbf{K}_N$, respectively, into the prescribed positions of $G_m(s)$ while the rest of parameters from $\mathbf{K}$ are chosen arbitrarily. If all these poles are dominant, initialize the counter of currently shifted poles as $n_{sp} = n_{sp,m} + n_{sp,opt} = n_D + 1$ where $n_{sp,m} = n_D$ and $n_{sp,opt} = 1$; otherwise, $n_{sp} = n_{sp,m} + n_{sp,opt} = n_D$, $n_{sp,m} = n_D$, $n_{sp,opt} = 0$. Similarly for zeros, if they are the rightmost ones, set $n_{sz} = n_{sz,m} + n_{sz,opt} = n_N + 1$, $n_{sz,m} = n_N$, $n_{sz,opt} = 1$; in the contrary, $n_{sz} = n_{sz,m} + n_{sz,opt} = n_N$, $n_{sz,m} = n_N$, $n_{sz,opt} = 0$.

Step 5. Check that (5) holds for the number n_{sp} of the rightmost poles and $\mathbf{K}_D$, and for n_{sz} dominant zeros along with $\mathbf{K}_N$. If not, go to Step 4 and reset the initial assignment; otherwise, shift mutually the number $n_{sp,m}$ and $n_{sz,m}$ rightmost feedback system poles and zeros, respectively, towards the prescribed locations the number $n_{sp,opt}$ and $n_{sz,opt}$ of poles and zeros, respectively, is pushed to the left along the real axis. If necessary, increase $n_{sp,opt} \Rightarrow n_{sp}$ and/or $n_{sz,opt} \Rightarrow n_{sz}$. If $n_{sp,m} + n_{sp,opt,R} + n_{sp,opt,C} \leq r_D$ and $n_{sz,m} + n_{sz,opt,R} + n_{sz,opt,C} \leq r_N$ and $n_{sz,m} + n_{sp,m} + n_{sp,opt,C} + n_{sz,opt,R} + n_{sz,opt,C} \leq r$, or the shifting is no more successful, go to Step 6.

Step 6. If all $n_{sp,m}$ poles and $n_{sp,z}$ zeros are dominant, the procedure is finished. Otherwise, select a suitable cost function $\Phi(\mathbf{K})$ reflecting the distance of dominant poles and zeros of $G(s)$ from prescribed positions and spectral abscissae of both the spectra. Minimize $\Phi(\mathbf{K})$ starting with results from Step 5.

Output. The vector of controller parameters $\mathbf{K}$, positions of the rightmost poles and zeros and the spectral abscissae.

The methodology is useful in case $r_{N\setminus D} = 0$ (and/or $r_{D\setminus N} = 0$). Roughly speaking to summarize it, poles and zeros are moved simultaneously over a common set $\mathbf{K}$ of adjustable parameters, therefore their positions are not independent to each other.

A trade-off between Algorithm 1 and Algorithm 2 can be done by a procedure when only a subset $\mathbf{K}_{\overline{ND},D} \subset \mathbf{K}_{\overline{ND}}$ is dedicated to poles while a subset $\mathbf{K}_{\overline{ND},N} \subset \mathbf{K}_{\overline{ND}}$ is given to zeros to be modified, where $\mathbf{K}_{\overline{ND},D} \cap \mathbf{K}_{\overline{ND},N} = \emptyset$. Hence, these disjunctive sets provide a certain kind of independency.

The last conceivable strategy consists in the accurate setting of a part of the spectrum of zeros, which results in that some parameters from $\mathbf{K}_N$ are dependent to others, and consequently, find the optimal setting of independent parameters by strategies from Algorithm 1 or Algorithm 2. This idea, however, does not guarantee the dominancy of the placed zeros.

Due to the limited space, these two strategies mentioned above will be a topic of any of our future papers.

5 Illustrative Example

A very concise demonstrative example follows to provide the reader with the idea of control of TDS and the PPSA.

In [15] a mathematical model of a skater on the swaying bow, which represents an unstable TDS system, was introduced, and a corresponding controller designed in the R_{MS} ring was derived in [16]. The eventual reference-to-output transfer function reads

$$\begin{aligned} G(s) &= \frac{bb_Q(s)}{(s+m_0)^4 m_Q(s)} \exp(-(\tau+\vartheta)s) \\ b_Q(s) &= b\left(q_3 s^3 + q_2 s^2 + q_1 s + q_0\right)(s+m_0)^4 + p_0 m_0^4 s^2\left(s^2 - a\,\exp(-\vartheta s)\right) \\ m_Q(s) &= s^2\left(s^2 - a\,\exp(-\vartheta s)\right)\left(s^3 + p_2 s^2 + p_1 s + p_0\right) \\ &\quad + b\,\exp(-(\tau+\vartheta)s)\left(q_3 s^3 + q_2 s^2 + q_1 s + q_0\right) \end{aligned} \tag{10}$$

where delays τ, $\vartheta \geq 0$ stand for the skater's and servo latencies, respectively, b, a are real plant parameters. Note that the spectral assignment for the polynomial factor $(s+m_0)^4, m_0 > 0$ is trivial, then the goal is to find unknown parameters of $m_Q(s)$. To cancel the impact of the quadruple real pole $s_1 = -m_0$ to the feedback dynamics, it must hold that $m_0 \gg -\alpha(\mathbf{K})$ where $\alpha(\mathbf{K})$ expresses the spectral abscissa of the quasipolynomial factor. Hence, we have $\mathbf{K} = \{p_2, p_1, p_0, q_3, q_2, q_1, q_0\}$ with $\mathbf{K}_{\overline{ND}} = \{q_3, q_2, q_1, q_0, p_0\}$, $\mathbf{K}_{N\setminus D} = \emptyset$, $\mathbf{K}_{D\setminus N} = \{p_2, p_1\}$, $\mathbf{K}_N = \mathbf{K}_{\overline{ND}}$, $\mathbf{K}_D = \mathbf{K}$, that is $r = r_D = 7, r_{\overline{ND}} = r_N = 5$, $r_{N\setminus D} = 0, r_{D\setminus N} = 2$. Let us follow Algorithm 2 which is suitable in this case since $r_{N\setminus D} = 0$ and hence Algorithm 1 can not be used.

We attempt to set $n_D = 2$, then the conditions (9) reads $0 \leq n_N < 2$; therefore, let $n_D = 1$ and consider the model

$$G_m(s) = \frac{b_1 s + b_0}{s^2 + a_1 s + a_0} = k\frac{s - z_1}{(s - s_1)(s - \bar{s}_1)} \tag{11}$$

According to the desired dynamic properties, we prescribe a zero $z_1 = -0.18$ and a complex conjugate pair of poles $s_1 = -0.1 + 0.2\mathrm{j}$. Since the initially place roots are not dominant with abscissas for poles and zeros as $\alpha_P(\mathbf{K}) = 0.8959$ and $\alpha_Z(\mathbf{K}) = -0.1373$, respectively, set $n_{sp} = 2, n_{sz} = 1$ and perform Steps 5-6 of the PPSA by means of the QCSA.

In Figs. 1 and 2 distances of the rightmost poles pair σ and the zero ζ from the prescribed ones are displayed, and the evolution of $\mathbf{K}$ during the quasi-continuous shifting is provided in Fig. 3.

Further, the SOMA is used to minimize the cost function $\Phi(\mathbf{K}) = |\sigma_1 - s_1| + |\zeta_1 - z_1| + 0.01\alpha_{r,P}(\mathbf{K}) + 0.01\alpha_{r,Z}(\mathbf{K})$ where $\alpha_{r,P}(\mathbf{K})$, $\alpha_{r,Z}(\mathbf{K})$ mean the spectral abscissa of the rest of poles and zeros, respectively. It is worth noting that the

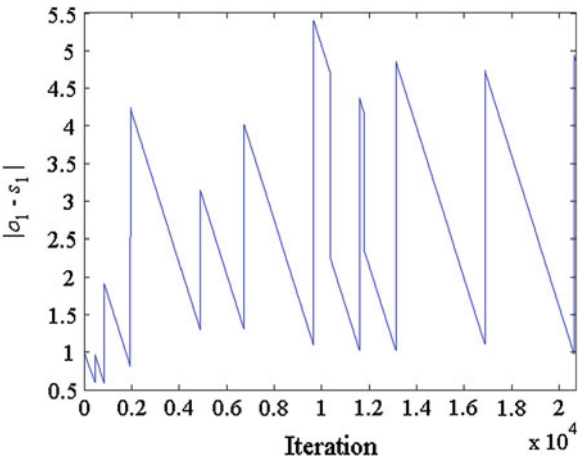


Fig. 1 Evolution of $|\sigma_1 - s_1|$ using the PPSA with QCSA

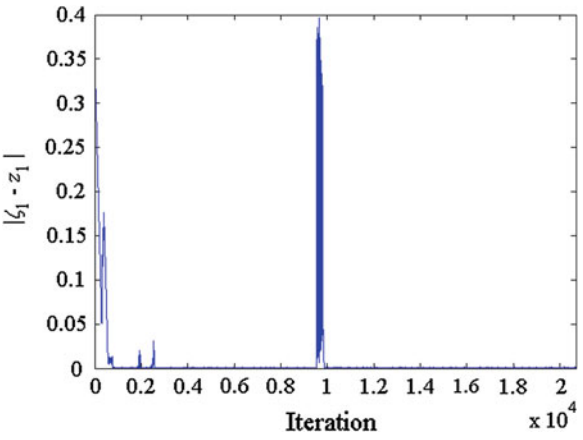


Fig. 2 Evolution of $|\zeta_1 - z_1|$ using the PPSA with QCSA

optimization yields only a slightly improvement giving the eventual spectra and the parameters set as in (12). However, final poles and zeros positions are quite far from the desired ones, which proves the fact about TDS that the desired spectrum can not be chosen arbitrarily in general.

$$\begin{aligned}\Omega_{P,opt} &= \{-0.1158 \pm 0.0674\text{j}, -0.1161 \pm 5.1163\text{j}, -0.1211 \pm 1.2103\text{j},\ldots\}\\ \Omega_{Z,opt} &= \{-0.1801, -0.2247 \pm 0.1032\text{j}, -0.7607, -2.817 \pm 8.1939\text{j},\ldots\}\\ \mathbf{K}_{opt} &= [5235.169,\ 9829.219,\ 1060.87,\ 78.2405,\ 30.9684,\ 1.763,\ 947.517]^T\end{aligned} \tag{12}$$

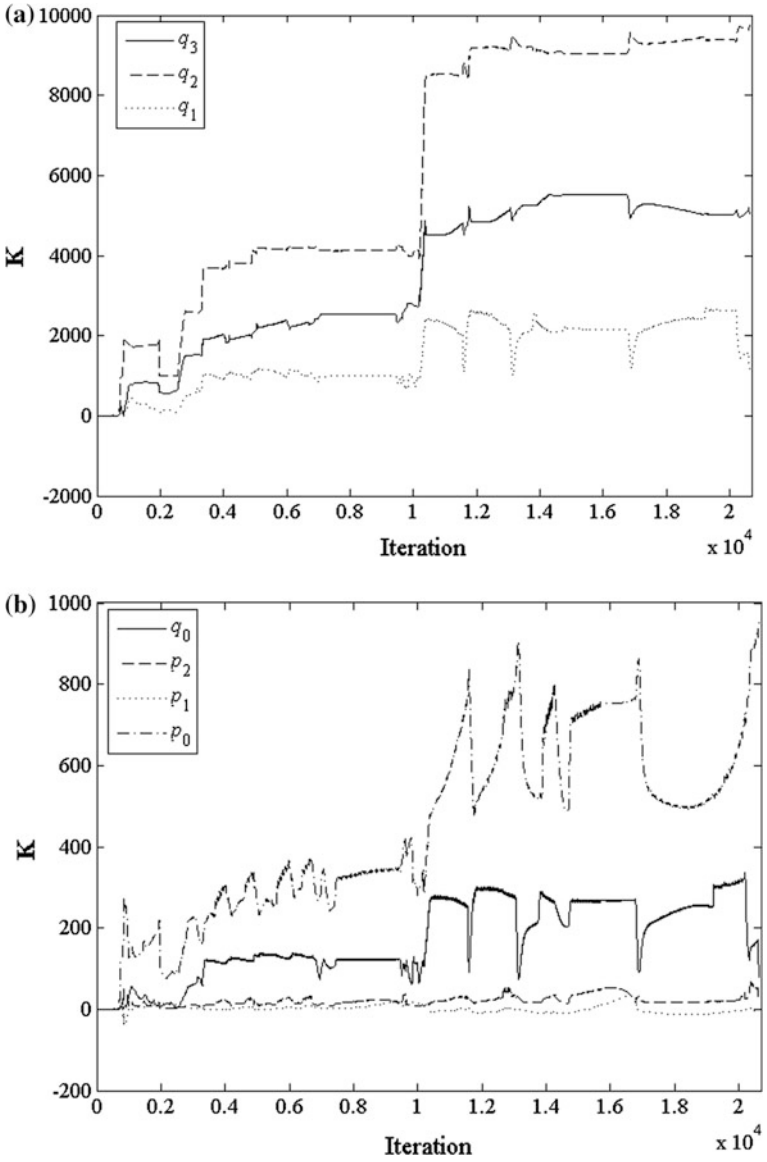


Fig. 3 Evolution of **K** using the PPSA with QCSA

6 Discussion

Let us now present some ideas how to modify, extend or improve the PPSA, regarding computation acceleration, shifting strategies, model selection etc.

Considering these aspects in the chronological order according to the running of Algorithm 1 or Algorithm 2, we can start with the selection of a finite-dimensional matching model. In the example above, it is supposed that the feedback dynamics is primarily given by positions of the rightmost poles and zeros where the model is found from the desired maximum overshoot, time-to-overshoot and the relative dumping. Naturally, other strategies how to prescribe the model (with corresponding roots) can be adopted. Moreover, the dominancy of the roots can be evaluated in a different way, e.g. in [14], the method based on the "weights" of modes of the impulse response was presented.

The initial shifting, convergence and the speed of the PPSA may be improved by the use of other "approaching" strategies, e.g. only roots of the same type (real, complex) are approaching to each other, or by thorough consideration that a complex conjugate pair means two separate roots instead of one (as it used here).

Last but not least another optimization procedures can be utilized in, e.g. the well-known and efficient NM algorithm [13] or some of many modern evolutionary or genetic algorithms. In fact, computationally the most time-consuming operation is the finding of the spectrum; hence the aim is to minimize the number of these spectral evaluations. For instance, it would be desirable to parallelize an existing spectrum-searching procedure and to utilize distributed computations on graphical cards, e.g. Compute Unified Device Architecture (CUDA) or Open Computing Language (OpenCL).

7 Conclusion

It is always difficult to tackle optimal or suboptimal control design or controller tuning for TDS. The presented paper has summarized and revised the basic principles of the PPSA which is based on quasi-continuous feedback poles and zeros shifting to the described dominant ones according to a selected finite-dimensional feedback model. The semi-finite result from the shifting has been then improved by an optimization procedure. Two possible PPSA strategies have been introduced and discussed, and the explanation has been supported by an illustrative example. In the future research, the other possible strategies will be analyzed and, moreover, the two presented ideas will be tested, compared and enhanced by tools discussed in this paper. Hence, the reader is kindly asked to participate on the future research, with the accent to provide us with the computational and programming support, to benchmark and verify the discussed ideas.

Acknowledgements The authors kindly appreciate the financial support which was provided by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089.

References

1. Hale, J.K., Verduyn Lunel S.M.: Introduction to functional differential equations. Appl. Math. Sci. **99**, (1993)
2. Kolmanovskii, V.B., Myshkis, A.: Introduction to the theory and applications of functional differential equations. Cluwer Academy, Dordrecht (1999)
3. Niculescu S.I.: Delay effects on stability. Lecture Notes in Control and Information Sciences, vol. 269. Springer, Berlin (2001)
4. Richard, J.P.: Time-delay systems: An overview of some recent advances and open problems. Automatica **39**, 1667–1694 (2003)
5. Zítek, P., Kučera, V.: Algebraic design of anisochronic controllers for time delay systems. Int J Control **76**, 1654–1665 (2003)
6. Pekař, L.: A ring for description and control of time-delay systems. In: WSEAS Transaction on Systems 11. Special Issue on Modelling, Identification, Stability, Control and Applications, pp. 571–585 (2012)
7. Michiels, W., Engelborghs, K., Vansevevant, P., Roose, D.: Continuous pole placement for delay equations. Automatica **38**, 747–761 (2002)
8. Michiels, W., Vyhlídal, T.: An eigenvalue based approach for the stabilization of linear time-delay systems of neutral type. Automatica **41**, 991–998 (2005)
9. Michiels, W., Vyhlídal, T., Zítek, P.: Control design for time-delay systems based on quasi-direct pole placement. J Process Control **20**, 337–343 (2010)
10. Pekař, L.: On a controller parameterization for infinite-dimensional feedback systems based on the desired overshoot. WSEAS Trans. Syst. **12**, 325–335 (2013)
11. Zelinka, I.: SOMA-self organizing migrating algorithm. In: Onwobolu, G.C., Babu, B.V. (eds.) New optimization techniques in engineering, pp. 167–217. Springer, Berlin (2004)
12. Vanbiervliet, T., Verheyden, K., Michiels, W., Vandewalle, S.: A nonsmooth optimization approach for the stabilization of time-delay systems. ESIAM: Control Optim. Calc. Var. **14**, 478–493 (2008)
13. Nelder, J.A., Mead, R.: A simplex method for function minimization. Comput. J. **7**, 308–313 (1965)
14. Vyhlídal, T.: analysis and synthesis of time delay system spectrum. Ph.D. thesis. Faculty of Mechanical Engineering, Czech Technical University in Prague, Prague (2003)
15. Zítek, P., Kučera, V., Vyhlídal, T.: Meromorphic observer-based pole assignment in time delay systems. Kybernetika **44**, 633–648 (2008)
16. Pekař, L., Prokop, R.: Algebraic optimal control in RMS ring: a case study. Int. J. Math. Comput. Simul. **7**, 59–68 (2013)

Logistic Warehouse Process Optimization Through Genetic Programming Algorithm

Jan Karasek, Radim Burget and Lukas Povoda

Abstract This paper introduces process planning, scheduling and optimization in warehouse environment. The leading companies of the logistics warehouse industry still do not use planning and scheduling by automatic computer methods. Processes are planned and scheduled by an operational manager with detailed knowledge of the problem, processed tasks and commodities, warehouse layout, performance of employees, parameters of equipment etc. This is a quantum of information to be handled by a human and it can be very time-consuming to plan every process and schedule the timetable. The manager is usually also influenced by stress conditions, especially by the time of holidays when everyone is making supplies and the performance of the whole warehouse management goes down. The main contribution of this work is (a) to introduce the novel automatic method for optimization based on the evolutionary method called genetic programming, (b) to give a description of a tested warehouse, and (c) to show the metrics for performance measurement and to give a results which states the baseline for further research.

Keywords Genetic programming · Logistics · Optimization · Scheduling

J. Karasek (✉) · R. Burget · L. Povoda
Faculty of Electrical Engineering and Communication, Department of Telecommunications, Brno University of Technology, Technicka 12, 616 00 Brno, Czech Republic
e-mail: karasekj@feec.vutbr.cz
URL: http://vutbr.cz/en/; http://splab.cz/en/

R. Burget
e-mail: burgetrm@feec.vutbr.cz

L. Povoda
e-mail: xpovod00@stud.feec.vutbr.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_3,

1 Introduction

The processes which come under operational level management competences such as planning and scheduling of daily routine tasks are important parts of everyday decision making. When these problems are handled in a big company with hundreds of employees, they become more and more complex. Furthermore, the complexity of the problem may arise from various sources, such as attributes related to performance of employees and attributes which describe the equipment, logistic warehouse and commodities layout, processed tasks, sub-tasks, and others. In a nutshell, the problem often becomes so complex that it is very difficult or nearly impossible to solve it only by skilled operational manager or any kind of mathematical programming method. By the time of writing this paper, the leading companies of the logistics and warehousing industry still have not used automated methods for process planning and scheduling.

The main aim of this paper lies (a) in introduction of the novel automatic method for process planning and scheduling based on genetic programming algorithms, (b) in description of a tested logistic warehouse, and (c) in introduction of the metrics for performance measurement and the initial results as reference points for further research with a more detailed view on one single example where the performance of the automated system has been proved to surpass the human operator. The main contribution of this paper is to help community dealing with logistics and warehouse optimization to set the baseline results for further development and joint research in the considered problem domain and to provide the metrics for performance measurement.

The rest of the paper is organized as follows. Section 2 deals with the work related to the problem considered in this paper. Similar problems and methods dealing with process planning and scheduling are discussed there. Section 3 describes in a nutshell a novel automatic method based on a genetic programming algorithm. Section 4 describes the standard layout of the logistic warehouse center which is also used as the reference point for further research. Section 5 describes a simple metrics for progress measurement, results of an operational manager, and results reached by a proposed system. Furthermore, this section also briefly describes an example where the proposed system has reached better results than the operational manager. The paper is concluded with Sect. 6.

2 Related Work

The problem of process planning and scheduling in logistic warehouses described in this paper deals with complex optimization of logistic warehouses and distribution centers. The problems addressed within the logistic warehouse optimization

deal mostly with the optimization of some part of the logistic warehouse, such as design of warehouse layout, design of receiving and shipping areas and design of other parts of the logistic warehouse. The products handled in the warehouse are also quite often subject to optimization—the optimization deals with product grouping, classing, and zoning. For more information see [1, 2].

There are two basic approaches to solving the warehouse optimization problem regarding process planning and scheduling. The first approach, commonly used when the problem is not so complex, are methods of mathematical programming [3]. The second approach uses heuristic methods. In the past, a lot of heuristic methods were used to solve the considered problem such as shifting bottleneck, dispatching rules, simulated annealing [4], particle swarm optimization [5] and/or tabu search [6]. The biggest group of algorithms used is Evolutionary Algorithms (EA) [7]. Genetic algorithms (GA), one of the biggest part of EAs, have demonstrated their potential for solving difficult optimization problems, and they have proved to be very efficient and adaptive solutions for complex problem solving.

3 Optimization Method

The GA showed potential for problem solving in difficult and complex situations which require a certain demand of adaptability and robustness. The problem of using GA is that in the case of this work the structure of chromosome is not given by any prescription. Therefore, the Genetic Programming (GP) as an optimization method has been chosen instead of GAs. The GP has demonstrated the same or even better potential than GAs and in addition to that the GP algorithm is able to design the structure of chromosome automatically. In recent years, GP algorithms have been successfully applied in many problem domains, where the algorithms were creating relatively complex problem solutions or whole automated systems. For instance, in the paper [8] the GP was successfully used for creation of the image detector that is able to detect relatively complex objects in noisy ultrasound data, in another work [9] the GP was used to select optimal features to classify emotions in textual data, and in [10] the GP driven by the context-free grammar was used to design a non-cryptographic hash.

Figure 1 shows the proposed GP algorithm which represents the computational core of the proposed automated system. The input of the algorithm consists of two basic parts. The first part of input is a buffer of all tasks waiting to be processed. The task in this context is perceived as one complete assignment given to the employee by the manager, e.g. a process of commodity storing can be considered a task. This task consists of several independent activities, so-called jobs. The concrete jobs for the mentioned task are (a) folding (the commodity from truck), (b) transfer (the commodity to the target coordinates in the warehouse), and (c) storing (the commodity to the rack). The second input is a set of employees who are able to process the tasks and their assigned equipment. Logical structure of the system is depicted in Fig. 2.

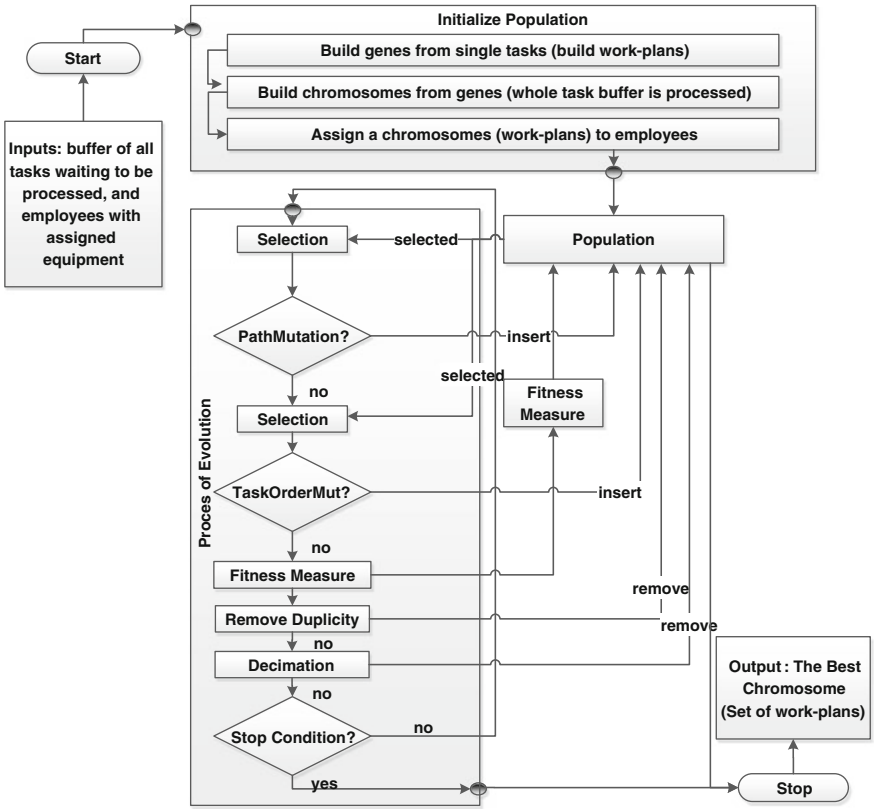


Fig. 1 Block scheme of genetic programming algorithm

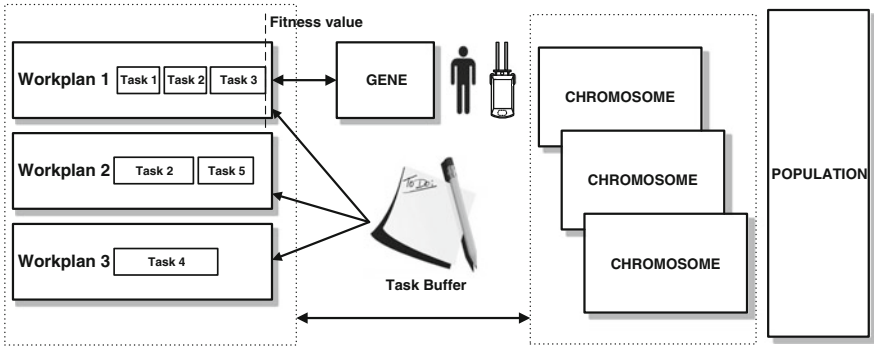


Fig. 2 Internal structure of algorithm core

Figure 2 represents inputs (task buffer, employees, and vehicles). The smallest logical structure of the proposed algorithm is a gene. A gene, in fact, is represented by a work-plan of employee. In the system there are as many work-plans as employees. The work-plans are filled automatically and randomly with tasks from the task buffer. All tasks waiting in the buffer to be processed have to be assigned to employee's work-plan. The work-plans form a chromosome and this is actually how the chromosomes are created. It is a completely random initialization process. Figure 2 also shows how the fitness function is calculated. The fitness value is determined as a finished time of the last task in a chromosome.

The evolution process is controlled by several parameters such as *population size*, *number of generation*, probabilities of evolutionary operators—a number of individuals who are copied to a new population *elitism*, probability of mutation operators application—*path mutation rate*, and *task order mutation rate*, a parameter which tells the evolution process to remove duplicities and the operator of decimation which holds the number of individuals under the prescribed level.

The whole evolution process is divided into several parts. Before the evolution process starts, the initial population has to be created as described in one of the previous paragraphs. When the initial population is created, the evolution process can run. The first step of evolution is to maintain the level of the best individuals in the population. This prevents the process from a decreasing tendency in the meaning of the best candidate solution. This process is called elitism, and a certain number of individuals R_e of the previous population is simply copied to a new population. In the case of this work $R_e = 1$. The second step is to apply genetic operators. First is the path mutation operator which is applied with the rate $R_{pm} = 30\,\%$. It means that 30 % of population will be mutated by path mutation. The second operator is the task order mutation which is applied with the rate $R_{to} = 30\,\%$. At this point the fitness value is calculated for all new individuals. When the genetic operators have been applied the algorithm can continue with duplicity removing and decimation of population, which secures the permanent number of individuals in the population. After this control processes the stop criterion is checked. If the stop condition is true, the evolution process is at the end and the best individual is stated as a solution. If the stop condition is false, the evolution process continues with next evolution step. The pseudo algorithm of the evolution process is described below.

```
method Output Evolution (PopulationSize, EvolutionSteps)
   var
     Double R_pm = 0.3;            %rate of path mutation
     Double R_to = 0.3;            %rate of task order mutations
   begin
     Population <- InitializePopulation(PS);
     EvaluatePopulation(Population);
     for (i = from 1 to ES)
       if (isPathMutationApplied) then
         n individuals <- Select(Population,R_pm);
         for (j = from 1 to n)
           Population <- PathMutation(get j from n);
         endfor
       endif
       if (isTaskOrderMutationApplied) then
         n individuals <- Select(Population,R_to);
         for (j = from 1 to n)
           Population <- TaskOrderMutation(get j from n);
         endfor
       endif
     endfor
     Return GetBestIndividual(Population);
   end
end.
```

3.1 Path Mutation

Path Mutation (Fig. 3a) is the first genetic operator designed for the purpose of this work. This kind of mutation is the simplest operator and its purpose is to change the path used to process a specific task (e.g. transportation of a pallet). The advantage of this operator lies in changing the path, especially when the collision of two vehicles is very probable or the lane between racks is under congestion.

The example in Fig. 3a shows how the operator works. R_{pm} percent of chromosomes is selected. First, the work-plan is selected at random in a chromosome, in this case it is work-plan no. 3. Second, the task is selected at random in the work-plan, in this case it is task no. 4, because it is only one in the work-plan. Then, the path mutation is applied and the transportation path is changed.

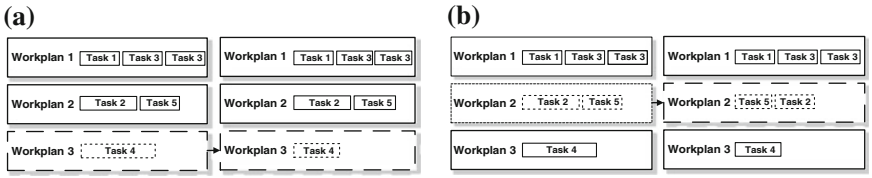


Fig. 3 Examples of path mutation operator and task order mutation operator

3.2 Task Order Mutation

The Task Order Mutation (Fig. 3b) is the second genetic operator designed. This operator is also quite simple, and its aim is to shuffle the tasks in the work-plan. This operator can show its advantage especially when the first task in the work-plan looks to be quite distant and it is more logic to process a closer task and then go further and further and process more distant tasks. The example in Fig. 3b shows how the operator works. R_{to} percent of chromosomes is selected. First, the work-plan is selected at random (work-plan no. 2). Then two tasks are randomly swapped, in this case there are only two tasks, so they are swapped.

4 Warehouse Layout

The reference warehouse described is based on a real-world situation. The warehouse is represented in Fig. 4 and it consists of several parts, such as: (a) trucks importing and exporting commodities; (b) receiving and shipping areas; (c) warehouse gates, in this example these gates are bi-directional (in/out traffic); (d) offices of employees; (e) hand pallet trucks (able to operate with shelves at level 0, level 0 represents the floor); (f) a low forklift truck (operates with shelves at levels 0–2); (g) a high forklift truck (operates with shelves at levels 0–9); $(h_1 - h_n)$ stationary racks in the warehouse, with shelves 0–9 for commodity storing, in this example n = 10; (i) a lane between racks and other warehouse space for commodity manipulation as receiving, packing, checking and others.

The warehouse is in fact described by three coordinates $\{x, y, z\}$. In the reference model, 10 columns of racks are in the warehouse. Each column has 19 racks standing next to each other and every rack has 10 shelves one above another to store pallets (0 indicates standing on the floor). The coordinate z which represents the level of shelves is not considered in this example, so the reference model of warehouse is represented as a two dimensional matrix. The warehouse space (i) is divided into x equal sized cells, where the cell size was chosen in view of the fact that it coincided with the largest dimensions of the truck (g). The speed is the most important parameter of vehicles and it is a central parameter of the time simulation when moving commodities through the warehouse. Time delay with

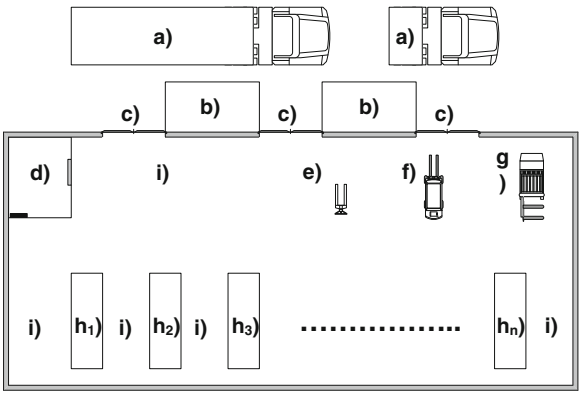


Fig. 4 Warehouse layout description

the imposition of the floor rack is now negligible. This implies that the speed of the vehicle has in this basic benchmark the most significant impact on the time of processing of the whole task buffer. The time of processing was chosen as the only fitness criterion in the work presented in this paper.

5 Results and Discussion

The benchmark was created as follows. First, the data from the warehousing company with which we cooperate under the terms of this project were obtained. Since the project is still in progress, the company must not be named because of license conditions of the contract. The data were selected according to the fully occupied time, which is the time before Christmas. During this period of time the operational manager who plans work for employees is influenced by stress conditions and is tired more than in any other season of the year. The data obtained were processed and simple scenarios were extracted. These simple scenarios put together two sets of benchmarks, 10 scenarios each. How these simple scenarios are processed is designed originally by operational manager as a reference point for results obtained by the proposed GP algorithm.

Each set of benchmarks consists of 10 simple warehouse scenarios. The first 10 scenarios contain from 2 to 4 employees with associated equipment. Special competences of employees are not considered in these scenarios. The first condition is that each employee is equipped only with a hand pallet truck. The speed of the hand pallet truck is set to 2 s.u. [speed units]. This type of truck was chosen with respect to that all kinds of warehouses contain this truck. The second condition is that each employee has his own simple task which has to be fulfilled from the very beginning to the end. And the third condition is that collisions of trucks, the performance of employees and the distance of employee and task are not taken into account when calculating the processing time. These scenarios are in fact only simplified scenarios from the second benchmark set.

Table 1 Results of benchmark set 1 and benchmark set 2

#	Manager	Optimization by GP algorithm			#	Manager	Optimization by GP algorithm		
		Op.1	Op.2	Op.1&2			Op.1	Op.2	Op.1&2
01	13.00	15.50 ↘	13.00 →	13.00 →	11	08.00	11.50 ↘	08.00 →	08.00 →
02	16.50	16.50 →	16.50 →	16.50 →	12	14.00	14.50 ↘	14.50 ↘	14.50 ↘
03	13.00	28.50 ↘	28.50 ↘	28.50 ↘	13	13.00	15.50 ↘	13.88 ↘	14.63 ↘
04	16.50	16.50 →	18.50 ↘	16.50 →	14	14.00	12.50 ↗	12.25 ↗	12.25 ↗
05	12.50	12.50 →	12.50 →	12.50 →	15	11.00	11.00 →	11.00 →	11.00 →
06	14.50	26.50 ↘	26.50 ↘	26.50 ↘	16	14.50	16.50 ↘	15.00 ↘	15.00 ↘
07	15.00	15.00 →	15.00 →	15.00 →	17	15.00	11.50 ↗	11.50 ↗	11.50 ↗
08	09.00	08.00 ↗	08.00 ↗	08.00 ↗	18	08.50	08.00 →	08.00 →	08.00 →
09	13.00	13.00 →	12.50 ↗	14.00 ↘	19	13.00	12.50 ↗	12.00 ↗	12.00 ↗
10	16.50	16.00 ↗	16.00 ↗	16.00 ↗	20	16.50	12.13 ↗	13.00 ↗	12.13 ↗

The second 10 scenarios consist of simple situations extracted from warehousing data, and at least one truck is a low forklift truck. The low forklift truck speed is set to 8 s.u. [speed units], so the scenarios are processed differently. The tasks in each scenario can be finished in different time, which implies that the collisions between the trucks can arise, e.g. Set 1 Scenario 1 was processed by employee with truck no. 1 (hand pallet), but now the truck no. 1 is low forklift truck. The speed units are used because the trucks have been standardized to two groups according to its type and employee performance.

The fitness function is calculated as follows. The final task of processing is given by the time when the last task in the scenario is finished. The time of processing each task is $T = S/R$, where T stands for the time of processing the task in t.u. [time units], S stands for the path length given in the number of cells in the warehouse which has to be exceeded when the truck is moving the commodity [cells], and R stands for the truck speed in s.u. [speed units].

5.1 Experimental Results

The first experimental results are shown in Table 1. While the arrow oriented to the upper row represents an increase in performance, the arrow oriented to the lower row represents a decrease in performance. The arrow oriented to the right shows the same performance level as that of the warehouse operational manager.

Table 1 left part, shows that the automated process of optimization reached comparable results to the operational manager. It is obvious that neither both the operators nor their combination reached the same or better results than the manager in all cases, but the first results can be considered comparable. Scenarios 1, 2, 4, 5, and 7 show the same level of performance as the manager, scenarios 8, 9, and 10 show better performance, and scenarios 3 and 6 show that the proposed algorithm failed in these cases. Table 1, right part provides little more interesting

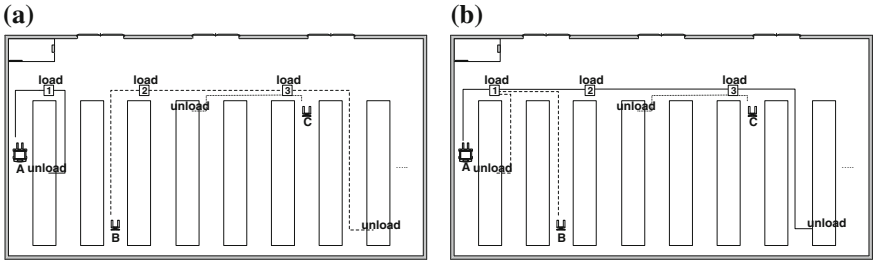


Fig. 5 Example—design by operational manager

results, because these scenarios also use a low forklift truck. Scenarios 11, 15, and 18 show the same level of performance as the manager, scenarios 14, 17, 19, and 20 show the performance increase, and scenarios 12, 13, and 16 show the performance decrease, but not that significantly as in the first set of benchmarks. Especially scenarios 3 and 6 from the first set of benchmarks will be subject to further testing and improving of the designed system.

5.2 Concrete Example of Scenario

The benchmark scenarios represent pieces of work-plans from historical data of the real logistic warehouse. One of these pieces is described in more details in the following text. The scenario is from the Christmas period when the operational manager is influenced by stress conditions and even a quite simple example can be designed in a non-optimal way. The scenario as it was designed by the operational manager is shown in Fig. 5a and the scenario designed by automated system is depicted in Fig 5b. The scenario contains three trucks (two hand trucks, one low forklift truck) associated to employees, and three pallets to store.

Figure 5a shows how the operational manager solve the situation in warehouse. The operational manager sends each employee (marked as A, B, and C) to manage one task (pallets are marked as 1, 2, and 3). Employee A processes pallet 1, paths of employee A are depicted by solid lines. Employee B processes pallet 2, paths of employee B are depicted by dashed lines, and employee C processes pallet 3, paths of this employee are depicted by dotted lines. The total time of this scenario is 16.50 t.u. (see Table 1 scenario 20). Total time is computed as a time when the last task (pallet in this case) has been processed.

Figure 5b shows how the proposed automated method solve the situation in warehouse. The resulting system distributed tasks also into three work-plans. In this case, the tasks of employee A and employee B have been switched. Employee A processes pallet 2, paths of employee A are depicted by solid lines. Employee B processes pallet 1, paths of employee B are depicted by dashed lines, and employee C processes pallet 3 with no change. Because the employee A has

associated truck with higher speed, it is better to process the more distant tasks. Therefore, the whole scenario is processed in 12.13 t.u. (see Table 1 column Op. 1&2). This looks like almost unimportant optimization, but it is only a small part of the huge and complex unit of work (e.g. a few minutes of the day in a warehouse). So, if the scenario like this is optimized just a little bit every few minutes, the total cost and the time of processing can be reduced.

6 Conclusion

In this paper a novel job-shop scheduling problem has been introduced, including the benchmark definition and baseline results reached by the genetic programming algorithms with support of the operational manager domain knowledge. This work was encouraged by a demand from the logistic distribution and warehousing industry and was created with the help and intensive consultation with experts involved in the logistic warehouse process optimization for many years. In total, 20 benchmark scenarios were created, each of them has multiple tasks with different optimization problems. The aim of this work is to set a common platform for collaborative research of the logistic warehouse process scheduling and to help solve the problem automatically or to simplify decision process.

Acknowledgments This research work is funded by projects SIX CZ.1.05/2.1.00/03.0072, MPO FR-TI1/444, and project FEKT-S-11-17.

References

1. de Koster, R., Le-Duc, T., Roodbergen, K.J.: Design and control of warehouse order picking: a literature review. Eur. J. Oper. Res. **182**(2), 481–501 (2007)
2. Geraldes, C.A.S., Sameiro, M., Carvalho, F., Pereira, G.A.B.: A warehouse design decision model case study. In: IEEE International Engineering Management Conference, IEMC Europe, pp. 397–401 (2008)
3. Būlbūl, K., Kaminsky, P.: A linear programming-based method for job shop scheduling. J. Sched. **16**(2), 161–183 (2013)
4. van Laarhoven, P.J.M., Aarts, E.H.L., Lenstra, J.K.: Job shop scheduling by simulated annealing. Oper. Res. **40**(1), 113–125 (1992)
5. Tasgetiren, M.F., Liang, Y-Ch., Sevkli, M., Gencyilmaz, G.: A particle swarm optimization algorithm for makespan and total flowtime minimization in the permutation flowshop sequencing problem. Eur. J. Oper. Res. **177**(3), 1930–1947 (2007)
6. Nowicki, E., Smutnicki, C.: An advanced tabu search algorithm for the job shop problem. J. Sched. **8**(2), 145–159 (2005)
7. Kōskolan, M., Keha, A.B.: Using genetic algorithm for single-machine bicriteria scheduling problems. Eur. J. Oper. Res. **145**(3), 543–556 (2003)
8. Benes, R., Karasek, J., Burget, R., Riha, K.: Automatically designed machine vision system for the localization of CCA transverse section in ultrasound images. Comput. Methods Programs Biomed. **109**(1), 92–103 (2013)

9. Burget, R., Karasek, J., Smekal, Z.: Recognition of emotions in Czech newspaper headlines. Radioengineering **20**(1), 39–47 (2011)
10. Karasek, J., Burget, R., Morsky, O.: Towards an automatic design of non-cryptographic hash function. In: 34th International Conference on Telecommunications and Signal Processing, pp. 19–23 (2011)

A New Approach to Solve the Software Project Scheduling Problem Based on Max–Min Ant System

Broderick Crawford, Ricardo Soto, Franklin Johnson, Eric Monfroy and Fernando Paredes

Abstract This paper presents a new approach to solve the Software Project Scheduling Problem. This problem is NP-hard and consists in finding a worker-task schedule that minimizes cost and duration for the whole project, so that task precedence and resource constraints are satisfied. Such a problem is solved with an Ant Colony Optimization algorithm by using the Max–Min Ant System and the Hyper-Cube framework. We illustrate experimental results and compare with other techniques demonstrating the feasibility and robustness of the approach, while reaching competitive solutions.

Keywords Software engineering · Software project scheduling problem · Project management · Ant colony optimization · Max–Min ant system

B. Crawford · R. Soto · F. Johnson (✉)
Pontificia Universidad Católica de Valparaíso, Valparaíso, Chile
e-mail: franklin.johnson.p@mail.ucv.cl; franklin.johnson@upla.cl

R. Soto
e-mail: ricardo.soto@ucv.cl

B. Crawford
e-mail: broderick.crawford@ucv.cl

F. Johnson
Universidad de Playa Ancha, Valparaíso, Chile

B. Crawford
Universidad Finis Terrae, Santiago, Chile

R. Soto
Universidad Autónoma de Chile, Temuco, Chile

E. Monfroy
CNRS, LINA, Université de Nantes, Nantes, France
e-mail: eric.monfroy@univ-nantes.fr

F. Paredes
Universidad Diego Portales, Santiago, Chile
e-mail: fernando.paredes@udp.cl

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_4,

1 Introduction

In this paper, we present a new approach to the Software Project Scheduling Problem (SPSP), which consists in finding a worker-task schedule that minimizes cost and duration for the whole project, so that task precedence and resource constraints are satisfied [2]. This problem is known to be NP-hard, being difficult to solve it by a complete search method in a limited amount of time. We propose then to solve the problem with Ant Colony Optimization (ACO) [10], in particular with the Max–Min Ant System and the Hyper-Cube framework [12, 13]. ACO is a probabilistic method, inspired from the behavior of real ant colonies searching for food. Ants initially explore at random, but once food is found they return to the colony leaving a pheromone trail, which can therefore be followed by other ants to reach the food quickly. The Max–Min Ant System is a popular variation of the classic ACO algorithm which we tune with the ACO Hyper-Cube (ACO-HC). This combination allows one to automatically handle the limits of pheromone values by a modification in the update pheromone rule, resulting in a more robust and easier algorithm to implement [14]. We illustrate encouraging experimental results where our approach noticeably competes with other well-known optimization methods reported in the literature.

This paper is organized as follows. In Sect. 2 presents the definition of SPSP, in Sect. 3 presents a description ACO-HC for SPSP. In its subsection presents the construction graph, pheromone update rules, and the heuristic information. In Sect. 4 presents the experimental results, The conclusions are outlined in Sect. 5.

2 The Software Project Scheduling Problem

The software project scheduling problem is one of the most common problems in managing software engineering projects [16]. It consists in finding a worker-task schedule for a software project [3, 18]. The most important resources involved in SPSP are; the tasks, which is the job needed for completing the project, the employees who work in the tasks, and finally the skills.

Description of Skills: As mentioned above, the skills are the abilities required for completing the tasks, and the employees have all or some of these abilities. These skills can be for example: design expertise, programming expert, leadership, GUI expert. The set of all skills associated with software project is defined as $S = \{s_1, \ldots, s_{|S|}\}$, where $|S|$ is the number of skills.

Description of Tasks: The tasks are all necessary activities for accomplishing the software project. These activities are for example, analysis, component design, programming, testing. The software project is a sequence of tasks with different precedence among them. Generally, we can use a graph called task-precedence-graph (TPG) to represent the precedence of these tasks [5]. This is a non-cyclic directed graph denoted as $G(V, E)$. The set of tasks is represented by

$V = \{t_1, t_2, \ldots, t_{|T|}\}$. The precedence relation of tasks is represented by a set of edges E. An edge $(t_i, t_j) \in E$, means t_i is a direct predecessor task t_j. Consequently, the set of tasks necessary for the project is defined as $T = \{t_1, \ldots, t_{|T|}\}$, where $|T|$ is the maximum number of tasks. Each task has two attributes: t_j^{sk} is a set of skills for the task j. It is a subset of S and corresponds to all necessary skills to complete a task j, t_j^{eff} is a real number and represents the workload of the task j.

Description of Employees: The problem is to create a worker-task schedule, where employees are assigned to suitable tasks. The set of employees is defined as $EMP = \{e_1, \ldots, e_{|E|}\}$, where $|E|$ is the number of employees working on the project. Each employee has tree attributes: e_i^{sk} is a set of skills of employee i. $e_i^{sk} \subseteq S$, $e_i^{\max d}$ is the maximum degree of work. it is the ratio between hours for the project and the workday. $e_i^{\max d} \in [0, 1]$, if $e_i^{\max d} = 1$ the employee has total dedication to the project, if the employee has a $e_i^{\max}$ less that one, in this case is a part-time job, e_i^{rem} is the monthly remuneration of employee i.

Model Description: The SPSP solution can be represented as a matrix $M = [E \times T]$. The size $|E| \times |T|$ is the dimension of matrix determined by the number of employees and the number of tasks. The elements of the matrix $m_{ij} \in [0, 1]$, correspond to real numbers, which represent the degree of dedication of employee i to task j. If $m_{ij} = 0$, the employee i is not assigned to task j. If $m_{ij} = 1$, the employee i works all day in task j.

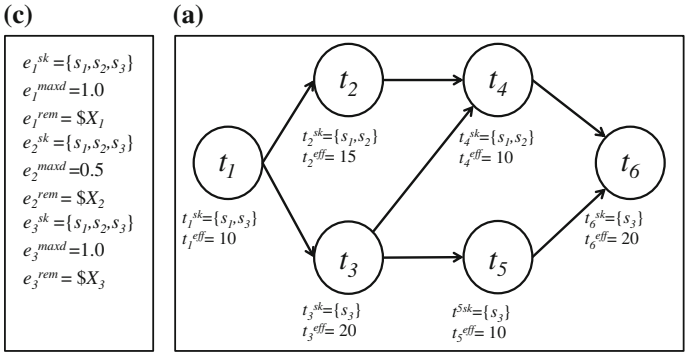
The solutions generated in this matrix M are feasible if they meet the following constraints. Firstly, all tasks are assigned at least one employee as is presented in Eq. 1. Secondly, the employees assigned to the task j have all the necessary skills to carry out the task, it is presented in Eq. 2.

$$\sum_{i=1}^{|E|} m_{ij} > 0 \; \forall j \in \{1, \ldots, T\} \tag{1}$$

$$t_j^{sk} \subseteq \bigcup_{i \mid m_{ij} > 0} e_i^{sk} \; \forall j \in \{1, \ldots, T\} \tag{2}$$

We represent in Fig. 1a an example for the precedence tasks TPG and their necessary skills t^{sk} and effort t^{eff}. For the presented example we have a set of employees $EMP = \{e_1, e_2, e_3\}$, and each one of these have a set of skills, maximum degree of dedication, and remuneration. A solution for problem represented in Fig. 1a, c is depicted in Fig. 1b.

First, it should be evaluated the feasibility of the solution, then using the duration of all tasks and cost of the project, we appraise the quality of the solution. We compute the length time for each task as t_j^{len}, $j \in \{1, \ldots, |T|\}$, for this we use matrix M and t_j^{eff} according the following formula:



(b)

	t_1	t_2	t_3	t_4	t_5	t_6
e_1	1.00	0.50	0.00	0.25	0.00	0.00
e_2	0.25	0.50	1.00	0.50	0.00	0.25
e_3	0.50	1.00	0.00	0.50	1.00	0.25

Fig. 1 **a** Task precedence graph TPG. **b** A possible solution for matrix M. **c** Employees information

$$t_j^{len} = \frac{t_j^{eff}}{\sum_{i=1}^{|E|} m_{ij}} \tag{3}$$

Now we can obtain the initialization time t_j^{init} and the termination time t_j^{term} for task j. To calculate these values, we use the precedence relationships, that is described as TPG $G(V, E)$. We must consider tasks without precedence, in this case the initialization time $t_j^{init} = 0$. To calculate the initialization time of tasks with precedence firstly we must calculate the termination time for all previous tasks. In this case t_j^{init} is defined as $t_j^{init} = \max\{t_l^{term} \| (t_l, t_j) \in E\}$, the termination time is $t_j^{term} = t_j^{init} + t_j^{len}$.

Now we have the initialization time t_j^{init}, the termination time t_j^{term} and the duration t_j^{len} for task j with $j = \{1, \ldots, |T|\}$, that means we can generate a Gantt chart. For calculating the total duration of the project, we use the TPG information. To this end, we just need the termination time of last task. We can calculate it as $p^{len} = \max\{t_l^{term} \| \forall l \neq j(t_j, t_l)\}$. For calculating the cost of the whole project, we need firstly to compute each cost associate to task us t_j^{cos} with $j \in \{1, \ldots, |T|\}$, and then the total cost p^{cos} is the sum of costs according to the following formulas:

$$t_j^{cos} = \sum_{i=1}^{E} e_i^{rem} m_{ij} t_j^{len} \tag{4}$$

$$p^{cos} = \sum_{j=1}^{T} t_j^{cos} \tag{5}$$

The target is to minimize the total duration p^{len} and the total cost p^{cos}. Therefore a fitness function is used, where w^{cos} and w^{len} represent the importance of p^{cos} and p^{len}. Then, the fitness function to minimize is given by $f(x) = (w^{cos} p^{cos} + w^{len} p^{len})$.

An element not considered is the overtime work that may increase the cost and duration associated to a task, consequently increase p^{cos} and p^{len} of the software project. We define the overtime work as e_i^{overw} as all work the employee i less $e_i^{\max d}$ at particular time.

To obtain the project overwork p^{overw}, we must consider all employees. We can use the following formula:

$$p^{overw} = \sum_{i=1}^{|E|} e_i^{overw} \tag{6}$$

With all variables required, we can determine if the solution is feasible. In this case, it is feasible when the solution can complete all tasks, and there is no overwork, that means the $p^{overw} = 0$.

3 ACO Hyper-Cube Framework for Schedule Software Project

The ACO algorithm exploits an optimization mechanism for solving discrete optimization problems in various engineering domains [11]. This framework implements ACO algorithms, which explicitly defines the multidimensional space for the pheromone values as the convex hull of the set of 0–1 coded feasible solutions of the combinatorial optimization problem under consideration. The Hyper-Cube was proposed by Blum and Dorigo [4, 8]. This framework makes a modification in the pheromone update rule, which is obtained through a normalization of the original pheromone update equation. This allows a more robust and autonomous handling of pheromone values to improve the exploration of the solution space.

To adapt ACO to SPSP using the Hyper-Cube Framework [13] must establish an appropriate construction graph and define the use of pheromone for Max–Min Ant System (MMAS) as well as heuristic information associated with the specified problem [1, 6]. The Max–Min Ant System is an Ant Colony Optimization algorithm [17], which establishes a minimum and maximum value for the pheromone, and provides that only the best ant can update the pheromone trail.

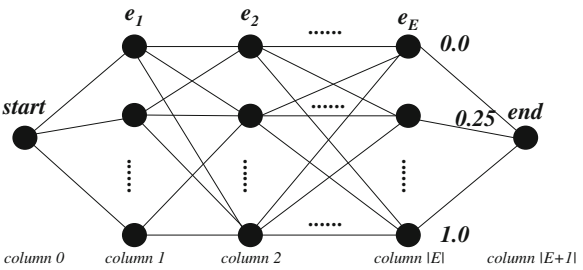


Fig. 2 Construction graph is a matrix $CG = [den \times E]$ with $mind = 0.25$ for a task

Construction Graph: For constructing a solution the ants travel through the construction graph. The ants start from an initial node and then select the nodes according to a probability function. This function is given by the pheromone and heuristic information of the problem, their relative influence is given by α and β respectively.

The proposed construction graph represents the association of employee and their dedication to a task. This representation is constructed for each task in the TGP; it is split into a graph with node and edge. The construction graph consists of each employee and their ratio of dedication contributions for the task. It is defined as *den*, this variable is density of nodes and it is defined as $den = \frac{1}{mind} + 1$, where *mind* is the lowest degree of dedication to a task. This structure is presented in Fig. 2. The employees dedication to a task can be 0 or integer multiple of *mind*.

The ants travel to the start node to the end node choosing edges from the column 1 to column |*E*| without returning. The ants choose only one node per each column. When the ant completes a tour, the dedication distribution of employees to the task is complete. To calculate the dedication of the employee *i* to the task, we just need the node *j*, with column *i* and the calculation is *j* * *mind*. These activities of ants must be done for each task in SPSP model.

The ants travel through the construction graph selecting ways of probabilistically way, using the following function:

$$p_{ij}^{t} = \frac{\left[\tau_{ij}\right]^{\alpha}\left[\eta_{ij}\right]^{\beta}}{\sum_{l=0}^{den}\left[\tau_{il}\right]^{\alpha}\left[\eta_{il}\right]^{\beta}}, \quad j \in \{1, \ldots, den\} \tag{7}$$

where τ_{ij} is the pheromone and η_{ij} is the heuristic information of the problem on the path between node *i* to *j* in the graph *CG* for de *t* task. α and β are two fixed parameters, which are used to determine the pheromone and heuristics influences.

Pheromone Update: In the hyper-cube framework the pheromone trails are forced to stay in the interval [0, 1] and the Max–Min Ant System the pheromone stay in the interval $[\tau_{\min}, \tau_{\max}]$. To adapt the Hyper-Cube framework to MMAS we define a $\tau_{\min} = 0$ and, $\tau_{\max} = 1$. In MMAS only one single ant is used to update the pheromone trails after each iteration, that ant can be the *iteration—best* ant or *global—best* ant. In this case we used only *iteration—best* ant.

We computationally represent the evaporation of pheromone and in addition the amount of pheromone in the ant path through the graph once a tour is completed using the following formula:

$$\tau_{ij} = \rho\tau_{ij} + (1 - \rho)\Delta\tau^{upd}, \tag{8}$$

where ρ is a rate of evaporation $\rho \in [0, 1]$. If ρ is high, the new pheromone value is less influenced by $\Delta\tau^{upd}$, but much influenced by the previous pheromone value, vice versa. And $\Delta\tau^{upd}$ it is associated with quality of the current solution of *upd* ant. *upd* ant is *iteration—best* ant [13]. We can use an updating pheromone strategy considering the duration, cost, and overwork of the project as follows:

$$\Delta\tau^{upd} = (w^{cos}p^{cos} + w^{len}p^{len} + w^{overw}p^{overw})^{-1}, \tag{9}$$

where w^{cos}, w^{len}, and w^{overw} are values that weight the importance of p^{cos}, p^{len}, and p^{overw} of the software project. The Δupd is the amount of pheromone added based on the quality of solution generated by *upd* ant.

Heuristic Information: We need to represent the heuristic information, that information is used to enhance the search ability of ants. The ants need to find the proper nodes using the problem information. The ants travel for a matrix m_{ij} as the node at column i and row j. To obtain the dedication of an employee e_i to a task which has been selected, we must calculate $j * minded$. We use as heuristic information the dedication of employee e_i to other task. If an employee works more in the previous tasks, that employee will has less dedication available for subsequent tasks. Consequently, the employee has less probability to be assigned to the current task. The heuristic information $h[i]$ to select node i for task t_k can be calculated as follows:

$$h[i] = \begin{cases} \frac{tmp[den-i-1]}{sum}, & \text{if} \quad allocD[k] > 0.5, \\ \frac{tmp[i]}{sum}, & else \end{cases} \tag{10}$$

where $tmp = \{1, \ldots, den\}$ is an array of temporal values generated with summation of possible dedication and allocated dedications for employees ($allocD[k]$, for k employee). *sum* is the summation of all values of the array *tmp*.

ACO-HC algorithm: The algorithm firstly reads the problem instance. That instance provides all the necessary data to generate the SPSP, such as number of tasks, and their required skills, number of employees, task precedence information to generate the TGP, the set of skills of every employee, and their remunerations. Then, we have to split operation to the task and then using the ACO-HC to generate solutions. To determine the quality of solutions, we use the fitness function. That function minimizes the cost and duration for whole project.

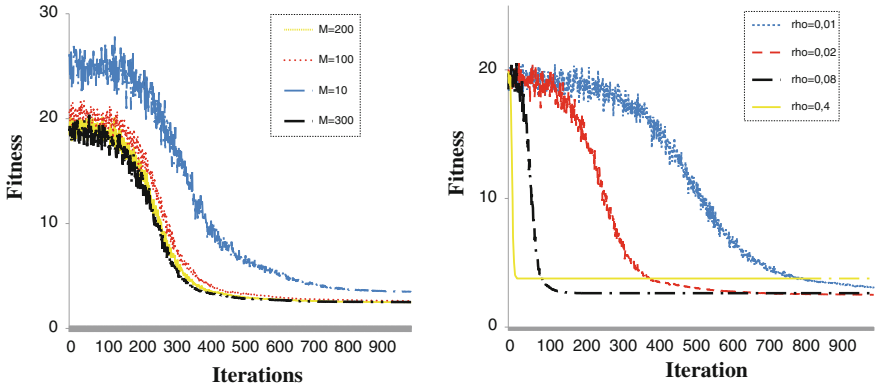


Fig. 3 Avg fitness using different values for m and ρ

Algorithm 1 ACO-HC for SPSP Algorithm

```
initialize feromone values τ_max and τ_min
repeat
    for g = 1 to G do
        for a = 1 to M do
            for t = 1 to T do
                the a ant travel on the matrix
            end for
            compute the feasibility of a ant and fitness of the solution
        end for
        select the best solution
        update pheromone values
    end for
until (iterations or time) is complete
```

4 Experimental Results

In this section we present the experimental results. The algorithm was ran 10 trials for each instance and we report the average value from those 10 trials. For the experiments, we use a random instances created by a generator.[1] The instances are labelled as <tasknumber> t<employeesnumber> e<skillsnumber> s. To compare the different results we use the *hit rate*: feasibles solution in 10 runs, *cost*: average cost of feasible solutions in 10 runs, *duration*: average duration of feasible solutions in 10 runs, and *fitness*: average fitness of feasible solutions in 10 runs.

[1] http://tracer.lcc.uma.es/problems/psp/generator.html

Table 1 Comparison with other techniques

Instance	Algorithms	Hit rate	Fitness	p^{len}	p^{cos}
10t5e5s	ACO-HC	100	**3.136531**	23	836531
	ACS	100	3.42039	22	1220390
	GA	95	3.52431	23	1224310
10t10e5s	ACO-HC	100	**2.134546**	13	834546
	ACS	100	2.55633	14	1156330
	GA	97	2.81331	16	1213310
20t10e5s	ACO-HC	30	6.741111	45	2241111
	ACS	67	6.36584	39	2465840
	GA	19	**6.28734**	38	2487340
10t5e10s	ACO-HC	100	**2.90579**	21	805790
	ACS	100	3.39316	22	1193160
	GA	90	3.51354	23	1213540
10t10e10s	ACO-HC	100	2.612948	17	912948
	ACS	100	2.62331	14	1223310
	GA	100	**2.51203**	13	1212030
20t10e10s	ACO-HC	50	6.249782	42	2049782
	ACS	65	6.31455	38	2514550
	GA	71	**6.19601**	37	2496010

4.1 Parameter Tunning and Convergence Analysis

It is known that the ACO results may vary depending on the parameters used. For this reason is important to make a simple but significant test. In this case we conducted a series of experiments to find the best parameter values for number of ants m and evaporation rate ρ. We used 10t10e10 s instance with $mind = 0.25$ and some parameters are constant which are $\alpha = 1$, $\beta = 2$ number of iterations $N_{it} = 1{,}000$ the results is presented in Fig. 3.

We can observe (left chart in Fig. 3) that the best fitness (low fitness) is obtained with $m = 200$ and $m = 300$ and the worst fitness is obtained with $m = 100$ and $m = 10$. To obtain the best results in shortest time, we used $m = 200$. In the right chart in Fig. 3 we observe the different fitness obtained with different values for ρ. For this parameter the best fitness is obtained with $\rho = 0.02$. When $\rho = 0.01$ the fitness converge too slow, if the ρ is very large (0.08 or 0.4) the fitness converges very fast to a suboptimal value, with $\rho = 0.02$ converges smoothly to fitness better value.

In order to analyse the convergence to feasible solutions of the algorithm, we can observe in Fig. 3 how to obtain better solutions from iteration 300 and converges slowly to a best solution.

4.2 Comparative Results with Other Techniques

Some results are presented in [18] by Xiao, using the similar parameter to our instances. Xiao presents results using Ant Colony System (ACS) and Genetic Algorithms (GA). For the sake of clarity we transform the fitness presented by the autor as fitness^{-1} to obtain the same fitness used by us. The comparative results are presented in Table 1.

From Table 1 we can compare the hit rate and the fitness of the solutions. In this case for the instances with task $= 10$ always have a hit rate of 100 % for all numbers of employees or skills. But in the instances with task $= 20$, ACS has better hit rate.

Regarding the fitness we can see that ACO-HC has better results for all instances with task $= 10$. For instances with task $= 20$, the best results are using GA. If we analyse the results based on project cost, we can see that our proposal provides the best results for all instances compared.

5 Conclusion

We presented an overview to the resolution of the SPSP using an ACO-HC framework. We design a representation of the problem in order to ACO algorithm can solve it, proposing a construction graph and a pertinent heuristic information. Furthermore, we defined a fitness function able to allow optimization of the generated solutions.

We implement our proposed algorithm, and we conducted a series of tests to analyse the convergence to obtain better solutions. In addition we realized different tests to get the best parameterization. The tests were performed using different numbers of tasks, employees, and skills. The results were compared with other techniques such as Ant Colony System and Genetic Algorithms. We demonstrate that our proposal gives the best results for smaller instances. For more complex instances was more difficult to find solutions, but our solutions always obtained a low cost of the project, in spite of increasing the duration of the whole project.

An interesting research direction to pursue as future work is about the integration of autonomous search in the solving process, which in many cases has demonstrated excellent results [7, 9, 15]

References

1. Abdallah, H., Emara, H.M., Dorrah, H.T., Bahgat, A.: Using ant colony optimization algorithm for solving project management problems. Expert Syst. Appl. **36**(6), 10004–10015 (2009)
2. Alba, E., Chicano, F.: Software project management with gas. Inf. Sci. **177**(11), 2380–2401 (2007) (in press)

3. Barreto, A., Barros, MdO, Werner, C.M.L.: Staffing a software project: a constraint satisfaction and optimization-based approach. Comput. Oper. Res. **35**(10), 3073–3089 (2008)
4. Blum, C., Dorigo, M.: The hyper-cube framework for ant colony optimization. Syst. Man Cybern. Part B Cybern. IEEE Trans. **34**(2), 1161–1172 (2004)
5. Chang, C.K., yi Jiang, H., Di, Y., Zhu, D., Ge, Y.: Time-line based model for software project scheduling with genetic algorithms. Inf. Softw. Technol. **50**(11), 1142–1154 (2008)
6. Chen, W., Zhang, J.: Ant colony optimization for software project scheduling and staffing with an event-based scheduler. Softw. Eng. IEEE Trans. **39**(1), 1–17 (2013)
7. Crawford, B., Soto, R., Castro, C., Monfroy, E.: Extensible cp-based autonomous search. In: Proceedings of HCI International, vol. 173 of CCIS, pp. 561–565. Springer (2011)
8. Crawford, B., Soto, R., Johnson, F., Monfroy, E.: Ants can schedule software projects. In: Stephanidis, C. (ed.) HCI International 2013—Posters Extended Abstracts, volume 373 of Communications in Computer and Information Science, pp. 635–639. Springer, Berlin (2013)
9. Crawford, B., Soto, R., Monfroy, E., Palma, W., Castro, C., Paredes, F.: Parameter tuning of a choice-function based hyperheuristic using particle swarm optimization. Expert Syst. Appl. **40**(5), 1690–1695 (2013)
10. Dorigo, M. Di Caro, G.: Ant colony optimization: a new meta-heuristic. In: Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on, vol. 2, p. 1477 (1999)
11. Dorigo, M., Gambardella, L.M.: Ant colony system: a cooperative learning approach to the traveling salesman problem. IEEE Trans. Evol. Comput. **1**(1), 53–66 (1997)
12. Dorigo, M., Stutzle, T.: Ant Colony Optimization. MIT Press, USA (2004)
13. Johnson, F., Crawford, B., Palma, W.: Hypercube framework for ACO applied to timetabling. In: IFIP AI, pp. 237–246 (2006)
14. Liao, T.W., Egbelu, P., Sarker, B., Leu, S.: Metaheuristics for project and construction management a state-of-the-art review. Autom. Constr. **20**(5), 491–505 (2011)
15. Monfroy, E., Castro, C., Crawford, B., Soto, R., Paredes, F., Figueroa, C.: A reactive and hybrid constraint solver. J. Exp. Theor. Artif. Intell. **25**(1), 1–22 (2013)
16. Ozdamar, L., Ulusoy, G.: A survey on the resource-constrained project scheduling problem. IIE Trans. **27**(5), 574–586 (1995)
17. Stutzle, T., Hoos, H.H.: Maxmin ant system. Future Gener. Comput. Syst. **16**(8), 889–914 (2000)
18. Xiao, J., Ao, X.T., Tang, Y.: Solving software project scheduling problems with ant colony optimization. Comput. Oper. Res. **40**(1), 33–46 (2013)

An Artificial Bee Colony Algorithm for the Set Covering Problem

Rodrigo Cuesta, Broderick Crawford, Ricardo Soto and Fernando Paredes

Abstract In this paper, we present a new Artificial Bee Colony algorithm to solve the non-unicost Set Covering Problem. The Artificial Bee Colony algorithm is a recent metaheuristic technique based on the intelligent foraging behavior of honey bee swarm. Computational results show that Artificial Bee Colony algorithm is competitive in terms of solution quality with other metaheuristic approaches for the Set Covering Problem problem.

Keywords Artificial bee colony algorithm · Combinatorial optimization · Heuristic · Set covering problem

R. Cuesta · B. Crawford (✉) · R. Soto
Pontificia Universidad Católica de Valparaíso, Valparaiso, Chile
e-mail: broderick.crawford@pucv.cl

R. Cuesta
e-mail: rodrigo.cuesta.a@mail.pucv.cl

R. Soto
e-mail: ricardo.soto@pucv.cl

B. Crawford
Universidad Finis Terrae, Santiago, Chile

R. Soto
Universidad Autónoma de Chile, Santiago, Chile

F. Paredes
Universidad Diego Portales, Santiago, Chile
e-mail: fernando.paredes@udp.cl

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_5,

1 Introduction

The Set Covering Problem (SCP) is a classic problem in combinatorial analysis, sciences of the computation and theory of the computational complexity. It is a problem that has led to the development of fundamental technologies for the field of the algorithms of approximation. Also it is one of the problems of the List of 21 Karp's NP-complete problems whose NP-completeness it was demonstrated in 1972. Many algorithms have been developed to solve it. Exact algorithms are mostly based on branch-and-bound [1] and branch-and-cut [2]. However, these algorithms are rather time consuming and can only solve instances of very limited size. For this reason, many research efforts have been focused on the development of heuristics to find good or near-optimal solutions within a reasonable period of time. Classical greedy algorithms are very simple, fast, and easy to code in practice, but they rarely produce high quality solutions for their myopic and deterministic nature [3]. In [4] improved a greedy algorithm by incorporating randomness and memory into it and obtained promising results. Compared with classical greedy algorithms, heuristics based on Lagrangian relaxation with sub-gradient optimization are much more effective. The most efficient ones are those proposed by Ceria et al. [5] and Caprara et al. [6]. As top-level general search strategies, metaheuristics were also applied to the SCP. An incomplete list of this kind of metaheuristics for the SCP includes genetic algorithm [7], simulated annealing algorithm [8], and tabu search algorithm [9]. For a deeper comprehension of most of the effective algorithms for the SCP in the literature, we refer the interested reader to the survey by Caprara et al. [10].

In this paper we present the metaheuristics Artificial Bee Colony (ABC) that is relatively new in the area and inside which have not been observed attempts of solving the SCP. Researches on ABC for SCP not been seen to date.

2 Problem Description

2.1 Set Covering Problem

Problem Definition A general mathematical model of the Set Covering Problem can be formulated as follows:

$$Minimize\ Z = \sum_{j=1}^{n} c_j x_j \quad j = \{1, 2, 3, \ldots, n\} \tag{1}$$

Subject to:

$$\sum_{j=1}^{n} a_{ij} x_j \geq 1 \quad i = \{1, 2, 3, \ldots, m\} \tag{2}$$

$$x_j = \{0, 1\} \tag{3}$$

Equation 1 is the objective function of SCP, where c_j refers to the weight or cost of j-column, and x_j is the decision variable. Equation 2 is a constraint to ensure that each row is covered by at least one column, where a_{ij} is a constraint coefficient matrix of size m x n whose elements can be "1" or "0". Finally, Eq. 3 is the integrality constraint in which the value x_j can be "1" if column j is activated (selected) or "0" otherwise.

Different solving methods have been proposed in the literature for the set covering problem. There exist examples using exact methods [11], linear programing and heuristic methods [12], and metaheuristic methods [13–16]. Has being pointed out, that one of the most relevant applications of SCP is given by crew scheduling problems in mass transportation companies where a given set of trips has to be covered by a minimum-cost set of pairings, a pairing being a sequence of trips that can be performed by a single crew.

3 Artificial Bee Colony Algorithm

ABC is one of the most recent algorithms in the domain of the collective intelligence. Created by Dervis Karaboga in 2005, who was motivated by the intelligent behavior observed in the domestic bees to take the process of forage [17].

ABC is an algorithm of combinatorial optimization based on populations, in which the solutions of the problem of optimization, so called sources of food, are modified by the artificial bees, that fungen as operators of variation. The aim of these bees is to discover the food sources with major nectar.

In the algorithm ABC, the artificial bees move in a space of multidimensional search choosing sources of nectar depending on his past experience and that of his companions of beehive or fitting his position. Some bees (exploratory) fly and choose food sources randomly without using experience. When they find a source of major nectar, they memorize his position and forget the previous one. Thus, ABC combines methods of local search and global search, trying to balance the process of the exploration and exploitation of the space of search.

3.1 Elements and Behavior

The model defines three principal components which later are enunciated:

Food source: The value of a food source depends on many factors, as his proximity to the beehive, wealth or the concentration of the energy and the facility of extraction of this energy.

Employeed Bees: They are associated with a current food source, or in exploitation. They take with them information about this source especially, his distance, location and profitability her to share, with a certain probability, to his other companions.

Exploratory bees: They are in constant search of a food source. There are two types:

- *Scout*: They take charge searching in the environment that surrounds to the beehive new sources of food.
- *In wait*: They look for a food source across the information shared by the employees or by other explorers in the nest.

3.2 Artificial Behavior

In Table 1 the elements of the ABC are described in a general way.

The procedure for determining a food source in the neighborhood of a particular food source depends on the nature of the problem. Karaboga [18] developed the first ABC algorithm for continuous optimization. His method for determining a food source in the neighborhood of a particular food source is based on changing the value of one randomly chosen solution parameter while keeping other parameters unchanged. This is done by adding to the current value of the chosen parameter the product of a uniform variate in $[-1, 1]$ and the difference in values of this parameter for this food source and some other randomly chosen food source. This approach can not be used for discrete optimization problems for which it generates at best a random effect. Singh [19] subsequently proposed a method, which is appropriate for subset selection problems. In his model, to generate a neighboring solution, an object is randomly dropped from the solution and in its place another object, which is not already present in the solution is added. The object to be added is selected from another randomly chosen solution. If there are more than one candidate objects for addition then ties are broken arbitrarily. This approach is based on the idea that if an object is present in one good solution then it is highly likely that this object is present in many good solutions. This method provide another advantage, consist in if the method fails to find an object different from the others objects in the original solution, this mean the two solutions are equals, such situation is called "Collision" and was resolved making the employed bee associated with the original solution a scout bee. This eliminates one duplicate solution.

Table 1 Summary of the elements of the original ABC and its details

Generation of the food sources	It removes in a random way and with base in the limits low and superior of every variable of the problem. A food source is a solution to the problem of optimization
Employeed bee	His number is proportional to the number of food sources, that is to say for every source there is an used bee, and his function is to evaluate and to modify the current solutions to improve them (it looks for new sources near to the current one). If the current position is not better than the current kept the original position
Bees in wait	Analogously to his number must be proportional to the number of food sources. These bees will choose a food source, with base in the information that the bees used by means of the dance share. This dance is simulated by means of a tilt of size "t", where the food source with better value of the function I object is selected
Scout bees	These bees generate a new source of food of a random way, to supplant existing sources that have not been improved
Limit	It defines the maximum number of cycles that a food source can remain without improving before being replaced. The limit increases from that a source that is not modified by the bees, already they are used or in wait, up to obtaining his maximum allowed value, after this the bees scout they take charge initializing the limit to 0 for every new generated position. The limit is initialized to 0 whenever a source is modified (improved) by an used bee or in wait
Column ADD	It defines the number of column to add to the current food source
Columns to eliminating	It defines the number of column to eliminate of the current food source

4 Description of the Proposed Approach

Step 1. Initialization

To initialize the such parameters of the ABC as size of the colony, number of bees workers and curious, limit of attempts and maximum number of cycles.

Step 2. Generation of initial population

To generate the initial population we crosses every row of the counterfoil of restrictions and by every row a column is selected at random of this one and this one happens to form a part of the solution that this one represented by means of an entire vector like appears in Fig. 1 staying the column of the solution with equal value to the column chosen to cover the row. This procedure realizes for all the rows of a such way that the generated solution expires with all the restrictions.

$$\boxed{333}\boxed{-1}\boxed{5}\boxed{-1}\boxed{88}\boxed{-1}\boxed{657}\boxed{-1}$$

Fig. 1 Representation of solution

Step 3. Evaluation of the fitness of the population

For our case the function of fitness is equal to the function aim of the SCP Eq. 1.

Step 4. Modification of position and selection of sites for the hard-working bees

A hard-working bee was modifying the position in the one that is by means of the creation of a new solution based on a source of different food to in the one that is, for this it was selecting a source aleatoriamente and it sees if at least it has a different column to in the one that is nowadays in case of having not even a different column is considered to be equal solutions what is called "collision" in this case the hard-working bee it was transforming in an explorer of way of eliminating the duplicated solutions, in opposite case one proceeds to add a certain random number of columns that they do not find in our position. After this, in our position one proceeds to eliminate a certain random number of columns to this way the population diversify. In case the new solution does not expire with the restrictions it is proceeded to repair so that the restrictions are satisfied, after this the fitness of the solution is evaluated by means of Eq. 1 and if the fitness is minor that the solution that tape-worm in a beginning replaces in opposite case increases the number of attempts for improving this solution and to pass to another hard-working bee.

Step 5. To recruit curious bees for the selected sites

A curious bee evaluates the information of the nectar through the workers and chooses a source of food with a probability Pr_i using the Eq. 4 in relation with the value fitness.

$$Pr_i = \frac{f_i}{\sum_{j=1}^{n} f_j} \tag{4}$$

Step 6. Modification of position for the curious ones

They work of equal way that the hard-working bees in Step 4.

Step 7. To leave a source exploited by the bees

If the solution representing a source of food improves for a predetermined number of attempts, then the source of food is left and is replaced by a new source of food that in our case is generated like in Step 1.

Step 8. To memorize the best opposing solution and to increase the account of the cycle
Step 9. The process stops if the criterion of satisfaction expires in opposite case to return to Step 3

5 Experiments and Results

The algorithm ABC SCP has been implemented in C in a 2.5 GHz Dual Core with 4 GB RAM computer, running windows 7. In all the experiments the ABC SCP run 1,000 iterations. We use a population of 200 bees, where 100 corresponds to hard-working and 100 to curious. Limit = 50, number of columns to add = 0,5 % of the total columns of the problem, number of columns to eliminating = 1,2 % of the total columns of the problem. These parameters threw good results but they cannot be the ideal ones for all the instances. ABC SCP has been tested on 65 standard non-unicost SCP instances available from OR-Library at http://people.brunel.ac.uk/mastjjb/jeb/info.html. Table 2 summarizes the characteristics of each of these sets, where column labeled Density shows the percentage of non-zero entries in the matrix of each instance. ABC SCP was executed 30 times on each instance, each time with a different random seed. The best value found and average of the 30 runs is shown in the Table 3.

5.1 Convergence to the Best Solution

The purpose of this chart is to allow observing how each version converges through time to a better solution. Plot (Fig. 2) is based on benchmark scp41, scp42 and scp43. A point jointly in every graph is that they converge in a very rapid way on the first iterations (Table 3).

6 Conclusions

In this paper we have presented an ABC algorithm for the SCP. We have performed experiments through several instances, where an approach has demonstrated to be very effective, providing an unattended solving method, for quickly producing solutions of a good quality.

The literature reviewed is rich in definitions and state-of-art techniques, providing useful tools for experimenting with novel approaches. Related to this it is

Table 2 Details of the test instances 65

Instance set	No. of instances	m	n	Cost range	Density (%)	Optimal solution
4	10	200	1,000	[1, 100]	2	Known
5	10	200	2,000	[1, 100]	2	Known
6	5	200	1,000	[1, 100]	5	Known
A	5	300	3,000	[1, 100]	2	Known
B	5	300	3,000	[1, 100]	5	Known
C	5	400	4,000	[1, 100]	2	Known
D	5	400	4,000	[1, 100]	5	Known
NRE	5	500	5,000	[1, 100]	10	Unknown
NRF	5	500	5,000	[1, 100]	20	Unknown
NRG	5	1000	10,000	[1, 100]	2	Unknown
NRH	5	1000	10,000	[1, 100]	5	Unknown

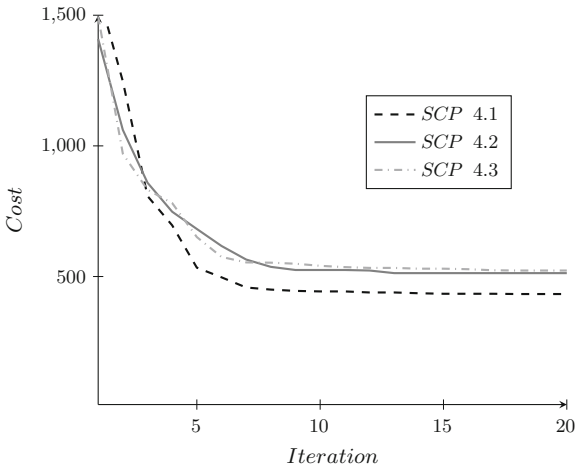


Fig. 2 Convergence analysis to a better solution. Benchmark: SCP41, SCP42, SCP43

encouraging to reach a point where there are not direct examples to get help, and where innovation and trial-and-error techniques find their ways. Benchmarks have shown interesting results in terms of robustness, where using the same parameters for different instances giving good results.

The promising results of the experiments open up opportunities for further research. An interesting proposal by Glover and Kochenberger [20] involves parallelizing strategies for metaheuristics. The author sets a basis on the idea that the central goal of parallel computing is to speed up computation by dividing the work load among several threads of simultaneous execution, then a type of metaheuristic parallelism could come from the decomposition of the decision variables into disjoint subsets. The particular heuristic is applied to each subset and the variables outside the subset are considered fixed. Another interesting research

Table 3 Computational results on 65 instances of SCP

Instance	Optimum	Best value found	Average
4.1	429	430	430.5
4.2	512	512	512
4.3	516	516	516
4.4	494	494	494
4.5	512	512	512
4.6	560	561	561.7
4.7	430	430	430
4.8	492	493	494
4.9	641	643	645.5
4.10	514	514	514
5.1	253	254	255
5.2	302	309	310.2
5.3	226	228	228.5
5.4	242	242	242
5.5	211	211	211
5.6	213	213	213
5.7	293	296	296
5.8	288	288	288
5.9	279	280	280
5.10	265	266	267
6.1	138	140	140.5
6.2	146	146	146
6.3	145	145	145
6.4	131	131	131
6.5	161	161	161
A.1	253	254	254
A.2	252	254	254
A.3	232	234	234
A.4	234	234	234
A.5	236	237	238.6
B.1	69	69	69
B.2	76	76	76
B.3	80	80	80
B.4	79	79	79
B.5	72	72	72
C.1	227	230	231
C.2	219	219	219
C.3	243	244	244.5
C.4	219	220	224
C.5	215	215	215
D.1	60	60	60
D.2	66	67	67
D.3	72	73	73
D.4	62	63	63
D.5	61	62	62

(continued)

Table 3 (continued)

Instance	Optimum	Best value found	Average
NRE.1	29	29	29
NRE.2	30	30	30
NRE.3	27	27	27
NRE.4	28	28	28
NRE.5	28	28	28
NRF.1	14	14	14
NRF.2	15	15	15
NRF.3	14	14	14
NRF.4	14	14	14
NRF.5	13	13	13
NRG.1	176	176	176
NRG.2	154	154	154
NRG.3	166	166	166
NRG.4	168	168	168
NRG.5	168	168	168
NRH.1	63	63	63
NRH.2	63	63	63
NRH.3	59	59	59
NRH.4	58	58	58
NRH.5	55	55	55

direction to pursue is about the integration of autonomous search in the solving process, which in many cases has demonstrated excellent results [21–25].

Acknowledgments The author Broderick Crawford is supported by Grant CONICYT/FONDECYT/REGULAR/1140897.

The author Ricardo Soto is supported by Grant CONICYT/FONDECYT/INI- CIACION/11130459.

The author Fernando Paredes is supported by Grant CONICYT/FONDECYT/REGULAR/1130455.

References

1. Balas, E., Carrera, M.C.: A dynamic subgradient-based branch-and-bound procedure for set covering. Oper. Res. **44**(6), 875890 (1996)
2. Fisher, M.L., Kedia, P.: Optimal solution of set covering/partitioning problems using dual heuristics. Manage. Sci. **36**(6), 674688 (1990)
3. Chvatal, V.: A greedy heuristic for the set-covering problem. Math. Oper. Res. **4**(3), 233235 (1979)
4. Lan, G., DePuy, G.W.: On the effectiveness of incorporating randomness and memory into a multi-start metaheuristic with application to the set covering problem. Comput. Ind. Eng. **51**(3), 362374 (2006)
5. Ceria, S., Nobili, P., Sassano, A.: A Lagrangian-based heuristic for large-scale set covering problems. Math. Program. **81**, 215228 (1998)

6. Caprara, A., Fischetti, M., Toth, P.: A heuristic method for the set covering problem. Oper. Res. **47**(5), 730743 (1999)
7. Beasley, J.E., Chu, P.C.: A genetic algorithm for the set covering problem. Eur. J. Oper. Res. **94**(2), 392404 (1996)
8. Brusco, M.J., Jacobs, L.W., Thompson, G.M.: A morphing procedure to supplement a simulated annealing heuristic for cost- and coverage-correlated set-covering problems. Ann. Oper. Res. **86**, 611627 (1999)
9. Caserta, M.: Tabu search-based metaheuristic algorithm for large-scale set covering problems. In: Doerner, K.F., et al. (eds.) Metaheuristics: Progress in Complex Systems Optimization, pp. 43–63. Springer, New York (2007)
10. Caprara, A., Toth, P., Fischetti, M.: Algorithms for the set covering problem. Ann. Oper. Res. **98**, 353371 (2000)
11. Beasley, J.E., Jornsten, K.: Enhancing an algorithm for set covering problems. Eur. J. Oper. Res. **58**(2), 293–300 (1992)
12. Caprara, A., Fischetti, M., Toth, P.: Algorithms for the set covering problem. Ann. Oper. Res. **98**, 2000 (1998)
13. Aickelin, U.: An indirect genetic algorithm for set covering problems, CoRR,0803.2965 (2008)
14. Crawford, B., Soto, R., Monfroy, E.: Cultural algorithms for the set covering problem. ICSI **2**, 27–34 (2013)
15. Crawford, B., Castro, C., Monfroy, E.: A new ACO transition rule for set partitioning and covering problems, pp. 426–429. In: SoCPaR 2009 (2009)
16. Crawford, B., Lagos, C., Castro, C., Paredes, F.: A evolutionary approach to solve set covering. ICEIS **2**(2007), 356–363 (2007)
17. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. J. Global Optim. **39**(3), 459–471 (2007)
18. Karaboga, D.: An idea based on honey bee swarm for numerical optimization, Technical Report TR06. Computer Engineering Department, Erciyes University, Turkey (2005)
19. Singh, A.: An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem. Appl. Soft Comput. **9**(2), 625–631 (2009)
20. Glover, F., Kochenberger, G.A.: Handbook of Metaheuristics. Springer, Berlin (2003)
21. Crawford, B., Soto, R., Monfroy, E., Palma, W., Castro, C., Paredes, F.: Parameter tuning of a choice-function based hyperheuristic using particle swarm optimization. Expert Syst. Appl. **40**(5), 1690 (2013)
22. Monfroy, E., Castro, C., Crawford, B., Soto, R., Paredes, F., Figueroa, C.: A reactive and hybrid constraint solver. J. Exp. Theor. Artif. Intell. **25**(1), 1–22 (2013)
23. Crawford, B., Castro, C., Monfroy, E., Soto, R., Palma, W., Paredes, F.: A Hyperheuristic Approach for Guiding Enumeration in Constraint Solving Advances in Intelligent Systems and Computing, p. 175. Springer, Berlin (2012)
24. Soto, R., Crawford, B., Monfroy, E., Bustos, V.: Using autonomous search for generating good enumeration strategy blends in constraint programming. In: Proceedings of the 12th International Conference on Computational Science and Its Applications (ICCSA), p 7335 (2012)
25. Crawford, B., Soto, R., Castro, C., Monfroy, E.: A hyperheuristic approach for dynamic enumeration strategy selection in constraint satisfaction. In: Proceedings of the 4th International Work-conference on the Interplay Between Natural and Artificial Computation (IWINAC), p. 668 (2011)

A Binary Firefly Algorithm for the Set Covering Problem

Broderick Crawford, Ricardo Soto, Miguel Olivares-Suárez and Fernando Paredes

Abstract The non-unicost Set Covering Problem is a well-known NP-hard problem with many practical applications. In this work, a new approach based on Binary Firefly Algorithm is proposed to solve this problem. The Firefly Algorithm has attracted much attention and has been applied to many optimization problems. Here, we demonstrate that is also able to produce very competitive results solving the portfolio of set covering problems from the OR-Library.

Keywords Set covering problem · Binary firefly algorithm · Metaheuristic

1 Introduction

The Set Covering Problem (SCP) is a class of representative combinatorial optimization problem that has been applied to many real world problems, such as crew scheduling in airlines [1], facility location problem [2], and production planning in

B. Crawford (✉) · R. Soto · M. Olivares-Suárez
Pontificia Universidad Católica de Valparaíso, Valparaíso, Chile
e-mail: broderick.crawford@ucv.cl

R. Soto
e-mail: ricardo.soto@ucv.cl

M. Olivares-Suárez
e-mail: miguel.olivares.s@mail.pucv.cl

B. Crawford
Universidad Finis Terrae, Santiago, Chile

R. Soto
Universidad Autónoma de Chile, Santiago, Chile

F. Paredes
Universidad Diego Portales, Santiago, Chile
e-mail: fernando.paredes@udp.cl

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_6,

industry [3]. The SCP is a well-known NP-hard in the strong sense [4]. Many algorithms have been developed to solve it has been reported to literature. Exact algorithms are mostly based on branch-and-bound and branch-and-cut [5, 6]. However, these algorithms are rather time consuming and can only solve instances of very limited size. For this reason, many research efforts have been focused on the development of heuristics to find good or near-optimal solutions within a reasonable period of time. Classical greedy algorithms are very simple, fast, and easy to code in practice, but they rarely produce high quality solutions for their myopic and deterministic nature [7]. An improved greedy algorithm by incorporating randomness and memory into it and obtained promising results [8]. Compared with classical greedy algorithms, heuristics based on Lagrangian relaxation with subgradient optimization are much more effective. The most efficient ones are those proposed in [9, 10]. As top-level general search strategies, metaheuristics were also applied to the SCP. An incomplete list of this kind of heuristics for the SCP includes genetic algorithm [11], simulated annealing algorithm [12], tabu search algorithm [13], evolutionary algorithms [14], ant colony optimization (ACO) [15], electromagnetism (unicost SCP) [16], gravitational emulation search [17] and cultural algorithms [18]. A deeper comprehension of most of the effective algorithms for the SCP can be found in [19].

In this paper, a new approach based on Binary Firefly Algorithm for the SCP is presented. Firefly Algorithm (FA) is a recently developed, population-based metaheuristic [20, 21]. So far, it has been shown that firefly algorithm is very efficient in dealing with multimodal, global optimization problems. For a deeper comprehension of review of firefly advances and applications please refer to [22, 23]. Researches on FA for SCP have not been seen to date.

This paper is organized as follows: In Sect. 2, we formally describe the SCP. The Sect. 3, we present the overview of FA. The description of the proposed approach is described in Sect. 3. In Sect. 5, we present experimental results obtained when applying the algorithm for solving the 65 instances different of SCP. Finally, in Sect. 6 we conclude the paper.

2 Problem Description

The Set Covering Problem (SCP) can be formally defined as follows. Let $A = (a_{ij})$ be an m-row, n-column, zero-one matrix. We say that a column j covers a row i if $a_{ij} = 1$. Each column j is associated with a nonnegative real cost c_j. Let $I = \{1, \ldots, m\}$ and $J = \{1, \ldots, n\}$ be the row set and column set, respectively. The SCP calls for a minimum cost subset $S \subseteq J$, such that each row $i \in I$ is covered by at least one column $j \in S$. A mathematical model for the SCP is

$$Minimize\, f(x) = \sum_{j=1}^{n} c_j x_j \tag{1}$$

subject to

$$\sum_{j=1}^{n} a_{ij} x_j \geq 1, \quad \forall i \in I \tag{2}$$

$$x_j \in \{0, 1\}, \quad \forall j \in J \tag{3}$$

The goal is to minimize the sum of the costs of the selected columns, where $x_j = 1$ if the column j is in the solution, 0 otherwise. The restrictions ensure that each row i is covered by at least one column.

3 Overview of Firefly Algorithm

Nature-inspired methodologies are among the most powerful algorithms for optimization problems. The Firefly Algorithm (FA) is a novel nature-inspired algorithm inspired by the social behavior of fireflies. By idealizing some of the flashing characteristics of fireflies, a firefly-inspired algorithm was presented in [20, 21]. The pseudo code of the firefly-inspired algorithm was developed using these three idealized rules:

- All fireflies are unisex and are attracted to other fireflies regardless of their sex.
- The degree of the attractiveness of a firefly is proportional to its brightness, and thus for any two flashing fireflies, the one that is less bright will move towards the brighter one. More brightness means less distance between two fireflies. However, if any two flashing fireflies have the same brightness, then they move randomly.
- Finally, the brightness of a firefly is determined by the value of the objective function. For a maximization problem, the brightness of each firefly is proportional to the value of the objective function and vice versa.

As the attractiveness of a firefly is proportional to the light intensity seen by adjacent fireflies, we can now define the variation of attractiveness β with the distance r by

$$\beta = \beta_0 e^{-\gamma r^2} \tag{4}$$

where β_0 is the attractiveness at $r = 0$. The distance r_{ij} between two fireflies is determined by

$$r_{ij} = \left\| x^i - x^j \right\| = \sqrt{\sum_{k=1}^{d} \left(x_k^i - x_k^j \right)^2} \tag{5}$$

where x_k^i is the kth component of the spatial coordinate of the ith firefly and d is the number of dimensions. The movement of a firefly i is attracted to another more attractive (brighter) firefly j is determined by

$$x_i^{t+1} = x_i^t + \beta_0 e^{-\gamma r_{ij}^2}\left(x_j^t - x_i^t\right) + \alpha\left(rand - \frac{1}{2}\right) \tag{6}$$

where x_{ij} is the firefly position of the next generation. x_i^t and x_j^t are the current position of the fireflies and x_i^{t+1} is the ith firefly position of the next generation. The second term is due to attraction. The third term introduces randomization, with α being the randomization parameter and "rand" is a random number generated uniformly but distributed between 0 and 1. The value of γ determines the variation of attractiveness, which corresponds to the variation of distance from the communicated firefly. When $\gamma = 0$, there is no variation or the fireflies have constant attractiveness. When $\gamma = 1$, it results in attractiveness being close to zero, which again is equivalent to the complete random search. In general, the value of γ [20, 21] is in between [0, 10].

4 Description of the Proposed Approach

In this section, the FA is proposed to solve the SCP using binary representation.

Step 1 Initialize the firefly parameters (γ, β_0, size for the firefly population and the maximum number of generation, for the termination process).

Step 2 Initialization of firefly position. Initialize randomly $M = [X_1; X_2; \ldots; X_m]$ of m solutions or firefly positions in the multi-dimensional search space, where m represents the size of the firefly population. Each solution of X is represented by the d-dimensional binary vector.

Step 3 Evaluation of fitness of the population. For this case the function of fitness is equal to the objective function SCP (Eq. 1).

Step 4 Modification of firefly position. A firefly produces a modification in the position based on the brightness between the fireflies. The new position is determined by modifying the value (old firefly position) using Eq. 6 for each dimension of a firefly. The result of the new component of the firefly, is probable to be a real number, to fix this, apply a threshold of 0 and 1. If x'_p is greater than the threshold, it is very likely to choose 1, otherwise 0. The threshold level can be made to range from 0 to 1, and in order to achieve this a tanh function is used as given in [24].

$$\tanh\left(\left|x'_p\right|\right) = \frac{\exp(2 * |x'_p|) - 1}{\exp(2 * |x'_p|) + 1} \tag{7}$$

Step 5 The new solution is subjected to an evaluation, if is not a feasible solution generated then is repaired. To make feasible solution is to determine which rows have not yet been covered and choose the columns needed for coverage. The search for these columns is based in: cost of a column/number

Table 1 Details of the test instances

Instance set	No. of instances	m	n	Cost range	Density (%)	Optimal solution
4	10	200	1,000	[1, 100]	2	Known
5	10	200	2,000	[1, 100]	2	Known
6	5	200	1,000	[1, 100]	5	Known
A	5	300	3,000	[1, 100]	2	Known
B	5	300	3,000	[1, 100]	5	Known
C	5	400	4,000	[1, 100]	2	Known
D	5	400	4,000	[1, 100]	5	Known
NRE	5	500	5,000	[1, 100]	10	Unknown
NRF	5	500	5,000	[1, 100]	20	Unknown
NRG	5	1,000	10,000	[1, 100]	2	Unknown
NRH	5	1,000	10,000	[1, 100]	5	Unknown

of rows not covered that cover the column *j*. Once the solution has become feasible applies optimization step to eliminate those redundant columns. A redundant column is that if removed, the solution remains feasible.

Step 6 Memorize the best solution achieved so far. Increment the generation count.

Step 7 Stop the process and display the result if the termination criteria are satisfied. Termination criteria used in this work are the specified maximum number of generations. Otherwise, go to step 3.

5 Experiments and Results

The performance of Binary Firefly Algorithm was evaluated experimentally using 65 SCP test instances from OR-Library of Beasley [25]. These instances are divided into 11 groups and each group contains 5 or 10 instances. Table 1 shows their detailed information where "Density" is the percentage of non-zero entries in the SCP matrix. The algorithm was coded in C in the development environment NetBeans 7.3 with support for C/C++ and run on a PC with a 1.8 GHz Intel Core 2 Duo T5670 CPU and 3.0 GB RAM, under Windows 8 system.

In all experiments, the Binary Firefly Algorithm is executed 50 generations, and 30 times each instance. This number was determined by the rapid convergence to a local optimal closest to global optimum. We used a population of 25 fireflies. The parameters γ, β_0 are initialized to 1. These parameters were selected empirically after a large number of tests and showed good results but may not be optimal for all instances.

Table 2 shows the results obtained of the 65 instances. Column "Optimum" reports the optimal or the best known solution value of each instance. Columns "Min. value found", "Max. value found" and "Average" reports the minimum, maximum, and average of the best solutions obtained in the 30 executions.

Table 2 Computational results on 65 instances of SCP

Instance	Optimum	Min. value found	Max. value found	Average
4.1	429	481	482	481.03
4.2	512	580	580	580.00
4.3	516	619	620	619.03
4.4	494	537	537	537.00
4.5	512	609	609	609.00
4.6	560	653	653	653.00
4.7	430	491	492	491.07
4.8	492	565	565	565.00
4.9	641	749	750	749.03
4.10	514	550	550	550.00
5.1	253	296	297	296.03
5.2	302	372	372	372.00
5.3	226	250	250	250.00
5.4	242	277	278	277.07
5.5	211	253	253	253.00
5.6	213	264	265	264.03
5.7	293	337	337	337.00
5.8	288	326	326	326.00
5.9	279	350	350	350.00
5.10	265	321	321	321.00
6.1	138	173	174	173.03
6.2	146	180	181	180.07
6.3	145	160	160	160.00
6.4	131	161	161	161.00
6.5	161	186	186	186.00
A.1	253	285	285	285.00
A.2	252	285	286	285.07
A.3	232	272	272	272.00
A.4	234	297	297	297.00
A.5	236	262	262	262.00
B.1	69	80	81	80.03
B.2	76	92	92	92.00
B.3	80	93	93	93.00
B.4	79	98	99	98.03
B.5	72	87	87	87.00
C.1	227	279	279	279.00
C.2	219	272	272	272.00
C.3	243	288	288	288.00
C.4	219	262	262	262.00
C.5	215	262	263	262.07
D.1	60	71	71	71.00
D.2	66	75	75	75.00
D.3	72	88	88	88.00
D.4	62	71	71	71.00
D.5	61	71	71	71.00
NRE.1	29	32	33	32.03

(continued)

Table 2 (continued)

Instance	Optimum	Min. value found	Max. value found	Average
NRE.2	30	36	36	36.00
NRE.3	27	35	35	35.00
NRE.4	28	34	34	34.00
NRE.5	28	34	34	34.00
NRF.1	14	17	18	17.03
NRF.2	15	17	17	17.00
NRF.3	14	21	21	21.00
NRF.4	14	19	19	19.00
NRF.5	13	16	16	16.00
NRG.1	176	230	231	230.03
NRG.2	154	191	191	191.00
NRG.3	166	198	198	198.00
NRG.4	168	214	214	214.00
NRG.5	168	223	223	223.00
NRH.1	63	85	86	85.07
NRH.2	63	81	82	81.03
NRH.3	59	76	76	76.00
NRH.4	58	75	75	75.00
NRH.5	55	68	68	68.00

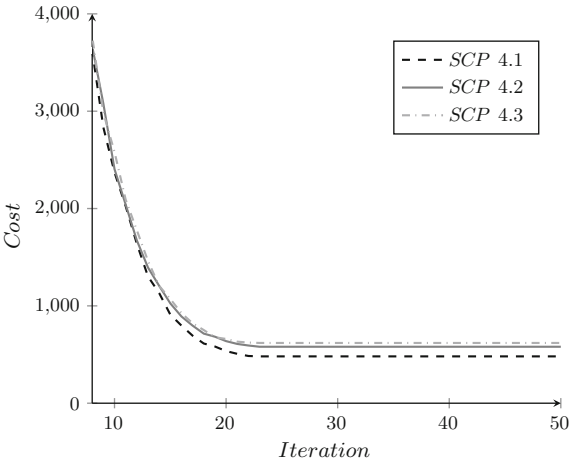


Fig. 1 Evolution of mean best values for SCP4.1, SCP4.2 and SCP4.3

In the Fig. 1 shows the evolution of mean best values for the instances 4.1, 4.2 and 4.3, which shows the rapid convergence of cost minimization.

6 Conclusions

As can be seen from the results obtained, the metaheuristic behaves of good way in almost all instances, with the first set columns instances between 1,000 and 2,000, there is a mean cost difference of 54 between the global optimum with the best optimum obtained, and starts to decrease. With a set of columns in 5,000, Firefly behaves very well coming to have a difference of 2 with respect to the best known solution value (NRF.2). This paper has demonstrated the Binary Firefly Algorithm is a valid alternative to solve the SCP, being that its main use is for continuous domains.

An interesting research direction to pursue in future work about the integration of autonomous search in the solving process, which in many cases has demonstrated excellent results [26–29].

Acknowledgements The author B. Crawford is supported by Grant CONICYT/FONDECYT/REGU- LAR/1140897. The author R. Soto is supported by Grant CONICYT/FON- DECYT/INICIACION/11130459. The author F. Paredes is supported by Grant CONICYT/FONDECYT/REGULAR/1130455.

References

1. Housos, E., Elmoth, T.: Automatic optimization of subproblems in scheduling airlines crews. Interfaces **27**(5), 68–77 (1997)
2. Vasko, F.J., Wilson, G.R.: Using a facility location algorithm to solve large set covering problems. Oper. Res. Lett. **3**(2), 85–90 (1984)
3. Vasko, F.J., Wolf, F.E.: Optimal selection of ingot sizes via set covering. Oper. Res. **35**(3), 346–353 (1987)
4. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman & Co, New York (1990)
5. Balas, E., Carrera, M.C.: A dynamic subgradient-based branch-and-bound procedure for set covering. Oper. Res. **44**(6), 875–890 (1996)
6. Fisher, M.L., Kedia, P.: Optimal solution of set covering/partitioning problems using dual heuristics. Manage. Sci. **36**(6), 674–688 (1990)
7. Chvatal, V.: A greedy heuristic for the set-covering problem. Math. Oper. Res. **4**(3), 233–235 (1979)
8. Lan, G., DePuy, G.W.: On the effectiveness of incorporating randomness and memory into a multi-start metaheuristic with application to the set covering problem. Comput. Ind. Eng. **51**(3), 362–374 (2006)
9. Ceria, S., Nobili, P., Sassano, A.: A Lagrangian-based heuristic for large-scale set covering problems. Math. Program. **81**(2), 215–228 (1998)
10. Caprara, A., Fischetti, M., Toth, P.: A heuristic method for the set covering problem. Oper. Res. **47**(5), 730–743 (1999)
11. Beasley, J.E., Chu, P.C.: A genetic algorithm for the set covering problem. Eur. J. Oper. Res. **94**(2), 392–404 (1996)
12. Brusco, M.J., Jacobs, L.W., Thompson, G.M.: A morphing procedure to supplement a simulated annealing heuristic for cost- and coverage-correlated set-covering problems. Ann. Oper. Res. **86**, 611–627 (1999)

13. Caserta, M.: Tabu search-based metaheuristic algorithm for large-scale set covering problems. In: Doerner, K., Gendreau, M., Greistorfer, P., Gutjahr, W., Hartl, R., Reimann, M. (eds.) Metaheuristics. Vol. 39 of Operations Research/Computer Science Interfaces Series, pp. 43–63. Springer, US (2007)
14. Crawford, B., Lagos, C., Castro, C., Paredes, F.: A evolutionary approach to solve set covering. In: Cardoso, J., Cordeiro, J., Filipe, J. (eds.) In: Proceedings of the 9th International Conference on Enterprise Information Systems (ICEIS '07), Vol. AIDSS, pp. 356-363. Funchal, Portugal. 12-16 June 2007
15. Ren, Z.G., Feng, Z.R., Ke, L.J., Zhang, Z.J.: New ideas for applying ant colony optimization to the set covering problem. Comput. Ind. Eng. **58**(4), 774–784 (2010)
16. Naji-Azimi, Z., Toth, P., Galli, L.: An electromagnetism metaheuristic for the unicost set covering problem. Eur. J. Oper. Res. **205**(2), 290–300 (2010)
17. Balachandar, S.R., Kannan, K.: A meta-heuristic algorithm for set covering problem based on gravity. J. Comput. Math. Sci. **4**, 223–228 (2010)
18. Crawford, B., Soto, R., Monfroy, E.: Cultural algorithms for the set covering problem. In: Tan, Y., Shi, Y., Mo, H. (eds.) ICSI (2). Vol. 7929 of Lecture Notes in Computer Science, pp. 27–34. Springer, Berlin (2013)
19. Caprara, A., Toth, P., Fischetti, M.: Algorithms for the set covering problem. Ann. Oper. Res. **98**, 353–371 (2000)
20. Yang, X.S.: Nature-Inspired Metaheuristic Algorithms. Luniver Press, UK (2008)
21. Yang, X.S.: Firefly algorithms for multimodal optimisation, In: Proceedings of the 5th International Conference on Stochastic Algorithms: Foundations and Applications. SAGA'09, pp. 169–178. Springer, Berlin (2009)
22. Fister, I., Fister Jr, I., Yang, X.S., Brest, J.: A comprehensive review of firefly algorithms. Swarm Evol. Comput. **13**, 34–46 (2013)
23. Yang, X.S., He, X.: Firefly Algorithm: Recent Advances and Applications. The Computing Research Repository, abs/1308.3898 (2013)
24. Chandrasekaran, K., Sishaj, P.S., Padhy, N.P.: Binary real coded firefly algorithm for solving unit commitment problem. Inf. Sci. **249**, 67–84 (2013)
25. Beasley, J.E.: A Lagrangian heuristic for set covering problems. Naval Res. Logistics **37**(1), 151–164 (1990)
26. Crawford, B., Soto, R., Monfroy, E., Palma, W., Castro, C., Paredes, P.: Parameter tuning of a choice-function based hyperheuristic using particle swarm optimization. Expert Syst. Appl. **40**(5), 1690–1695 (2013)
27. Soto, R., Crawford, B., Monfroy, E., Bustos, V.: Using autonomous search for generating good enumeration strategy blends in constraint programming. In: Proceedings of the 12th International Conference on Computational Science and Its Applications (ICCSA). Vol. 7335 of LNCS, pp. 607–617. Springer (2012)
28. Crawford, B., Soto R., Montecinos M., Castro C., Monfroy, E.: A framework for autonomous search in the eclipse solver. In: Proceedings of the 24th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE). Vol. 6703 of LNCS, pp. 79–84. Springer (2011)
29. Crawford, B., Soto R., Castro C., Monfroy, E.: Extensible CP-based autonomous search. In: Proceedings of HCI International. Vol. 173 of CCIS, pp. 561–565. Springer (2011)

Neural Networks in Modeling of CNC Milling of Moderate Slope Surfaces

Ondrej Bilek and David Samek

Abstract Computer numerical control (CNC) allows achieving a high degree of automation of machine tools by pre-programmed numerical commands. CNC milling process is widely used in industry for machining of complex parts. The need of a description of the CNC milling process is necessary for production of precise parts. This paper introduces artificial neural network based modeling, while the CNC milling of moderate slope shapes is studied. The developed neural models consist of two inputs and two outputs. The created neural models were experimentally tested on the real data. Then, the evaluation and comparison of all models were performed.

Keywords Artificial neural networks · CNC milling · Modeling · Surface quality

1 Introductions

CNC machining is one of the most widely used methods of conventional machining with defined tool geometry. Single CNC machine tool receives commands from a single computer. Computer control of a machining provides significant advantages [1–7] in comparison with human unpredictable machining. Moreover CNC machining leads to unique opportunities for process planning. Knowledge of the machining process and the optimum settings of the input parameters are essential for the quality and precision of the machined parts. Despite the fact that the machining process is influenced by a vast number of

O. Bilek (✉) · D. Samek
Faculty of Technology, Tomas Bata University in Zlin, Zlin, Czech Republic
e-mail: bilek@ft.utb.cz

D. Samek
e-mail: samek@ft.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_7,

factors relating to tool—machine tool—workpiece—clamping framework [8–12], current models of machining process are able to predict the required inputs moreover to control cutting conditions according to outputs [13].

Milling is typically used for machining of shaped functional surfaces. Milling is the main machining technology that has no comparable alternative for the field of conventional machining. Milling is successfully applied to a variety of operations, from roughing to finishing. Milling process is characterized by the use of different cutting strategy, and by the tools regularly with many cutting edges, while in most cases the creation of the part program is completed using Computer Aided Manufacturing (CAM) software [14]. The surface quality after milling process is a decisive indicator for the evaluation of manufactured parts. Surface quality affects a number of factors such as friction, corrosion resistance and distribution of lubricants, heat, light reflection and fatigue strength [15–23].

An important parameter characterizing the quality of surface is *Ra* parameter that is calculated as an arithmetic average roughness of the measured profile. This parameter is widely used and is commonly known in industry, while providing only limited information on the machined surface [24]. The parameter *Ra* is mostly accompanied by *Rz* parameter evaluating the maximum height of the profile. Parameter *Rz* is dependent on *Ra* parameter and carries important information on the quality of the surface, and therefore, together with *Ra* are considered to be key factors in the following experiments.

Methodology of surface roughness prediction includes various approaches, such as kinematic model, experimental investigation and analysis, implementation of artificial intelligence (AI) and approaches that use designed experiments. The scope of recent studies are established on knowledge of ball end milling process that is necessary for finishing oblique and free-form surfaces [25–34].

Still, only some researchers pay an attention to this problem and thus few models of surface roughness are suitable for manufacturing practices. The prediction of surface roughness in advance is a key requirement for industry to improve manufacturing process while reducing the cost of production [35, 36].

The paper presents modeling and prediction of technological parameters of CNC milling using artificial neural networks (ANN), while multilayered feed-forward neural network (MFFNN) was applied. The studied input parameters of the milling process are as follows: radial depth of cut a_e, feed per tooth f_z. The obtained results are verified on the experimental measurement.

2 Initial Experiments

The milling operations were performed on the three axis vertical milling center Mikron HSM 800. For the purpose of experiment were selected ball mill tools from sintered carbide with PVD coating. Cutting speed was constant 200 m/min, tool rake angle was negative −4°, and surface inclination angle was 15°. For the tool clamping was used the shrink fit holder HSK 50E on a modular system

Easyshrink 20. The tool overhang was 50 mm during milling. The investigated input parameters radial depth of cut a_e, feed per tooth f_z were set from the range 0.16–0.60 mm and 0.1–0.17 mm respectively. The complete list of process parameters is shown in Table 1. The process parameters settings selected for this experimental work were chosen optimal according to manufacturing practices.

Machined workpieces were from stainless steel X153CrMoV12-1 with hardness of 63 Rockwell Hardness (HRC), whereas the chemical composition was guaranteed by the supplier. Thirty specimens were machined for each input parameters combination.

NX CAM software was used to create CNC part program for the finishing operation within the ±0.01 tolerance, using the Z Level Profile strategy as can be seen in Fig. 1. The feed direction was parallel to the longest workpiece edge and climb cutting strategy was considered for the better surface roughness. Nevertheless, the machining of open areas such as standalone surfaces in described experiment significantly increase machining time due to a larger number of non-cutting movements.

Surfaces of the all thirty workpieces were measured using Mitutoyo SJ-301 after machining. According to standard, surface roughness was measured in the perpendicular direction to the feed rate, providing higher values of *Ra* and *Rz*. The parameter *Ra* was within the experiment considered as surface characteristic that is in addition typical parameter of surface roughness in manufacturing practices [37, 38].

3 Modeling of CNC Milling

The model design results from the problem definition—two observed input parameters (a_e and f_z) and two desired output parameters (*Ra* and *Rz*). Thus, the model has to contain two inputs and two outputs. The experimentally obtained data were loaded into Matlab, where all computations were performed. After the statistical analysis it was decided to use artificial neural network as the process model, because the data were noised, multidimensional and strongly nonlinear.

Using Matlab Neural Network Toolbox various structures of MFFNN were created in order to find optimal solution for the given problem. There were finally tested the seven artificial neural network structures that are denoted as net1–net7. Simplified structure of neural model of net7 is shown for illustration in Fig. 2.

The tested structures are listed in the Table 2, while NN stands for number of neurons, TF is abbreviation of transfer function, T represents hyperbolic tangent function and L stands for linear function.

The experimental data were transformed into the interval $\langle -1, -1 \rangle$. After that, the created artificial neural networks were trained to the transformed measured data using Levenberg-Marquart Algorithm. Certainly, after the prediction all output data had to be transformed back before validation to real experimental data.

For the all computations the batch programs were created using Matlab standard programming environment (M-Files).

Table 1 Process parameters of the experiment

Cutting speed (v)	200 m/min
Radial depth of cut (a_e)	0.16, 0.25, 0.32, 0.40, 0.60 mm
Feed per tooth (f_z)	0.1, 0.12, 0.135, 0.15, 0.17 mm
Surface slope /inclination angle (α)	Moderate slope /15°
Tool rake angle (γ)	12°
Tool diameter	12 mm
Tool overhang	50 mm

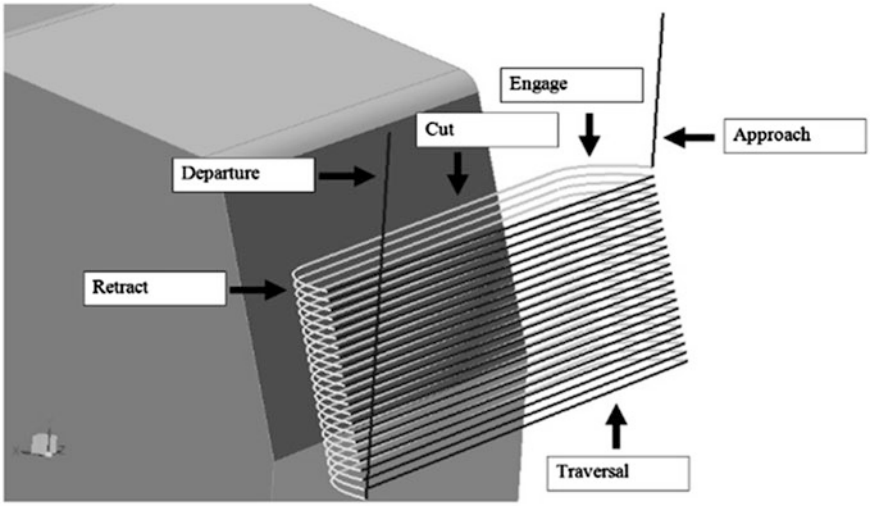


Fig. 1 Generated tool path by NX 8.5

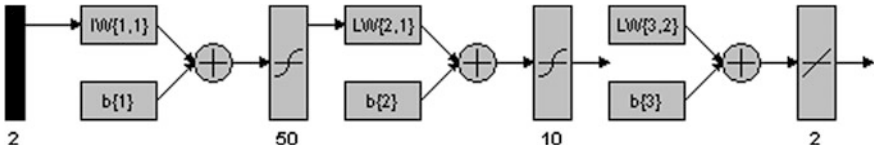


Fig. 2 Simplified structure of the model net7

4 Verification and Comparison of the Models

The prepared models were tested by new experimental data. Then, the neural models were used for prediction of resulting surface quality. After that, the fifteen of workpieces were milled and measured for each combination of inspected parameters. The machining and other conditions remained the same as in the part II of this paper. The obtained experimental results were compared to the predicted values.

Table 2 Tested neural network structures

	Input layer		Hidden layer 1		Hidden layer 2		Output layer	
	NN	TF	NN	TF	NN	TF	NN	TF
net1	2	–	5	T	–	net1	2	–
net2	2	–	10	T	–	net2	2	–
net3	2	–	20	T	–	net3	2	–
net4	2	–	50	T	–	net4	2	–
net5	2	–	100	T	–	net5	2	–
net6	2	–	20	T	10	net6	2	–
net7	2	–	50	T	10	net7	2	–

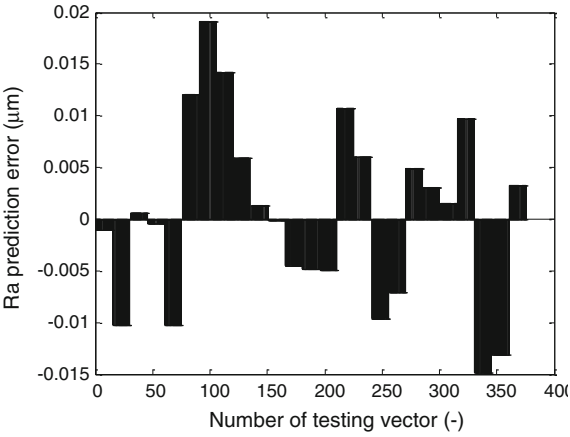


Fig. 3 Prediction error for *Ra* in μm using net7

The differences between measured data and output of the model were computed in micrometers and percent for the all seven tested artificial neural structures. Results of net7 are depicted in Figs. 3, 4, 5 and 6.

In order to numerically compare prediction accuracy of all predictors following criteria were defined. Average absolute value of prediction difference for *Ra* J_{Ra} and for *Rz* J_{Rz}:

$$J_{Ra} = \frac{\sum_{i=1}^{n} |t(i) - y(i)|}{n} \tag{1}$$

$$J_{Rz} = \frac{\sum_{i=1}^{n} |t(i) - y(i)|}{n} \tag{2}$$

where n is number of measurements, t stands for target (measured value) and y is predicted value (output of neural models).

Because it is very important to observe also extremes in prediction inaccuracy, the maximal prediction error for *Ra* E_{Ra} and E_{Rz} for *Rz* are used.

The resulting criteria are presented in the Table 3.

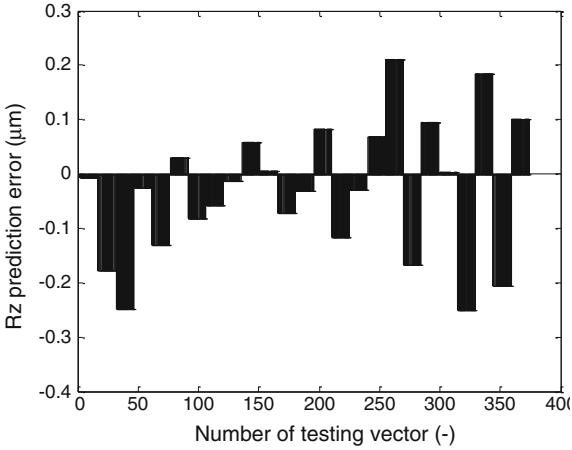


Fig. 4 Prediction error for *Rz* in μm using net7

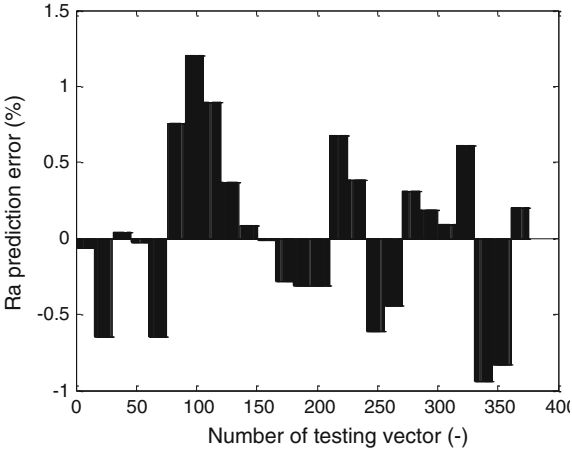


Fig. 5 Prediction error for *Ra* in % using net7

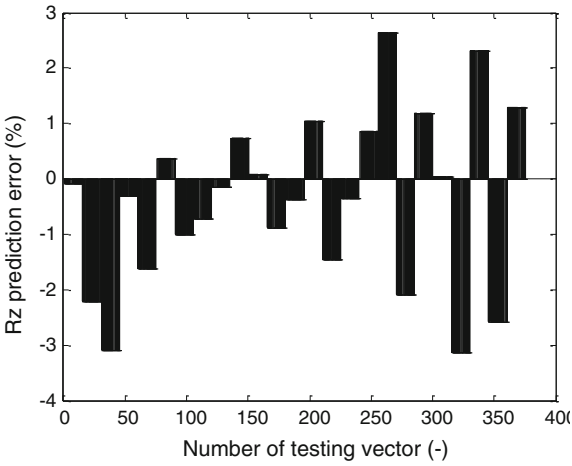


Fig. 6 Prediction error for *Rz* in % using net7

Table 3 Comparison of predictors

	J_{Ra} (μm)	J_{Rz} (μm)	E_{Ra} (μm)	E_{Rz} (μm)
net1	0.0145	0.1112	0.0431	0.2061
net2	0.0071	0.0946	0.0207	0.2951
net3	0.0070	0.1165	0.0194	0.2044
net4	0.0070	0.1171	0.0183	0.2963
net5	0.0073	0.1143	0.0251	0.2343
net6	0.0066	0.1238	0.0174	0.3815
net7	0.0069	0.0986	0.0192	0.2522

Table 4 Multiple attribute decision making

	$J_{Ra}(-)$	$J_{Rz}(-)$	$E_{Ra}(-)$	$E_{Rz}(-)$	Σ
net1	7	3	7	2	19
net2	5	1	5	5	16
net3	3	5	4	1	13
net4	3	6	2	6	17
net5	6	4	6	3	19
net6	1	7	1	7	16
net7	2	2	3	4	11

As can be seen, it is complicated to distinguish what model provides the best results, because there are four winners—from the point of view of each of the criteria the different network is the best. Therefore, it was used multiple attribute decision making method, while it was applied same weight for all four criteria. The ordinal method was utilized for assigning points to the individual networks/criteria. Therefore, the lowest sum of the points results to the winner. As can be seen from Table 4, the best score gives the net7.

5 Discussion

The experimental results show that the artificial neural network based model net7 provides reasonable good results. The maximum prediction error for *Ra* and *Rz* was 0.0192 and 0.2522 μm, respectively. The prediction error in percent was 1.2 % for *Ra* and 3.1 % for *Rz* at the most. The average value of the prediction error was 0.0069 μm for *Ra* and 0.0986 μm for *Rz*.

On the other hand, the other predictors are worth of noticing too. For example net6 is excellent for the prediction of *Ra*—it has the best values of the criteria J_{Ra} and E_{Ra}, but at the same time it has the worst prediction accuracy for *Rz*. The lowest average prediction difference J_{Rz} imparted net2 with the one hidden layer; on the contrary the net3 had the lowest E_{Rz}.

It can be concluded that design of models/predictors of technological processes is complex and at all cases the model has to be verified by new experiments in order to get the proof that predictor is appropriate. The proposed predictor based on multilayered feed-forward neural network can be supposed for obtaining optimal settings of the CNC milling machine for desired surface quality. What is more, the predictor enables the prediction outside the measured data.

References

1. Bouzakis, K.-D., Aichouh, P., Efstathiou, K.: Determination of the chip geometry, cutting force and roughness in free form surfaces finishing milling, with ball end tools. Int. J. Mach. Tools Manuf. **43**, 499–514 (2003)
2. Lukovics, I.: High speed milling of metal and polymer materials. Manuf. Technol. **4**, 29–33 (2004)
3. Kasina, M., Vasilko, K.: Experimental verification of the relation between the surface roughness and the type of used tool coating. Manuf. Technol. **12**, 27–30 (2012)
4. Miko, B., Beni B.: Study of surface roughness in case of Z-level finishing. In: International GTE Conference Manufacturing 2012, Budapest (2012)
5. Izol, P., Beno, J., Balazs, M.: Precision and surface roughness when free–form–surface milling. Manuf. Ind. Eng. **1**, 70–73 (2011)
6. Sebelova, E., Chladil, J.: Tool wear and machinability of wood-based materials during machining process. Manuf. Technol. **13**, 231–236 (2013)
7. Cerny, J., Ovsik, M., Bednarik, M., Mizera, A., Manas, D., Manas, M., Stanek, M.: Mod-ern methods of design of ergonomics parts. In: Recent Research Circuits Systems, vol. 7, pp. 321–324. Kos (2012)
8. Wu, C.-T.: Establishing a correlative model for improving NC machining process. Int. J. Mech. **5**, 100–112 (2011)
9. Ghionea, I., Ghionea, A.: Simulation Techniques in CAD-CAM Processing by Milling of Surfaces on NC Machine-Tools. In: Advances in Production, Automation and Transportation Systems, vol. 1, pp. 135-140. Brasov (2013)
10. Dragoi, M.V.: Ball nose milling cutter radius compensation in Z axis for CNC. In: Proceedings of 8th WSEAS International Conference on Software Engineering Parallel and Distributed Systems, pp. 57–60. Cambridge (2009)
11. Zebala, W.: Milling optimization of difficult to machine alloys. Management **1**, 59–70 (2010)
12. Zebala, W., Matras, A., Beno, J.: Optimization of free-form surface milling. Manuf. Eng. **3**, 17–20 (2011)
13. Benardos, P.G., Vosniakos, G.C.: Predicting surface roughness in machining: a review. Int. J. Mach. Tool. Man. **43**, 833–844 (2003)
14. Cubonova, N.: Postprocessing of CL data in CAD/CAM system Edgecam using the constructor of postprocessors. Manuf. Technol. **13**, 158–163 (2013)
15. Felho, C.: A method for calculation of theoretical roughness in face milling. In: Proceedings of 8th International Tools Conference, pp. 84–87. Zlin (2011)
16. Micietova, A., Neslusan, M., Cillikova, M.: Influence of surface geometry and structure after non-conventional methods of parting on the following milling operations. Manuf. Technol. **13**, 199–204 (2013)
17. Quinsat, Y., Sabourin, L., Lartigue, C.: Surface topography in ball end milling process: description of a 3D surface roughness parameter. J. Mater. Process. Technol. **195**, 135–143 (2008)
18. Ho, W.-H., Tsai, J.-T., Lin, B.-T., Chou, J.-H.: Adaptive network-based fuzzy inference system for prediction of surface roughness in end milling process using hybrid Taguchi-genetic learning algorithm. Expert Syst. Appl. **36**, 3216–3222 (2009)

19. Buj-Corral, I., Vivancos-Calvet, J., Dominguez-Fernandez, A.: Surface topography in ball-end milling processes as a function of feed per tooth and radial depth of cut. Int. J. Mach. Tools Manuf. **53**, 151–159 (2012)
20. Dhokia, V.G., Kumar, S., Vichare, P., Newman, S.T.: An intelligent approach for the prediction of surface roughness in ball-end machining of polypropylene. Rob. Comput. Integr. Manuf. **24**, 835–842 (2008)
21. Felho, C., Kundrak, J.: Characterization of topography of cut surface based on theoretical roughness indexes. Key Eng. Mater. **496**, 194–199 (2011)
22. Antoniadis, A., Savakis, C., Bilalis, N., Balouktsis, A.: Prediction of surface topomorphy and roughness in ball-end milling. Int. J. Adv. Manuf. Technol. **21**, 965–971 (2003)
23. Jung, T.-S., Yang, M.-Y., Lee, K.-J.: A new approach to analysing machined surfaces by ball-end milling, part I. Int. J. Adv. Manuf. Technol. **25**, 833–840 (2005)
24. Quintana, G., Ciurana, J., Ribatallada, J.: Surface roughness generation and material removal rate in ball end milling operations. Mater. Manuf. Process. **25**, 386–398 (2010)
25. Tandon, V., El-Mounayri, H., Kishawy, H.: NC end milling optimization using evolutionary computation. Int. J. Mach. Tools Manuf. **42**, 595–605 (2002)
26. Iliescu, M., Spanu, P., Rosu, M., Comanescu, B.: Simulation of cylindrical-face milling and modeling of resulting surface roughness when machining polymeric composites. In: Proceedings of 11th WSEAS International Conference on Automatic Control, Modelling and Simulation, pp. 219–224. Istanbul (2009)
27. Folea, M., Schlegel, D., Lupulescu, N., Parv, L.: Modeling surface roughness in high speed milling: cobalt based superalloy case study. In: Proceedings of 1st International Conference on Manufacturing Engineering Quality Production System, pp. 353–357. Brasov (2009)
28. Babur, O., Oktem, H., Kurtaran, H.: Optimum surface roughness in end milling Inconel 718 by coupling neural network model and genetic algorithm. Int. J. Adv. Manuf. Technol. **27**, 234–241 (2005)
29. Al-Zubaidi, S., Ghani, J.A., Haron, C.H.C.: Application of artificial neural networks in prediction tool life of PVD coated carbide when end milling of TI6aL4v alloy. Int. J. Mech. **6**, 179–186 (2012)
30. Mankova, I., Vrabel, M., Kovac, P.: Artificial neural network application for surface roughness prediction when drilling nickel based alloy. Manuf. Technol. **13**, 193–199 (2013)
31. Quintana, G., Garcia-Romeu, M.L., Ciurana, J.: Surface roughness monitoring application based on artificial neural networks for ball-end milling operations. J. Intell. Manuf. **22**, 607–617 (2011)
32. Yegnanarayana, B.: Artificial neural networks. PHI Learning Pvt. Ltd., New Delhi (2004)
33. Karayel, D.: Prediction and control of surface roughness in CNC lathe using artificial neural network. J. Mater. Process. Technol. **209**, 3125–3137 (2009)
34. Correa, M., Bielza, C., Pamies-Teixeira, J.: Comparison of Bayesian networks and artificial neural networks for quality detection in a machining process. Expert Syst. Appl. **36**, 7270–7279 (2009)
35. Oktem, H., Erzurumlu, T., Erzincanli, F.: Prediction of minimum surface roughness in end milling mold parts using neural network and genetic algorithm. Mater. Des. **27**, 735–744 (2006)
36. Venkatesan, D., Kannan, K., Saravanan, R.: A genetic algorithm-based artificial neural network model for the optimization of machining processes. Neural Comput. Appl. **18**, 135–140 (2009)
37. Zeroudi, N., Fontaine, M.: Prediction of machined surface geometry based on analytical modelling of ball-end milling. Procedia CIRP **1**, 108–113 (2012)
38. El-Mounayri, H., Kishawy, H., Briceno, J.: Optimization of CNC ball end milling: a neural network-based model. J. Mater. Process. Technol. **166**, 50–62 (2005)

Application of Linguistic Fuzzy-Logic Control in Technological Processes

Radim Farana

Abstract This paper presents the use of modern numerical methods such as Fuzzy Logic Control for control of fast technological processes with sampling period 0.01 [s] or less. The paper presents a real application of the Linguistic Fuzzy-Logic Control, developed at the University of Ostrava for the control of magnetic levitation model in the laboratory at the Institute for Research and Applications of Fuzzy Modeling and Department of Informatics and Computers, Faculty of Science. This technology and real models are also used as a background for problem-oriented teaching realized at the department for master students and their collaborative as well as individual final projects. The paper shows how the used technology can help people easily describe the control strategy from the technological control strategy point of view.

Keywords Fuzzy logic · Control · LFLC · Magnetic levitation

1 Introduction

Fuzzy logic has been invented by Prof. Zadeh [1] and used to describe uncertain systems [2] since the 60s of the 20th century. This technique has also been used in control systems. Fuzzy control is now the standard control method which is a constituent of many industrial systems and companies advertise it no more. The used technique is mostly based on application of fuzzy IF-THEN rules; either in the form first used by Mamdani [3], or by Takagi and Sugeno [4]. The success of fuzzy logic control is based on the fact that a description of real systems is quite often imprecise. The imprecision arises from several factors—too large complexity of

R. Farana (✉)
University of Ostrava, 30 dubna 22, 701 00 Ostrava 1, Czech Republic
e-mail: radim.farana@osu.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_8,

the controlled system, insufficient precise information, presence of human factor, necessity to save time or money, etc. Frequently, a combination of several of such factors is present.

A special system for fuzzy logic control has been developed at the University of Ostrava by Prof. Novák and his team [5–7] based on linguistic description. The Linguistic Fuzzy Logic Controller (LFLC) is the result of application of the formal theory of the fuzzy logic in broader sense (FLb). The fundamental concepts of FLb are evaluative linguistic expressions and linguistic description. Evaluative (linguistic) expressions are natural language expressions such as small, medium, big, about twenty-five, roughly one hundred, very short, more or less deep, not very tall, roughly warm or medium hot, roughly strong, roughly medium important, and many others. They form a small, but very important, constituent of natural language since we use them in common sense speech to be able to evaluate phenomena around. Evaluative expressions have an important role in our life because they help us determine our decisions, help us in learning and understanding, and in many other activities.

Simple evaluative linguistic expressions (possibly with signs) have a general form < linguistic modifier > <TE-adjective > (where < TE-adjective > is one of the adjectives (also called gradable) "small—sm, medium—me, big—bi" or "zero —ze". The < linguistic modifier > is an intensifying adverb such as "extremely—ex, significantly—si, very—ve, rather—ra, more or less—ml, roughly—ro, quite roughly —qr, very roughly—vr"), see Fig. 1. LFLC is a good tool to define the control strategy, then we also use it to control technological processes with sampling period 0.01 [s] or less. This paper presents results obtained when solving problems with control of a magnetic levitation model, representing a very fast control system. This model is very helpful for application, because its description and mathematical model is available, for example [8], see Fig. 2.

2 Magnetic Levitation

Magnetic levitation is a very complex nonlinear problem. We are unable to use classical identification methods to obtain a mathematical model. Fortunately, we have a very good mathematical model developed by the model producer [8], see Fig. 3.

We also have the PID controller set up to control the magnetic levitation object position, which could be used as a reference for our fuzzy controller, see Fig. 4.

A control result for a desired value generated as a pulse signal is shown in Fig. 5. We see that the magnetic levitation object is very sensitive and the control process is unstable. The control system stabilized the desired position closer to the electromagnet only once.

Analyzing the PID control, we can see that the first derivative value is hundred times higher than the control error value and the second derivative value is hundred times higher than first derivative value. This is caused by the sampling period

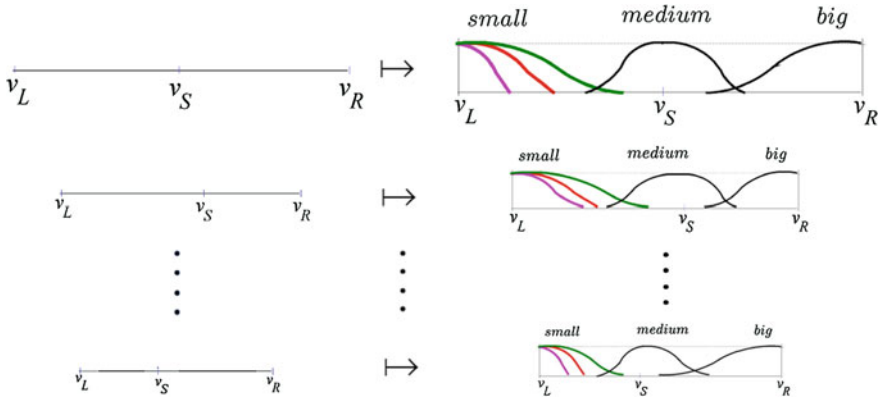


Fig. 1 A general scheme of intension of evaluative expressions (extremely small, very small, small, medium, big) as a function assigning a specific fuzzy set [7] to each context $w \in W$

Fig. 2 Magnetic levitation model

$T = 0.002$ [s]. Then we cannot develop a classical fuzzy controller based on three input values—control error and its first and second derivatives. For these cases we develop a special strategy based on multiple use of LFLC controllers, see Fig. 6.

Every partial LFLC controller will react to one input value. Outputs from all partial controllers will be summarized in a discrete integrator. It is also easy to change the control strategy, for example a very small reaction to a small error and a very big reaction to a big error to obtain the needed value faster, see Fig. 7.

Table 1 presents the LFLC controller contexts set up for a partial controller based on the controlled object behavior. The LFLC control result is shown in

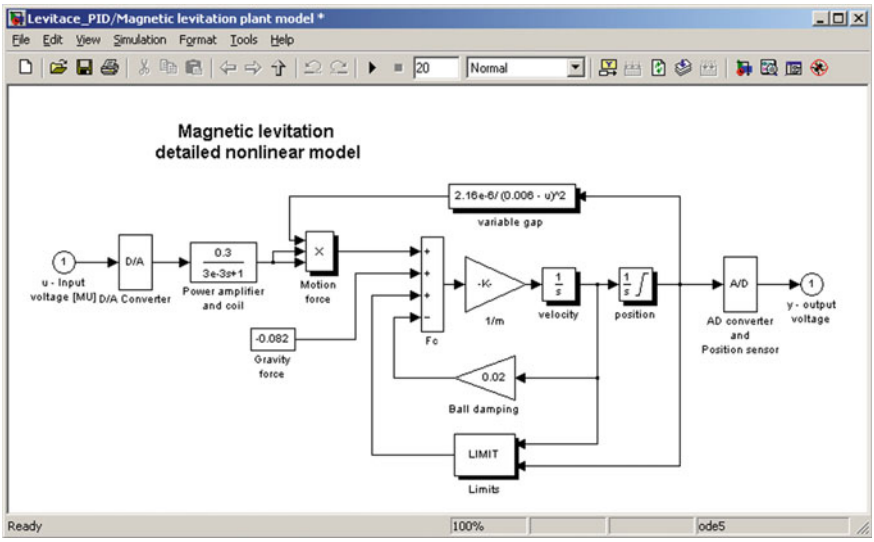


Fig. 3 Magnetic levitation simulation model

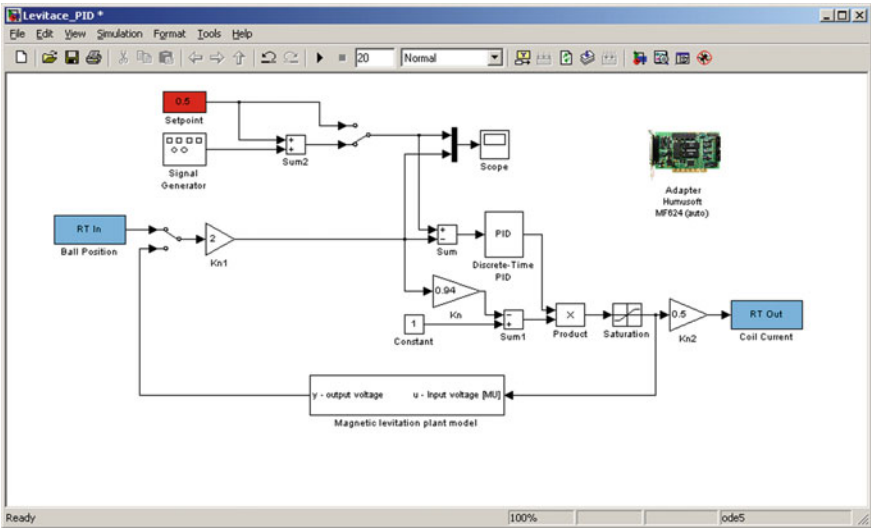


Fig. 4 PID control developed by the magnetic levitation model producer

Fig. 8. We see that the control process is much better than the PID control results (compare with Fig. 5). The controlled process is still very sensitive, the control accuracy is not ideal, but the problem with the stability has been mostly eliminated thanks to the LFLC control properties.

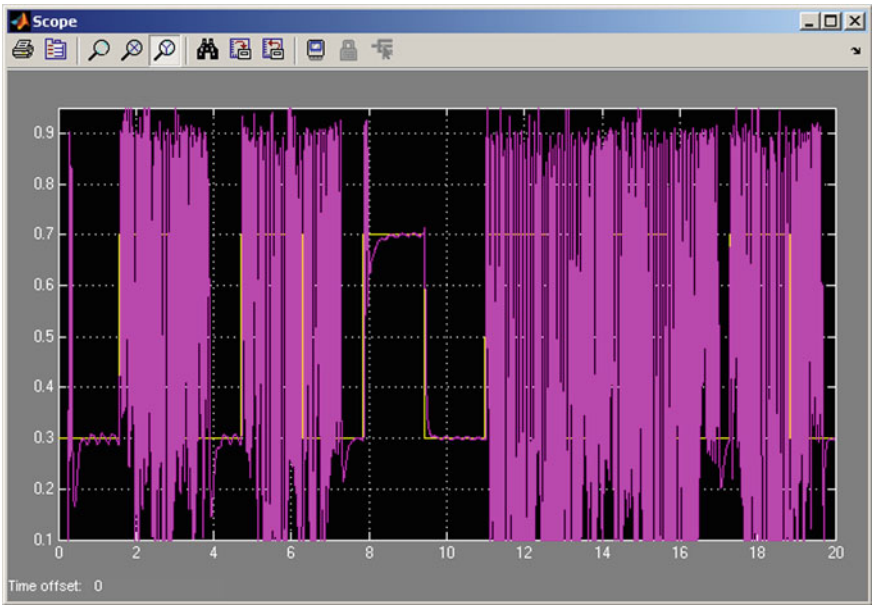


Fig. 5 PID control result

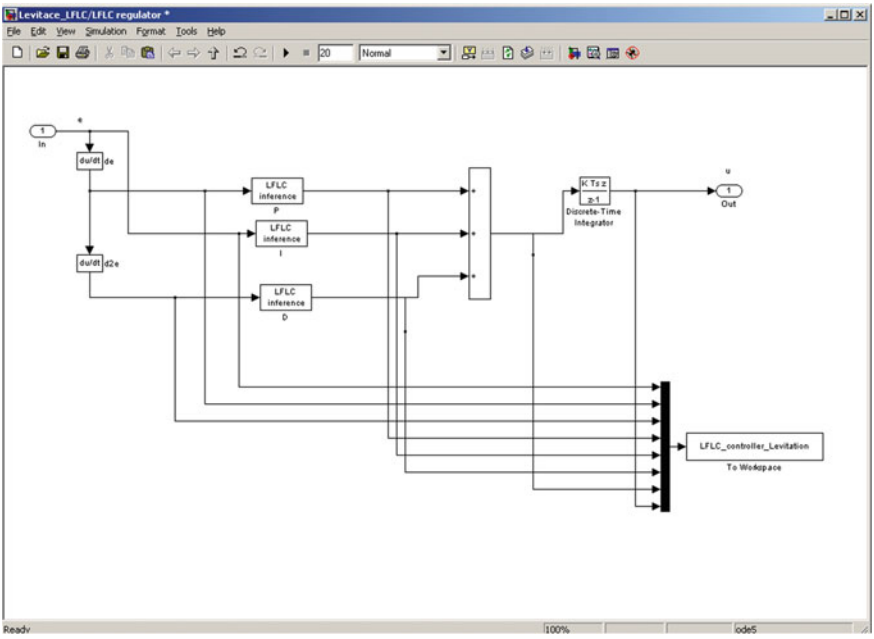


Fig. 6 LFLC controller developed for the magnetic levitation model

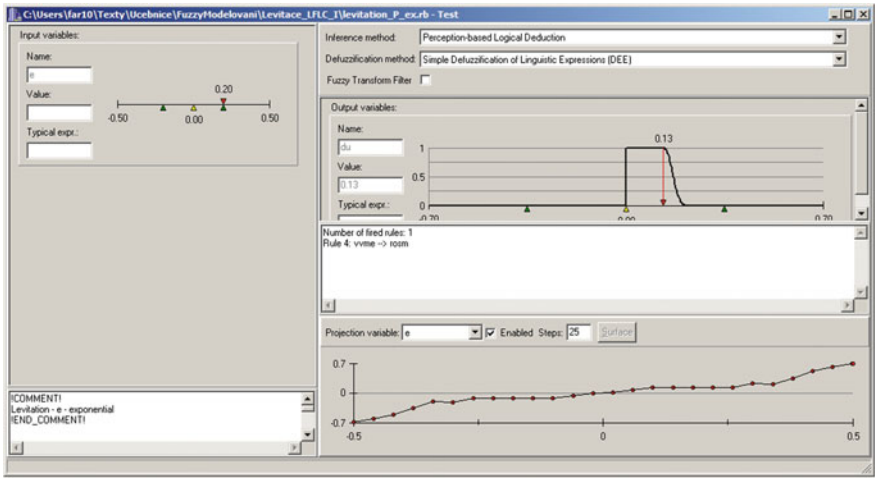


Fig. 7 LFLC controller behavior

Table 1 LFLC controller contexts

Input value	Scale	Transfer coefficient	Output scale
de	$-50 \div 50$	1	$-50 \div 50$
e	$-0.5 \div 0.5$	10	$-5 \div 5$
d^2e	$-40{,}000 \div 40{,}000$	0.03	$-1{,}200 \div 1{,}200$

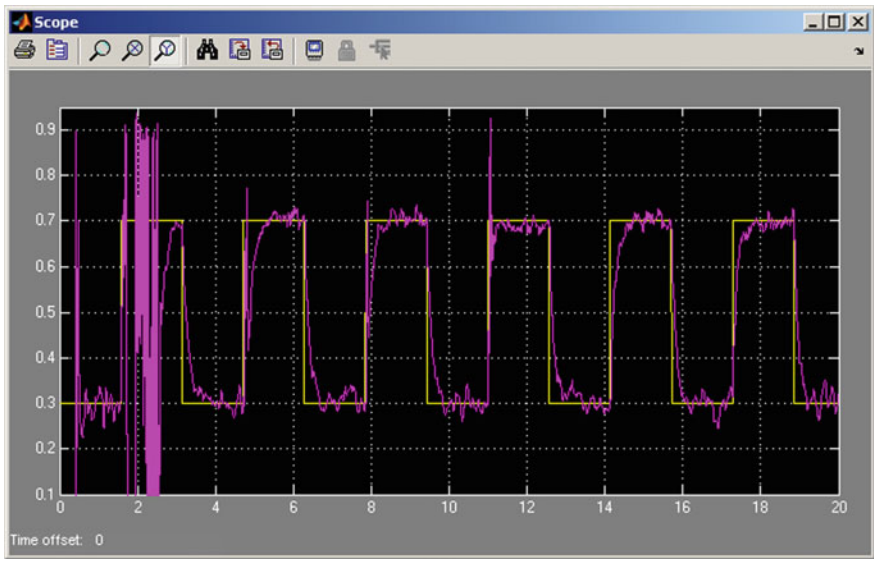


Fig. 8 LFLC control result

3 Conclusions

The presented example of LFLC use has been solved at the University of Ostrava. It is obvious that modern numerical methods such as Fuzzy Logic Control are usable for control of fast technological processes with sampling period 0.01 [s] or less. The Linguistic Fuzzy-Logic Control, developed at the University of Ostrava, is a very helpful tool for control strategy description. The presented results proved how the used technology can help people easily describe control strategy from the technological control strategy point of view. This technology and real models are used as a background for problem-oriented teaching realized at the Department of Informatics and Computers, Faculty of Science, for master students and their collaborative as well as individual final projects. Students learned how to define the control strategy and verify it on a real magnetic levitation model. Having completed these projects, students are able to define control strategies based on LFLC for any similar controlled system [9, 10].

Acknowledgements This work was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and during the completion of a Student Grant (SGS15/PřF/2014) with student participation, supported by the Czech Ministry of Education, Youth and Sports.

References

1. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
2. Zadeh, L.A., Kacprzyk, J.: Fuzzy logic for the management of uncertainty. Wiley, New York (1992)
3. Mamdani, E., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic control. Int. J. Man-Mach. Stud. **7**, 1–13 (1975)
4. Takagi, T., Sugeno, M.: Fuzzy identification of systems and its applications to modeling and control. IEEE Trans. Syst. Man Cybern. **15**, 116–132 (1985)
5. Novák, V.: Linguistically oriented fuzzy logic control and its design. Int. J. Approximate Reasoning **12**, 263–277 (1995)
6. Novák, V. Perfilieva, I.: Evaluating linguistic expressions and functional fuzzy theories in fuzzy logic. In: Zadeha, LA., Kacpryk, J. (eds.). Computing with Words in Information/ Intelligent Systems, vol. 1, pp. 383–406. Springer, Heidelberg (1999)
7. Novák, V.: Genuine linguistic fuzzy logic control: powerful and successful control method. In: Hüllermeier, E. and Kruse, R. and Hoffmann, F. (eds.). Computational Intelligence For Knowledge-Based Systems Design, pp. 634–644. Springer, Berlin (2010)
8. HUMUSOFT Web information system 2012 (on-line). Cite 10 Jan 2013. Available on web. http://www.humusoft.cz/produkty/models/ce152/ (2012)
9. Takosoglu, J.E., Laski, P.A., Blasiak, S.: A fuzzy logic controller for the positioning control of an electro-pneumatic servo-drive. J. Syst. Control. Eng. **226**(10), 1335–1343 (2012)
10. Godoy, W.F., Da Silva, I.N., Goedtel, A., Palácios, R.H.C.: Fuzzy logic applied at industrial roasters in the temperature control. In: 11th IFAC Workshop on Intelligent Manufacturing Systems, IMS 2013, Sao Paulo, Brazil, pp. 450–455 (2013)

Hybrid Intelligent System for Point Localization

Robert Jarusek, Eva Volna, Alexej Kolcun and Martin Kotyrba

Abstract The article introduces a hybrid intelligent system for point localization in 3D Euclidean space. There are two models presented. The first one is based on neural networks and the second one represents a classical approach. The classical model calculates Euclidean distances between two points in the defined domain. As regards the experimental study, we proposed appropriate topologies of the systems that depend on the required accuracy. At first, we identified distances between a randomly generated point and a reference points in the defined domain. Then a neural network uses the obtained distances as its inputs to determine the actual position of the point in the domain space. The experimental study was repeated several times. All obtained results are mutually compared in the conclusion.

Keywords Hybrid intelligent system · Point localization · Neural networks · Euclidean distance

R. Jarusek · E. Volna (✉) · A. Kolcun · M. Kotyrba
Department of Informatics and Computers, University of Ostrava,
30 dubna 22, 70103 Ostrava, Czech Republic
e-mail: eva.volna@osu.cz

R. Jarusek
e-mail: robert.jarusek@osu.cz

A. Kolcun
e-mail: alexej.kolcun@osu.cz

M. Kotyrba
e-mail: martin.kotyrba@osu.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_9,

1 Introduction

Motivation of the article emerges from the proposed acoustic motion capture system based on neural networks described in [1–3] and following works. At first, the distance between an active transmitter and a receiver was identified on the basis of sound pulses transmitted in the defined domain. It means that we gradually emitted an acoustic pulse from different transmitters into the microphone. The domain space was defined with regard to transmitters' placement. We are sure that one sound pulse leaves the room earlier prior to emitting a pulse by next transmitter. Thus, there is one pulse at a time in the area only. After noise removal, the onset of the sound pulse is found in the sample as a maximum of the signal. Here, the neural network approach was used. The distance from the transmitter to the receiver we obtained from the time difference of the emitted and received sound pulse. A two-dimensional case of such system was presented in [1–3] and the distance is also found using the neural network approach.

In this paper we aim to compare the standard approach and neural network approach for determining the position of the receiver when the system transmitters–receiver is three-dimensional. We particularly analyze cases of exact and vague values of distances transmitter–receiver.

2 Theoretical Background

2.1 Point Localization Systems

There are many methods for feature point localization, but some of these methods rely on hand or special hardware such as infrared illumination, electrodes to place on face, high resolution camera, etc.

Estimation of three-dimensional information in active systems is a crucial problem in computer vision because camera parameters may change dynamically depending on the scene. The problem of localizing objects in 3D while given multiple images from cameras in different locations is widely known as 'Stereo Vision' problem. Prevalent approaches to this issue are based on projective geometry and photogrammetry. Cameras taking photographs of the same scene from two different locations provide different 2D projections of the 3D environment. For a thorough review of approaches based on this principle, we refer to [4].

In this paper, we investigate a novelty approach in identifying point localizations from unreliable measurements, which is one of the basic machine learning problems in robotics [5], in a TDOA system [6] or in motion capturing systems [7]. We apply artificial neural networks that represent biologically inspired machine learning approaches [8]. The approach is a well-known approximation method for datasets, where samples of inputs and correlated output are available.

2.2 Classical Approach

Let us consider a set of $n + 1$ transmitters in 3D space where the ith one has coordinates $P_i = (x_i,\ y_i,\ z_i)$, $0 \le i \le n$. Let us consider a receiver in the same space. We can abbreviate r_i the distance from the ith transmitter to the receiver. The position of the receiver $P = (x,\ y,\ z)$ can be found solving a system of quadratic equations (1):

$$(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2 = r_i^2 \quad 0 \le i \le n. \tag{1}$$

Subtracting the first equation from each of the rest and using the substitutions (2)

$$\begin{aligned} a_i &= 2(x_0 - x_i) \\ b_i &= 2(y_0 - y_i) \\ c_i &= 2(z_0 - z_i) \\ d_i &= r_i^2 - r_0^2 + x_0^2 - x_i^2 + y_0^2 - y_i^2 + z_0^2 - z_i^2 \\ & 1 \le i \le n \end{aligned} \tag{2}$$

we obtain the following linear system (3):

$$\begin{gathered} a_i x + b_i y + c_i z + d_i = 0 \\ 1 \le i \le n \end{gathered} \tag{3}$$

Let us consider $n = 3$ and transmitters in the space in the positions $P_0 = (0,\ 0,\ 0)$, $P_1 = (\mathrm{R},\ 0,\ 0)$, $P_2 = (0,\ \mathrm{R},\ 0)$, $P_3 = (0,\ 0,\ \mathrm{R})$. In this case we obtain the following solution (4):

$$x = \frac{r_0^2 - r_1^2 + R^2}{2R}, \quad y = \frac{r_0^2 - r_2^2 + R^2}{2R}, \quad z = \frac{r_0^2 - r_3^2 + R^2}{2R}. \tag{4}$$

For more transmitters, the system (2) is overloaded. Due to a discretization error we are able to find only an approximate solution, e.g. we obtain the least square solution according to the condition (5):

$$\Phi = \sum_{i=1}^{n} (a_i x + b_i y + c_i z + d_i)^2 \to \min. \tag{5}$$

Using the standard technique for searching the argument of extreme value (6),

$$\frac{\partial \Phi}{\partial x} = \frac{\partial \Phi}{\partial y} = \frac{\partial \Phi}{\partial z} = 0 \tag{6}$$

the desired solution has to fulfill the following requirements (7).

$$\begin{aligned}
\sum_{i=1}^{n} a_i^2 x + \sum_{i=1}^{n} a_i b_i y + \sum_{i=1}^{n} a_i c_i z &= \sum_{i=1}^{n} a_i d_i \\
\sum_{i=1}^{n} a_i b_i x + \sum_{i=1}^{n} b_i^2 y + \sum_{i=1}^{n} b_i c_i z &= \sum_{i=1}^{n} b_i d_i \\
\sum_{i=1}^{n} a_i c_i x + \sum_{i=1}^{n} b_i c_i y + \sum_{i=1}^{n} c_i^2 z &= \sum_{i=1}^{n} c_i d_i
\end{aligned} \tag{7}$$

2.3 Backpropagation Neural Network Approach

A backpropagation neural network works as follows. Each neuron receives a signal from neurons in the previous layer, and each of those signals is multiplied by a separate weight value. The weighted inputs are summed and passed through a limiting function (e.g. sigmoid function), which scales the output to a fixed range of values. The received output is then sent to all of neurons in the next layer. Thus using the network to solve a problem, input values are applied to all inputs of the first layer, the signals are allowed to propagate through the network and output values are read.

Since real uniqueness or 'intelligence' of the network exists in the values of weights between the neurons, we need a method of adjusting the weights to solve a particular problem. For this type of networks, the most common learning algorithm is called backpropagation (BP) [8]. A BP network learns by examples, which is a learning set that consists of some input examples and the known desired output for each case. So, we use these input-output examples to show the network which type of behavior is expected, and the algorithm allows the network to adapt.

The backpropagation algorithm works in the following steps [8]. When an input vector $\boldsymbol{I} = (I_1, \ldots, I_n)$ is fed to the input layer, the weighted sum Net_j of the input to the jth neuron in the hidden layer is given by (8):

$$Net_j = \sum_i w_{ij} I_i + \theta_j \tag{8}$$

where θ_j is a bias of the jth neuron and w_{ij} is the appropriate weight value. Equation (8) is used to calculate the aggregate input to the neuron. The '*Net*' term, also known as the action potential, is passed onto an appropriate (sigmoid) activation function. The resulting value from the activation function determines the neuron's output (9). Similarly, Eqs. (8) and (9) are used to determine the output value for node k in the output layer.

$$O_j = \left(1 + e^{-Net_j}\right)^{-1} \tag{9}$$

If the actual activation value of the output neuron, k, is O_k, and the expected target output for node k is t_k, the difference between the actual output and the expected output is given by (10):

$$\Delta_k = t_k - O_k \tag{10}$$

The error signal for node k in the output layer can be calculated as (11):

$$\delta_k = \Delta_k O_k (1 - O_k) \tag{11}$$

where the $O_k(1 - O_k)$ term is a derivative of the sigmoid function. Next, the change in the weight connecting hidden neuron j and output neuron k is proportional to the error at neuron k multiplied by activation of node j. The formulas used to modify the weight, w_{jk}, between the output neuron, k, and the hidden neuron, j is (12):

$$\Delta w_{jk} = \alpha \cdot \delta_k I_j \tag{12}$$

where α is the learning rate. The error signal for neuron j in the hidden layer can be calculated as follows (13).

$$\delta_k = (t_k - O_k)\, O_k \sum_k w_{jk} \delta_k \tag{13}$$

where the 'sum' term adds the weighted error signal for all neurons, k, in the output layer. As before, the formula to adjust the weight, w_{ij}, between the input neuron, i, and the hidden neuron, j is (14):

$$\Delta w_{ij} = \alpha \cdot \delta_j I_i \tag{14}$$

Finally, backpropagation is derived by assuming that it is desirable to minimize the error on the output nodes over all the patterns presented to the neural network. The following equation is used to calculate the error function, E, for all patterns (15). Ideally, the error function should have a zero value if the neural network has been correctly trained. This, however, is numerically unrealistic.

$$E = \frac{1}{2} \sum_{pattern} \sum_k (t_k - O_k)^2. \tag{15}$$

3 Point Localization Via Neural Network

The chapter introduces an experimental study of a hybrid intelligent system for point localization developed via neural networks. We proposed the system topology containing four reference positions, which define the domain space. For this reason, the coordinate system consists of four reference points as follows: $P_0 = (0, 0, 0)$, $P_1 = (1, 0, 0)$, $P_2 = (0, 1, 0)$, $P_3 = (0, 0, 1)$.

The proposed system for point localization is based on neural networks that are able to quantify coordinates (x, y, z) of randomly generated points on the basis of their distances from defined reference points. We used a multilayer neural network with one hidden layer that was adapted by the backpropagation algorithm [8]. The neural network parameters were the following:

- Input layer: 4 units (distances between generated points and defined reference points)
- Hidden layer: 6 units
- Output layer: 3 units (x y z-coordinates of generated points)
- Activate function: a sigmoid
- Learning rate: 0.1.

The philosophy of the application is simple. The distance between randomly generated points and reference points (P_0, P_1, P_2, P_3) is calculated from the orthogonal xyz-coordinate system in 3D Euclidean space. The proposed system is able to transform these values to coordinates (x, y, z). The domain space is determined by a cube whose vertices are formed by defined reference points. The maximum distance between two points in the defined space domain is $\sqrt{3}$ which equals to a space diagonal of a cube with side length $s = 1$.

Each training pattern consists of four input components (distances between a randomly generated point and defined reference points) and three output components (x, y, and z coordinates of the generated point in the domain space). The neural network was adapted by a set of 1,000 training vectors, which uniformly cover the whole domain space. The condition of end of the adaptation algorithm specified the limit value of the overall network error (15), $E < 0.04$. It concerns perfect training set adaptation.

4 Comparative Experimental Study

In the test phase, we used the adapted neural network. Outcomes from the neural network are x, y, and z coordinates of all test points. From these resulting values, Euclidean distances between all test points and point $P_R = (1/2, 1/2, 1/2)$ are calculated. The point P_R is located at the centroid of the cube representing the domain space (Fig. 1). Obtained neural network experimental results were compared with the classical approach.

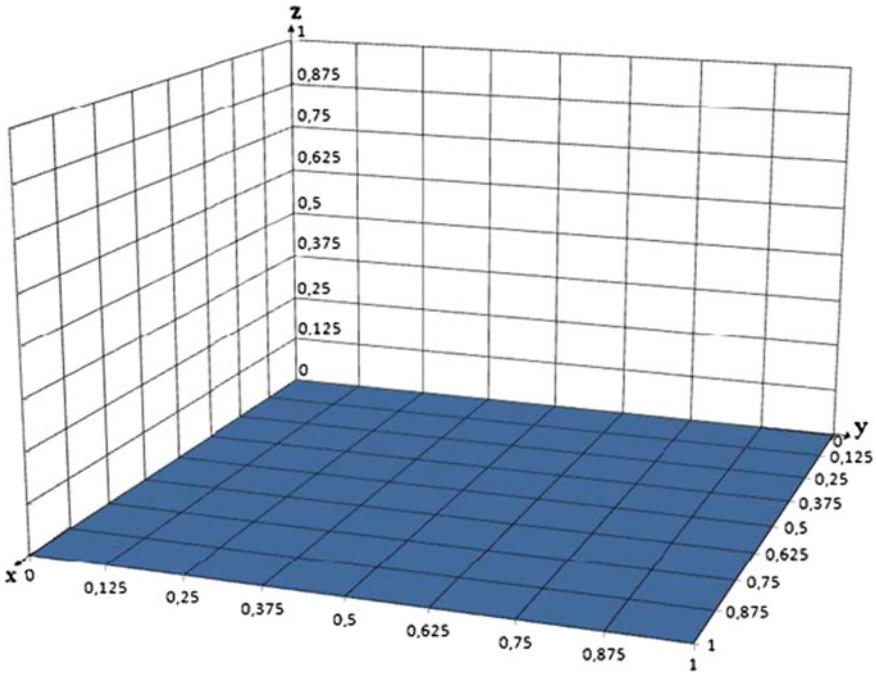


Fig. 1 Grid of test points' localizations

4.1 Point Localization from Accurate Distances

In the experiment, the test set included 729 points from the defined space domain, which are distributed in a regular grid with a step of 0.125. Relevant items are the following values: 0, 0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875, and 1 in all coordinates' directions x, y, and z (Fig. 1). All the test data is from the interval $<0, 1>$ as required by a backpropagation neural network with the sigmoid activation function.

Figure 2 shows the obtained experimental results. The horizontal coordinate represents Euclidean distances between the test points and the point P_R. The maximum distance $D\text{max}$ from the reference point P_R in the defined domain space is a half of the length of the space diagonal of a cube, $D\text{max} = \sqrt{3}/2 \approx 0.8660$.

Accuracy of the results depends on individual distances from the point P_R, as we can see from the graph in Fig. 2. Both graphs represent average error values that are given as differences between the calculated and the experimental value of the distances from the test points to the point P_R. The correct values are displayed on the coordinate x. In the classical approach, these distances are calculated by formulas (4) or (7).

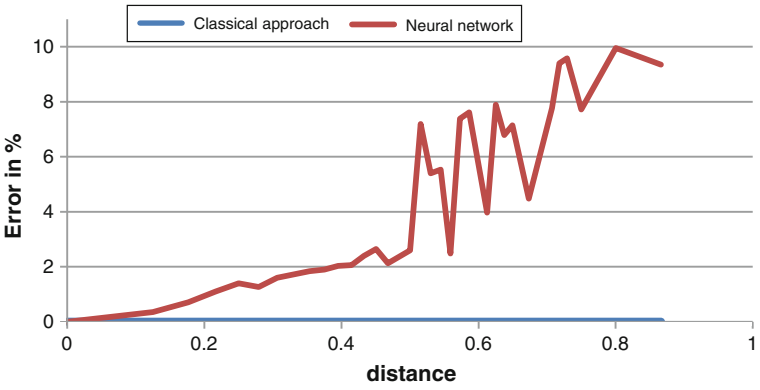


Fig. 2 History of experimental error displaying distances between test points and reference point P_R

Table 1 Experimental error of coordinates x, y, and z

Error	Neural network			Classical approach		
	Δx	Δy	Δz	Δx	Δy	Δz
Min	0	0	0	0	0	0
Max	0.081	0.083	0.082	0	0	0
Average	0.023	0.021	0.023	0	0	0

Table 1 shows experimental errors of coordinates determination, where distances between the test points and the reference points (P_0, P_1, P_2, P_3) were defined unambiguously. There is 'one percent' represented by value 0.005 because the maximum scope equals to coordinates of P_R, which are 0.5 in each coordinate direction.

4.2 Point Localization from Vague Distances

In the experiment, the test set included 125 points from the defined space domain. Their accurate localizations are known due to regular distribution of these points in a regular grid with a step of 0.25 (Fig. 1). Relevant items are the following values: 0, 0.25, 0.5, 0.75, and 1 in all axis directions x, y, and z. Distances between the test points and the reference points (P_0, P_1, P_2, P_3) do not represent exact Euclidean distances, but these values correspond to vague distances gradually with a deviation of 5, 10, and 20 % around the exact values. Obtained experimental results are shown in Figs. 3, 4 and 5. X coordinate represents exact Euclidean distances.

Figures 3, 4 and 5 show error values displayed 'point by point' that represent differences between experimental and correct values of the test points and the

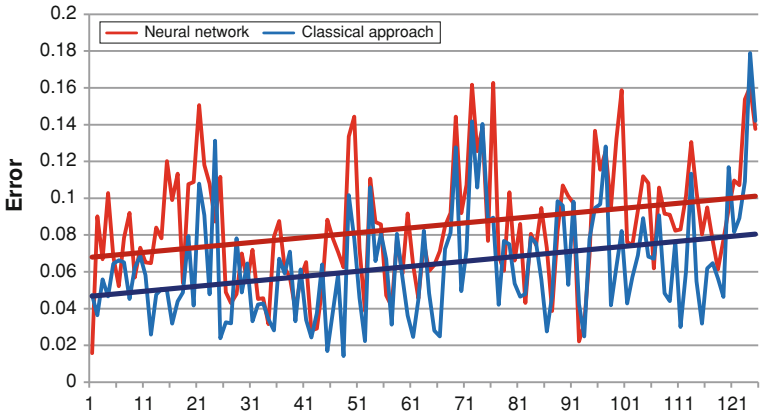


Fig. 3 Experimental results, where distances between the test points and the reference points were identified vaguely with inaccuracy 5 %

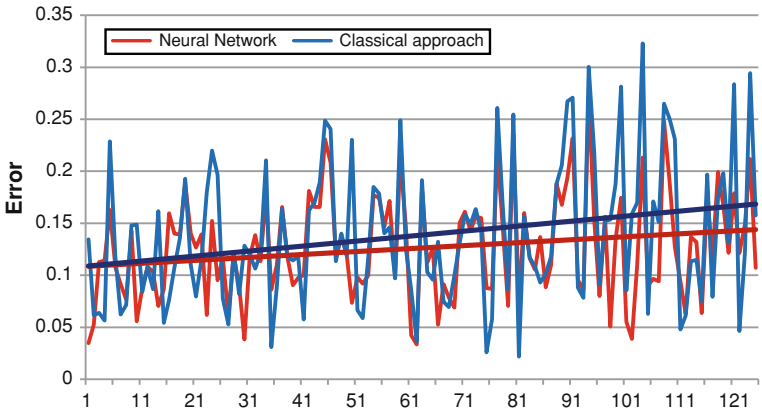


Fig. 4 Experimental results, where distances between the test points and the reference points were identified vaguely with inaccuracy 10 %

point P_R. It is clear that trends are the same in both graphs. Experimental results obtained from the classical approach, and from simulations using neural networks, have a similar course. These figures also revel as that error values related to determining coordinates x, y, and z increases with increasing distance of the test point from the point P_R.

Table 2 shows differences between experimental and correct values of the test points and the point P_R, where distances were identified vaguely with a deviation of 5, 10, and 20 % around the exact values. The maximum distance from the reference point P_R in the defined domain space is $D\max = \sqrt{3}/2 \approx 0.8660$. It is evident that neural networks have achieved more accurate results with increasing inaccuracy in experimental data.

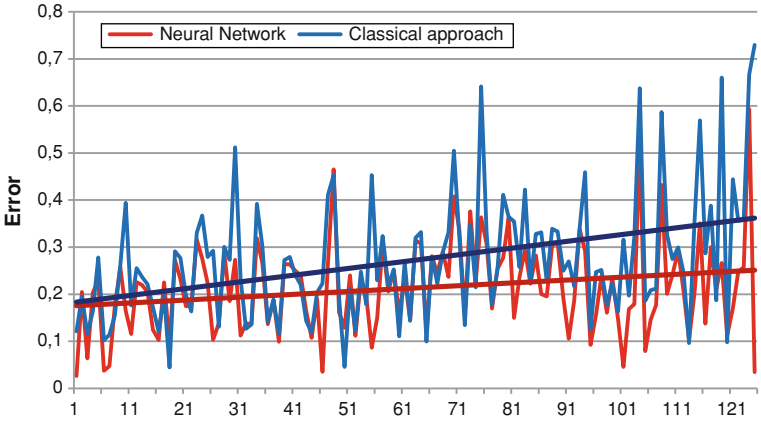


Fig. 5 Experimental results, where distances between the test points and the reference points were identified vaguely with inaccuracy 20 %

Table 2 Experimental results, where distances between the test points and the reference points were identified vaguely with a deviation of 5, 10, and 20 % around the exact values

Error	Neural network			Classical approach		
	5 %	10 %	20 %	5 %	10 %	20 %
Min	0.0156	0.0333	0.02622	0.0141	0.0217	0.0444
Max	0.1627	0.2720	0.5935	0.1788	0.3228	0.7295
Average	0.0845	0.1262	0.2133	0.0636	0.1387	0.2725

5 Conclusion

In conclusion, we would like to compare accuracies that we obtained during our experimental study. Two models were presented. The first one is based on neural networks and the second one represents a classical approach. The classical model calculates Euclidean distances between two points in the defined domain. The proposed hybrid intelligent system for point localization in three-dimensional Euclidean space is based on neural networks. The experimental study includes two kinds of experimental results. The first experimental study represents approaches where distances between the test points and the reference points were defined unambiguously. The second experimental study represents approaches where distances between the test points and the reference points were identified vaguely with inaccuracy about 5, 10, and 20 %. All obtained results are shown in Figs. 2, 3, 4, 5 and in Table 2. In contrast to the classical approach, it is evident that neural networks have achieved more accurate results with increasing inaccuracy from which distances between the test points and the reference points were identified.

Due to a relatively good accuracy and low cost, the proposed system based on neural networks could be used in robotics systems [7], in a TDOA system (Time Difference of Arrival) [6] or as an acoustic Motion Capturing system (MoCap) [9]. Motion Capture is a system for determining positions of points in the space which uses physical properties of audible sound. Since the speed of sound propagation in the environment is constant, it is possible to calculate audio signal's absolute distance according to the degree of its delay. If it happens for at least three transmitters, receivers can determine the position of the spatial coordinates via triangulation [1–3].

Acknowledgments The research described here has been financially supported by University of Ostrava grant SGS/PřF/2014. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not reflect the views of the sponsors.

References

1. Volná, E., Jarušek, R., Kotyrba, M., Janošek, M., Kocian, V.: Data extraction from sound waves towards neural network trainig set. In: Matoušek, R. (ed.) Proceedings of the 17th International Conference on Soft Computing, Mendel 2011, pp. 177–184. Brno, Czech Republic (2011). ISBN 978-80-214-4302-0, ISSN 1803-3814
2. Volná, E., Jarušek, R., Kotyrba, M., Rucký, D.: Dynamical Motion Capture System Involving via Neural Networks. In: Banerjee, S., Erçetin, Ş.Ş. (eds.) Chaos, Complexity and Leadership 2012. Springer Proceedings in Complexity, pp. 563–568. Springer Netherlands (2014). ISBN 978-94-007-7361-5. ISSN: 2213-8684. doi:10.1007/978-94-007-7362-2_62
3. Volná, E., Kotyrba M., Jarušek, R.: Acoustic signal processing via neural network towards motion capture systems. In Arabnia, H.R., Deligiannidis, L., Lu, Tinetti, F.G., You, J. (eds.) Proceedings of the International Conf. on Image Processing, Computer Vision, and Pattern Recognition, pp. 992–996. IPCV'13. CSREA Press, USA (2013). ISBN: 1-60132-252-6
4. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press, Cambridge (2000)
5. Leitner, J., Harding, S., Frank, M., Forster, A., Schmidhuber, J.: Learning spatial object localisation from vision on a humanoid robot. Int. J. Adv. Robot. Syst. (2012)
6. Bouet, M., dos Santos, A.L.: RFID tags: Positioning principles and localization techniques. In: Wireless Days. WD'08. 1st IFIP, pp. 1–5. IEEE (2008)
7. Pattacini, U.: Modular Cartesian controllers for humanoid robots: Design and implementation on the iCub. Ph.D. Dissertation, Italian Institute of Technology, Genova (2011)
8. Fausett, L.: Fundamentals of Neural Network. Prentice Hall, New Jersey (1994). ISBN: 0-13-334186-0
9. Gabai, O., Primo, H.: Acoustic motion capture. US Patent Application 12/746,532, 2008

On the Simulation of the Brain Activity: A Brief Survey

Jaromir Svejda, Roman Zak, Roman Jasek and Roman Senkerik

Abstract This article represents the brief introduction into the issues of simulation of brain activity. Firstly, there is shown a physiological description of the human brain, which summarizes current knowledge and also points out its complexity. These facts were obtained through the technologies, which are intended for observing electrical activity of the brain; for example invasive methods, electroencephalography (EEG) and functional magnetic resonance imaging (fMRI). Then, there are described approaches to simulate the brain activity. First of them is a standard model, which is the basis of most current methods. Second model is based on simulation of brain rhythm changes. Finally, there is discussed possible utilization of complex networks to create a biological neural network.

Keywords Hodgkin–Huxley model · Complex networks · Brain activity · Neural networks

J. Svejda (✉) · R. Zak · R. Jasek · R. Senkerik
Faculty of Applied Informatics, Tomas Bata University in Zlin,
Nam T.G. Masaryka 5555, 760 01 Zlin, Czech Republic
e-mail: svejda@fai.utb.cz

R. Zak
e-mail: rzak@fai.utb.cz

R. Jasek
e-mail: jasek@fai.utb.cz

R. Senkerik
e-mail: senkerik@fai.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_10,

1 Introduction

The oldest written record, in which the word "brain" can be found, originates from seventeenth century BC. It is a papyrus scroll, which contains a description of symptoms, diagnosis and prognosis of a complicate skull fracture of two Egyptians. Since then, the amount of knowledge about human brain has greatly increased. The research has been gradated in the last 170 years. Due to the modern approaches, interest in brain and amount of obtained knowledge has sharply rise in the last 20 years. In spite of the above mentioned facts, the complete understanding of brain activity is still impossible.

Scientific instruments were not able to solve a system as complex as the brain is until the first half of the twentieth century. More appropriate methods were discovered then, etc. cellular automaton and artificial intelligence. However, practice deployments of these methods were made possible at the beginning of nineties. At the same time, there was also gradual development of complex systems and networks, which became an alternative option to modelling of biological neural networks. Successful example of using neural networks in the diagnosis of neurological disorders is described in [1].

Many scientific disciplines deal with the human brain; for example numerical neuroscience, neuro-informatics, informatics or medicine. All of them bring theories, which could explain different brain activities. Numerical neuroscience provides mathematical and biophysical models, which are able to model basic processes in neurons and neural networks. The main goal of neuro-informatics is systematical development of database intended to collect information such as brain morphology, brain parts anatomy and their functional connection, brain electrophysiology, brain states obtained with magnetic resonance and their integration. Further, it seeks to develop tools for modeling, where the aim is the most accurate emulation of brain activity. In Informatics, complex networks are highly suitable to model a complex system among which the brain includes. The contribution of medicine is undisputable especially in brain anatomy research [2].

The human brain is a complex system, which is an object of our research. It is regarded as the most complex system in the universe. The modern science is currently attempting to understand the complex interconnection among individual parts of the brain [3]. Further, it is important to find how this connection contributes to normal or pathological brain function. There are many publications, which deal with description of the brain [1, 3, 4]. There are also a number of models, whose aim is to simulate the brain function as clearly as possible. Some models are shown in this article.

The main aim of this paper is to show possible approaches to design a model of the human brain.

2 Physiological Description of the Brain

The brain activity consists in enormous amount of electrochemical reactions. Therefore, it is a combination of chemical transformations and electrical processes. Changes in electrochemical activity of individual parts are correlated with sensory perceptions, motor activity, changes of attention, etc. [2].

The brain itself is composed of several parts, without which his activity could not be possible. One of its basic structural parts is a neuron. The neuronal cells are characterized by the fact that electrical activity is carried out in them. These cells communicate with each other by electrical signals. According to the last estimate, there are approximately 10^{11} neurons in the brain. Every one of them is connected with thousands of other neurons. The total number of connections (synapses) is roughly 10^{15}. Thus it is a very extensive biological neural network. Every neuron contains much neuritis (neural fibres) which takes care of input and output. The total length of neural fibres is estimated to 3 km per 1 mm^3 [2].

In terms of morphology and electrical properties, there are approximately 1 million types of neurons. The morphological description of pyramidal and inhibitory neurons is currently known. Pyramidal neurons stimulate electrical activity of cerebral cortex, while inhibitory elements restrain communication between pyramidal neurons and they also control excitatory activity. The lack of control over excitatory activity manifests itself in epilepsy [2].

Besides neurons there is the same number of support cells (neuroglia). They have the crucial role especially in brain development, because of their ensuring of correct intergrowth of neurons. However, they are also important in adulthood, because they perform maintenance of neurons (supply of nutrients) and they remove dead ones. Moreover, it was found that glial cells are able to communicate with neurons and they influence their activity [2].

Another part of the brain, which should be mentioned in relation to the neurons, is called vasculature. It ensures the supply of blood to the brain and also a discharge of blood from the brain. It consists of tiny capillaries and small arteries passing through the whole brain. The main arteries are surrounded by vascular smooth muscle, which is able to regulate the blood flow intensity. An increased electrical activity of neurons requires an increased supply of nutrients, which is ensured by vasculature [2].

3 Modelling of the Brain Activity

The field, which deals with mathematical and biophysical modelling of basic processes in neuron and neural networks, is called numerical neuroscience. The selection of the right model depends on the questions, whose answers are searched and on the amount of experimental data. Level of the model is chosen accordingly. The experimental data are used for correct definition of the model, i.e. determination of all free parameters, because the model should have a reasonable predictive value.

3.1 Standard Model

Alan Hodgkin and Andrew Huxley dealt with the issue of origination of neural action potential in the early fifties of the twentieth century. They performed a measurement of a squid nerve. The squid has the hugest axon among all animals. The axon has 1 mm in diameter. Therefore, it is possible to measure required data along the whole axon. Hodgkin and Huxley invented a mathematical description of neural action potential origination for which they won the Nobel Prize in 1963 [5].

Hodgkin–Huxley model consists of four differential equations of the first order given in (1–4). Their final forms were obtained through finding a solution to an alternate electronic diagram of a squid (Fig. 1). More detailed description of this model is given in [5–7].

An alternative connection is based on the following idea. A neuron membrane has an important role in information processing. It consists of two molecule layers (lipids). Protein complexes (intra-membrane proteins) are located between the lipids layers, and they create ion pumps, ion and receptor channels. They also have a very important metabolic function [8].

Ion pumps permanently transport ions Na^+ and K^- through the membrane. Due to this fact, the membrane is permanently polarized. Its surface is electrically positive, while the inner surface is electrically negative. The potential difference between them is 70 mV on average. Ion and receptor channels have a critical role for transferring and elaborating information in mechanism of membrane function [8].

In the diagram shown above, the capacitor represents the neuron membrane. Other branches connected with the capacitor in parallel interpret individual channels through which the respective ions pass. Each channel is specified by both: conductivity and rest potential. The latter arises due to the fact that there is a different concentration of respective ions inside the cells and outside the cell.

The final form of Hodgkin–Huxley equations is:

$$\frac{dU_m}{dt} = \frac{1}{C}\left[\bar{g}_k \cdot n^4 \cdot (U_m - U_k) - \bar{g}_{na} \cdot m^3 \cdot h \cdot (U_m - U_{na}) - g_{bg} \cdot \left(U_m - U_{bg}\right)\right] + I_m \cdot \frac{1}{C}. \quad (1)$$

$$\frac{dn}{dt} = \frac{1}{\tau_n}(n_\infty - n). \quad (2)$$

$$\frac{dm}{dt} = \frac{1}{\tau_m}(m_\infty - m). \quad (3)$$

$$\frac{dh}{dt} = \frac{1}{\tau_h}(h_\infty - h). \quad (4)$$

Formulas and diagram use following notations:

U_m	membrane potential
I_m	membrane current (short stimulus pulse)

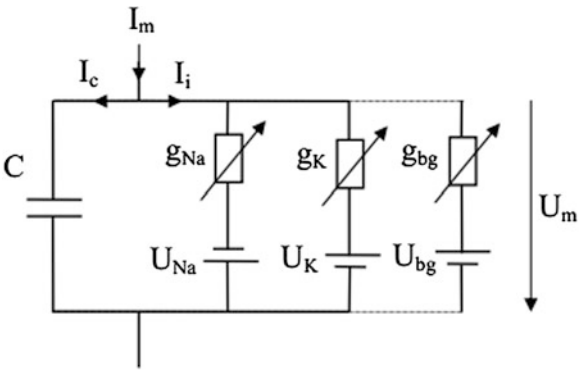


Fig. 1 Alternate electronic diagram of squid's axon [6]

C	surface capacity of the membrane
$\bar{g}_k \cdot n^4,\ \bar{g}_{na} \cdot m^3 \cdot h$	a course approximation of membrane surface conductivities (g_k a g_{na}) for ions of K or Na
g_{bg}	surface conductivity of membrane for other ionic types
U_k, U_{na}, U_{bg}	represent an equilibrium potential for respective types of ions depending on their concentration on both sides of the membrane [6–8].

Standard method of capturing the communication between neurons is based on the fact that each neuron can be divided into some segments in which Hodgkin–Huxley equations are used and linked to each other in a simple way. Coefficients (parameters), which appear in the main body of the neuron and in the axon, are different to those which are used in dendrites. This model is practically unusable for a greater number of neurons, because it becomes too complex with their increasing number. Current super-computers are able to simulate a realistic network of 1 million neurons; each of them is divided to several hundred or thousand segments.

This model requires a high amount of input data to ensure the most accurate prediction. If the data are not available, it is better to use some simpler model. However, this model is the basis for the simulation of neural activity [2].

3.2 Model of Changes in the Brain Rhythms

There are many rhythms that are crucial for human brain; for example natural rhythms, which appear during the sleep, or pathological rhythms accompanying various diseases (Epilepsy, Parkinson's disease, etc.). It is known that synchronization of these rhythms in different areas of the brain is correlated with the success rate of some cognitive tasks. Moreover, some brain diseases are

accompanied by changes in these rhythms. However, the meaning of brain rhythms is not yet fully elucidated [2, 9].

Current models firstly developed on individual neurons and then on simple networks, are able to model changes of brain rhythms. With these models, it is possible to investigate the meanings of individual brain rhythms [2].

3.3 Modeling by Means of Complex Networks

Modern network theory originated along with the discovery of small-world networks and scale-free networks at the end of the second millennium. It is currently the most studied approach to model complex systems. The study of complex networks is applied to a number of different areas such as metabolic system, air transport system, brain, etc. [10].

The complex networks are illustrated by extensive graphs which have a number of common properties. These are abstract models that arise from the combination of mathematical analysis and simulation. The following belong to basic properties of complex networks:

- Average remoteness of nodes (5)—represents the average distance between two nodes in a graph. It can be obtained from the following mathematical form, where N means the total number of nodes and $d_{i,j}$ specifies a distance between ith and jth node:

$$L = \frac{1}{N \cdot (N-1)} \cdot \sum_{i,j \in N, i \neq j} d_{i,j}. \tag{5}$$

- Clustering coefficient (6)—indicates a tightness of binding between the nodes in graph. In other words, it is a relation of the current number of connections between neighbors of the node e_i to the maximal possible number of these connections k_i. The formula has the form:

$$C_i = \frac{e_i}{\binom{k_i}{2}} = \frac{2e_i}{k_i \cdot (k_i - 1)}. \tag{6}$$

There are three basic architectures (See Fig. 2) whose properties are used in complex network study. These were firstly introduced in 1998 by Watts and Strogatz in their article, which dealt with the ubiquity of clustering in the most real networks [11].

The arranged network is usually illustrated in the same way as shown in Fig. 2. The nodes are organized in the circle and each of them is connected to the four

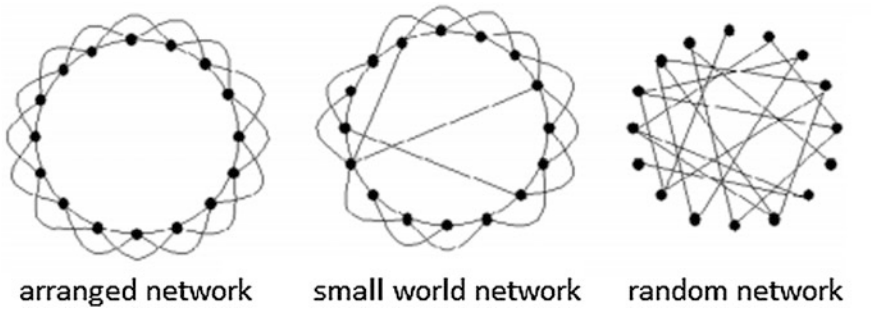


Fig. 2 Representation of three basic complex network architectures [10]

nearest neighbors (two on left side and two on right side). This network is characterized by a high value of C and also high value of L.

The connections between nodes are created randomly in the case of random network. This network has a low value of C and L.

The small world network arises from an arranged network so that two random selected nodes are connected together. Each repetition of this modification causes gradual reduction of L, while C stays practically unchanged; thus it stays on the high value.

Other architecture of complex networks was found in 1999. It is called scale-free and here the distribution of connections between individual nodes is governed by power laws. These can mathematically describe the fact that most nodes in the most the real networks have just few edges and that these small frequent nodes coexist with few large centers which have abnormally high number of edges. The few edges, which mutually connect smaller nodes, are not enough to ensure full interconnection of the network. This function is assured by few centers, which take care of network cohesion. They are called "scale-free", because it is not possible to determinate a typical number of connections or characteristic scale represented by the average node and fixed by the maximum of connection distribution [11]. Internet, social relations between people, air transport system, etc. belong among scale-free real systems.

There are a number of studies dealing with the utilization of complex networks for biological neural network modeling. Their description and also some results can be found in [10]. Most studies have proven that the small-world networks have the same properties as can be observed in biological neural networks, i.e. rapid response of a system and coherent oscillations. But there are also some studies, which discussed the application of scale-free networks for the purposes mentioned above. One of them dealt with the neural network of the simplest multi-cellular organism, whose genome is similar to the human genome. It is a worm called Caenorghabditis elegans with 302 neurons, whose connections are quite accurately mapped out.

The modern network theory is very useful for studying the network, which can be found in the brain. This theory provides efficient realistic models of complex

networks occurring in the brain. Thanks to the continuously increasing number of measurements, it is possible to study topological and dynamical properties of these networks. Further, the theory allows better understanding of the correlations between the network structure and processes, which take place within the networks with corresponding structure. Moreover, it offers potential procedures for forming complex networks and also their reactions to any damage such as random error or targeted attack [10].

4 Sensing of the Brain Activity

In the previous section, we described approaches to modelling the brain activity. The creation of model would not be possible without appropriate equipment, which could provide useful data extracted from the measured brain activity. There are several approaches for sensing brain activity. The most widely used is EEG technology, which belongs among the non—invasive methods. Devices based on EEG technology provide signal with very low voltage amplitude, because the signal has to pass through the relatively low conductive skull. The amplitude ranges from 10 to 100 mV. Recently, we use Emotiv EPOC neuroheadset to obtain EEG signal from the human brain.

4.1 Emotiv EPOC Neuroheadset

Emotiv Corporation developed personal brain—computer interface for human—computer interaction using neuro—technology, which is based on processing of electromagnetic waves occurring in human brain. The interface has wide range of possible applications; for example in interactive games, intelligent adaptive environment, audio visual art and design, medicine, robotics and automotive industry. Moreover, it can be deployed in large amount of scientific research.

Emotiv EPOC neuroheadset (Fig. 3) measures a signal wirelessly transferred to common personal computer. It is a device, which has a set of sensors intended for sensing the activity produced by human brain. Traditional EEG devices requires the use of conductive pasta to improve the conductivity between electrodes and hairs. On the other hand, the neuroheadset do not need any additional tools. It has 14 high resolution sensors, which are placed on optimal positions on the human head (Fig. 4). Moreover, it also includes gyroscope for determinate the position in the area. Each channel has its own label based on its position on the head: AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4. Sampling frequency of the neuroheadset is 2,048 Hz. More information about neuroheadset can be found in [12].

Emotiv provide basic software set containing many tools, which can be used for recording various signals such as electric potential from all 14 sensors, power spectrum of individual EEG channels in real time and rotational acceleration of the

Fig. 3 Emotiv EPOC neuroheadset [12]

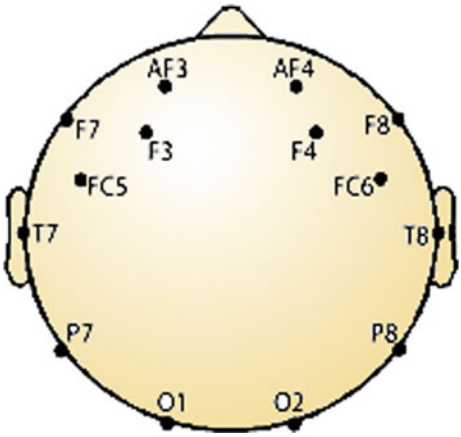


Fig. 4 Placement of electrodes of Emotiv EPOC neuroheadset

head in horizontal and vertical axis using data from gyroscope. All of these outputs are shown in graphs. Data are also available in raw form, which can be used for further analysis. If it is required special functionality, which is not provided by native software, it is desirable to develop own application using Emotiv SDK (Software Development Kit).

Native software consists of three classification suites. Each of them enables the usage of algorithm developed by Emotiv. First of them is Expressive suite, which contains identification system for recognition of facial expression such as smile, eyewink, etc. The muscle signals are used for this purpose. The sources for these signals are obtained by sensors, which are located around the face.

The second suite can be used to measure and identification of emotional state; for example nervousness, alertness, concentration, etc. Therefore, it is called Affective suite. Muscle signals and ocular signals are filtered by specially designed filters; thus, identification algorithm uses clear brain signal.

The last suite is called Cognitive. This classification mode uses whole measured signal, which contains both clear brain signal and muscle signal. Classification algorithm is based on artificial intelligence methods. Type and structure of applied

neural network is patented by Emotiv Corporation; therefore, the specific information about the algorithm is protected.

If it is required other processing of the signal than the native software allows, it may be processed by another software application.

Measured raw data can be subjected to offline analysis to research alternative usage of EEG signal; for example, design a brain model, classification of brain activity, diagnosis brain diseases, etc.

5 Conclusion

Human brain is the most complex known system in the universe. Study of its activity is extremely important mainly due to the more precise diagnosis of brain diseases and their treatment. Furthermore, acquired knowledge could be used in modern technologies with BCI systems, where an interaction between brain and computers appears.

This work introduces three possible approaches to the simulation of brain activity. The first model uses ordinary differential equations for mathematical description of neuron behavior and it is still used as a basis of many other models. Its main disadvantages are high computational complexity and requirement of high amount of input data.

The next model is more abstract than the previous one, because it is based on synchronization of rhythms, which appear in the human brain. Therefore, the brain activity is described by model of changes of brain rhythms instead of description of an individual neuron. The model could help to reveal the meanings of brain rhythms, which are not yet fully elucidated.

The complex network offers another approach to simulate a brain activity. This method is based on the creation of a similar network structure as can be found between neurons in the brain. A number of research studies revealed that the small world networks or scale free networks could have the structure appropriate for the modelling of a real neural network.

Even if there are many different methods, which are appropriate to simulate brain activity, it is still not possible to create a model that would be able to completely capture the behaviour of the human brain. This fact is caused by the high complexity of the brain and by the insufficient performance of computational equipment. Therefore, each simulation model has to be usually simplified so that it could be used with the current computing power. However, it also has to be as close as possible to the real human brain.

We are currently performing the measurement of an EEG signal in our research. Our aim is to discover interesting regularities in the EEG signal waveform, which could contribute to the improvement of current approaches of brain activity simulation. Moreover, these regularities could be used to recognize some specific states of the brain, which can be then used to control the software or equipment connected to the computer.

Acknowledgments This work was supported by Internal Grant Agency of Tomas Bata University under the project No. IGA/FAI/2013/35.

References

1. Adeli, H.: Wavelet-chaos-neural network models for EEG-based diagnosis of neurological disorders. In: Kim, T-H., Lee, Y-H., Kang, B-H., Ślęzak, D. (eds.) Future Generation Information Technology, vol. 6485, pp. 1–11. Lecture Notes in Computer Science. Springer, Berlin Heidelberg (2010). doi:10.1007/978-3-642-17569-5_1
2. Zapotocky, M.: Neuro-Informatics and Modelling of Brain Activity—Academy of Sciences of the Czech Republic (online in Czech). Available from. http://press.avcr.cz/Evropsky_tyden_mozku/zaznamy-z-prednasek/2011/110317-neuroinformatika-zapotocky.html (2011)
3. Sporns, O., Tononi, G., Kötter, R.: The human connectome: A structural description of the human brain. PLoS Comput. Biol. **1**(4), e42 (2005)
4. Damasio, H.: Human Brain Anatomy in Computerized Images. Oxford University Press, Oxford (1995)
5. Nelson, M., Rinzel, J.: The Hodgkin–Huxley model. In: The Book of Genesis, pp. 29–49. Springer, New York (1998)
6. Mrazek, J.: Modelling of Ionic Currents Appear in Isolated Cardiac Cells (Online in Czech). Available from. http://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=18861 (2009)
7. Abbott, L.F., Kepler, T.B.: Model neurons: From Hodgkin–Huxley to hopfield. Statistical Mechanics of Neural Networks, pp. 5–18. Springer, Berlin Heidelberg (1990)
8. Hille, B.: Ion Channels of Excitable Membranes. Sinauer Associates Inc., Sunderland (2001)
9. Buzsaki, G.: Rhythms of the Brain. Oxford University Press, Oxford (2009)
10. Cornelis, S., Reijneveld, J.C.: Graph theoretical analysis of complex networks in the brain. Nonlinear Biomed. Phys. **1**(1), 3 (2007). ISSN 17534631
11. Barabási, Albert-László: Linked: How Everything is Connected to Everything Else and What it Means for Business, Science, and Everyday Life. Plume, New York (2003). ISBN 04-522-8439-2
12. Emotiv | EEG System | Electroencephalography (Online). Available from. http://www.emotiv.com/index.php (2012)

Q-Learning Algorithm Module in Hybrid Artificial Neural Network Systems

Jaroslav Vítků and Pavel Nahodil

Abstract Presented topic is from the research field called Artificial Life, but contributes also to the field of Artificial Intelligence (AI), Robotics and potentially into many other aspects of research. In this paper, there is reviewed and tested new approach to autonomous design of agent architectures. This novel approach is inspired by inherited modularity of biological brains. During designing of new brains, the evolution is not directly connecting individual neurons. Rather than that, it composes new brains by connecting larger, widely reused areas (modules). In this approach, agent architectures are represented as hybrid artificial neural networks composed of heterogeneous modules. Each module can implement different selected algorithm. Rather than describing this framework, this paper focuses on designing of one module. Such a module represents one component of hybrid neural network and can seamlessly integrate a selected algorithm into the node. The course of design of such a module is described on example of discrete reinforcement learning algorithm. The requirements posed by the framework are presented, the modifications on the classical version of algorithm are mentioned and then the resulting performance of module with expectations is evaluated. Finally, the future use cases of this module are described.

Keywords Agent · Architecture · Artificial life · Creature · Behaviour · Hybrid · Neural networks · Evolution

J. Vítků (✉) · P. Nahodil
Faculty of Electrical Engineering, Department of Cybernetics, Czech Technical University in Prague, Technická 2, 16627 Prague 6, Czech Republic
e-mail: vitkujar@fel.cvut.cz
URL: http://cyber.felk.cvut.cz

P. Nahodil
e-mail: nahodil@fel.cvut.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_11,

1 Introduction

This paper deals with design of agent architectures in the domain of Artificial Life (ALife). Design of autonomous agents in this field was inspired by behaviour of animals for a relatively long time. There are numerous agent architectures whose design is inspired in *ethology*, the science field practically defined by Austrian zoologist Konrad Lorenz. Ethology looks on a biological "agent" from the outside and evaluates its *behaviour*.

On the other hand, there is approach which finds inspiration in principles involved in a mammalian brain. Compared to Ethology, the focus here is aimed to inner functionality, rather than external behaviour. It is often argued that intelligent system can observe its environment and understand the situation without producing any behaviour. This way of designing intelligent systems is often called *connectionism*. Recently, a bigger progress in more detailed connectionist models [4, 6] can be made by means of faster computers, or specialized hardware [8].

Each approach has own advantages and drawbacks. Our focus is aimed more towards combining the two above together into new, hybrid architectures. These architectures partly employ ethological principles and partly connectionist ones. Furthermore, this involves also hybrid approaches to designing.

More specifically, this work is a part of our framework called Hybrid Artificial Neural Network Systems (HANNS). This framework combines ANN paradigms with re-usable domain-independent modules implementing more "traditional AI" approaches. This paper will be aimed mainly for creating *module which implements discrete Reinforcement Learning* (RL) algorithm. The following chapters will briefly describe HANNS framework, while the main focus will be put onto integration of RL into the framework. By means of modular approach, the resulting RL module can be used in various of applications. For example, when the inputs of the algorithm are connected to nodes implementing fuzzy-logic operations, some kind of fuzzy-RL can be potentially created [3].

2 Theoretical Background

Rather than designing one particular architecture suitable for a particular task, our research focuses on modular systems [1]. In such systems, currently known modules can be interconnected and used in potentially new ways. As an example of a typical re-usable domain independent sub-system can be seen the Categorizing and Learning Module (CALM) [7].

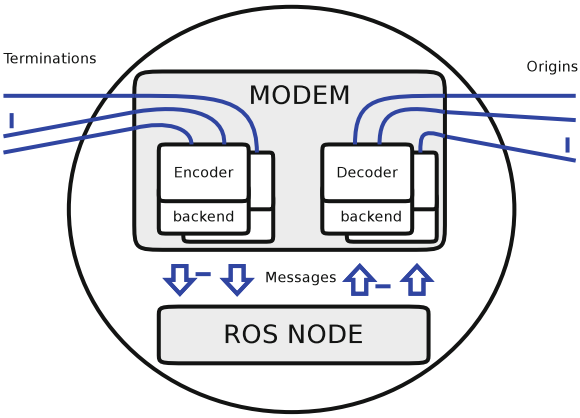


Fig. 1 Scheme of neural module with three inputs, three outputs and arbitrary inner structure. An encapsulated algorithm can be implemented by means of Robotic Operating System (*ROS*)

2.1 Hybrid Artificial Neural Network Systems

The main goal of our framework is to unify external representation of particular modules, so that these modules can be seamlessly connected into bigger systems. Particular sub-systems use for communication the same methods as ANNs and are defined as "Neural Modules". Each Neural Module can have Multiple Inputs/Multiple Outputs (MIMO), either real-valued or spiking type. Neural Module can implement theoretically any component of agent architecture: sensory systems, decision-making modules or actuators. Scheme of Neural Module can be seen in the Fig. 1.

Since these modules can implement various types of algorithms, the inputs to Neural Module are further divided into configuration and data inputs. Configuration inputs are used for setting-up parameters of inner algorithm, while data inputs are used for processing data. These two types may or may not be distinguished, which provides opportunity of changing algorithm parameters online during the simulation.

By employing this representation of particular algorithms, modular agent architectures can be defined as weighted connections between given set of Neural Modules. This approach also provides opportunity of automatic optimization of agent architecture by means of Evolutionary Algorithms (EAs). Despite the benefits mentioned above, the two main challenges for this approach still lie in the following: Need for correct definition of inputs/outputs including their encoding/decoding. Complete domain independence of algorithms used in Neural Modules with requirement of simple configuration.

Since this paper focuses mainly for RL module designed for the HANNS framework, the following text will describe basics of discrete RL and possibilities of its integration into our framework.

2.2 Implementation of the HANNS Framework

In order to be able to simulate networks of highly heterogeneous nodes, the appropriate simulator had to be found. Theoretically, the simulator has to be able to handle these main types of communication: discrete, continuous and spiking. Furthermore, the highest possible re-usability of algorithms was required. For this purpose, the open-source simulator of large-scale neural networks called Nengo[1] was modified. One of the main features is that the simulator was extended for Robotic Operating System[2] (ROS) support. The ROS is decentralized infrastructure based on nodes, which communicate by means of messages over the TCP/IP protocol. In the ROS, each node is separated process (several programming languages supported so far), by connecting several nodes together, a network-like structure can be cerated. Implementation of the HANNS framework employs the ROS and each Neural Module is implemented as ROS node with simple Jython interface for our modification of Nengo.[3] This way, new nodes can be used as a part of HANNS structure, or standalone from the command line.

2.3 Reinforcement Learning

When compared to the knowledge-based AI and to connectionism approaches, several types of RL algorithms have several advantages. Compared to ANNs, these algorithms do not require learning by examples. And compared to planning systems, they do not require even a model of the environment. RL is based only on rewards received as a result of some action executed. This makes RL algorithms suitable for unknown environments and also usable in the HANNS framework. For the integration, the type of RL, called Q-Learning was chosen.

Q-Learning algorithm The Q-Learning algorithm is suitable for online learning without need of environment model—it is model-free approach. During the learning, the algorithm updates the action-value function Q, which represents mapping set of agent's actions A and set of all admissible environment states S to real values according to Eq. (1).

$$Q : A \times S \rightarrow \mathbb{R} \tag{1}$$

Values in the matrix $Q(s, a)$ then define the benefit of each action in a given actual state. When exploiting the knowledge learned the by Q-Learning algorithm, the best action (with the highest value in the matrix) can be selected at each step

[1] University of Waterloo, Simulator of large-scale ANNs Nengo: nengo.ca

[2] Robotic Operating System http://www.ros.org/

[3] Nengoros: http://nengoros.wordpress.com/

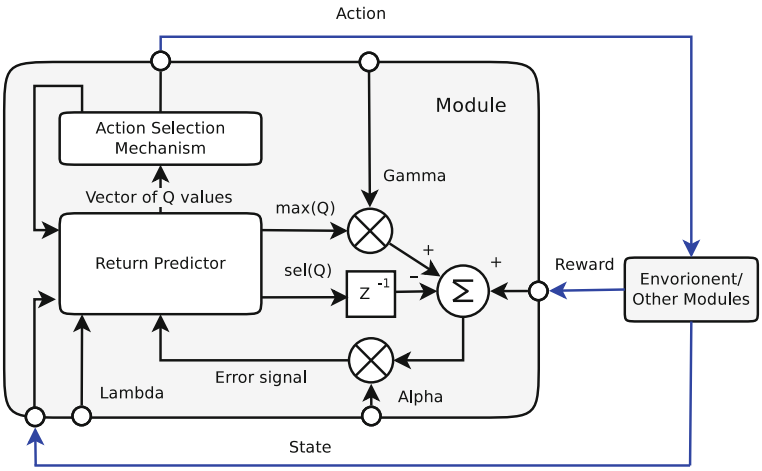


Fig. 2 Scheme of the Q-Learning system. The line labeled "max(Q)" is the prediction of return for the best action. The "Sel(Q)" is the Q actually taken, it is combined with return prediction and reinforcement received from the environment r_i through unit the delay z^{-1}, γ is the discount factor and α is the learning rage. The predictor predicts action values in a current state, based on this information the ASM selects action to be executed

for obtaining best known policy in a given situation. At each step of the algorithm, values in the $Q(s, a)$ matrix are updated according to Eq. (2):

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha[r_t + \gamma \max_a Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]. \tag{2}$$

Here, the $s_t \in S$ is a previous state of the environment, $a_t \in A$ is action which was just executed, r_t is reward received at a result to the action a_t, the current time step t, $\max_a Q(s_{t+1}, a_{t+1})$ is the action with the highest utility value in the current state. There are the following algorithm parameters: $\gamma \in \langle 0; 1)$ is a forgetting factor and $\alpha \in (0; 1\rangle$ is a learning rate, for more information see [9].

The scheme of Q-Learning system and the principle of it's function is depicted in the Fig. 2. The Stochastic Return Predictor (SRP) is composed of Q-Learning algorithm and Action Selection Method (ASM). The ASM selects the action and executes it, RL algorithm observes the reinforcements received and updates the value of the Q function for the previous state according to Eq. (2).

Action Selection Method Since the RL algorithm only learns from the observed experience, the complete SRP is composed of RL algorithm and Action Selection Method (ASM). The simplest case of action selection is the *Greedy* ASM. In each environment state, the $Q(s, a)$ matrix contains (current) utility values of all possible actions. In case of Greedy ASM, simple an action with the highest utility is selected for the following simulation step.

The main drawbacks of this algorithm are the fact that the agent can easily stuck in the local optimum and that the exploration of new states is often not performed.

These drawbacks are solved by the ϵ-Greedy algorithm, where the ε parameter defines amount of randomization. With the probability of ϵ, a random action is selected and with the probability of $1 - \epsilon$ the Greedy action is taken. This helps the agent escape from the local extreme and encourages exploration of new states.

Algorithm Improvements—Eligibility Traces There are several ways how to improve the speed of learning, the one chosen here is called Eligibility Traces. Instead of updating only one value of Q function at each step, all of the values of state-action pairs can be updated simultaneously. This modification is also called *Q-Lambda* algorithm. We can define the change of state based on the current step—δ by rewriting the equation above as follows:

$$\delta = r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t). \tag{3}$$

Now, Eq. (4) has the following form:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\delta. \tag{4}$$

By introducing the error function, which is the fundamental for the eligibility traces-based approaches, we can rewrite the equation as follows:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\delta e(s, a), \tag{5}$$

where the parameter error is defined for each state-action pair as follows:

$$e_t(s, a) = \begin{cases} \gamma\lambda e_{t-1}(s, a) & if \quad (s, a) \neq (s_t, a_t) \\ \gamma\lambda e_{t-1}(s, a) + 1 & if \quad (s, a) = (s_t, a_t) \end{cases} \tag{6}$$

Equation (6) is applied each time step. We can see that each state-action pair has own value of error, which decreases with time. If the state was visited, the error value is set to one.

The main advantage of the eligibility trace compared to one-step Temporal Difference (TD) method is depicted in the Fig. 3.[4] After reaching the reward, the one step TD algorithm stores information about one action. Compare to this, the Sarsa(Lambda) algorithm stores information about considerably bigger part of the path.

3 Our Q-Learning Module Design

This section describes the design of Neural Module for HANNS, which implements domain independent Q-Learning suitable for modular agent architectures. Several design requirements have to be met in order to successfully implement the

[4] Picture borrowed from: http://www.tu-chemnitz.de/informatik/KI/scripts/

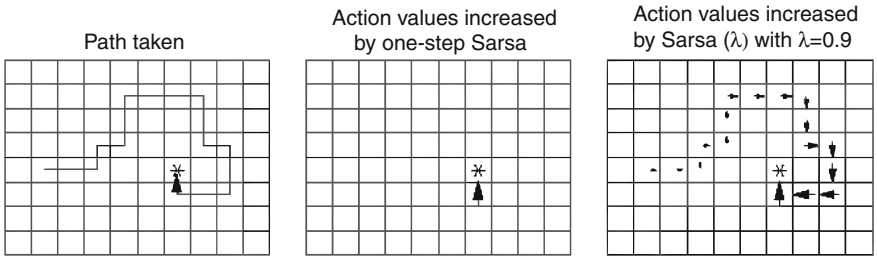


Fig. 3 Example comparing the Sarsa(Lambda) algorithm with the one-step TD method. In case of the Sarsa(Lambda) algorithm, multiple sates-action pairs (trace) are updated after receiving one reward

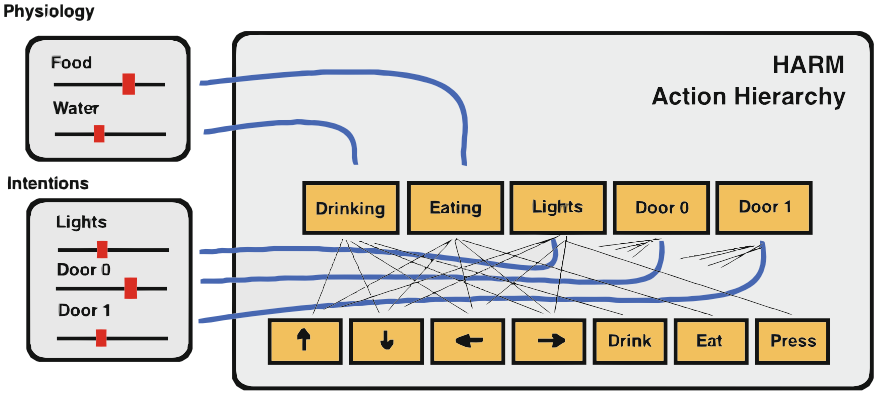


Fig. 4 Scheme of hierarchical RL with one level of action abstraction—only decision spaces composed of states and primitive actions are allowed. These RL modules may compete or cooperate in the architecture for control over the agent

Q-Lambda algorithm in the Neural Module. First, the typical use-case and main requirements for such a Neural Module in the HANNS framework will be described.

Requirements for Neural Module Each Neural Module is designed for the purposes of being connected into bigger network of potentially heterogeneous nodes. An example of hierarchical RL is depicted in Fig. 4. In this architecture, multiple SRPs are simultaneously learning different objectives of behaviour. Each of these RL "modules" has own decision space (matrix mapping state-action pairs to the real value of action utility) and is connected to own source of motivation. The resulting behaviour of the agent emerges from competing or cooperating these SRP modules.

The HANNS framework describes these SRPs as Neural Modules of MIMO (black-box) type. These systems can have *configuration inputs* (defining values of parameters). By convention, each node should provide one output with a real value defining its *Prosperity*. The prosperity is heuristics defining how well a given

module performs in given situation and can be implemented by arbitrary informative function.

Finite Length of Eligibility Trace In our algorithms, we use slight improvement of eligibility traces. Instead of updating values of all state-action pairs, we store only finite number n of currently visited state-action pairs. This approach enables us to improve the one-step TD algorithm significantly, while sustaining the computation requirements low, even in high-dimensional spaces.

Representing the Inputs and Outputs Neural Modules in the HANNS communicates by vector of real values on the interval $\langle 0, 1 \rangle$. Since the module should be as compatible with classical ANN paradigms as possible, the encoding of input/output values is selected *1ofN*. In case of actions, only the currently selected one has non-zero value on its output. Compared to this, array of input values represent array of state variables. Each discrete state variable is sampled with predefined step form the interval $\langle 0, 1 \rangle$.

Operation in non-Episodical Experiments The Q-Learning belongs into the group of algorithms which learn episodically. At the beginning of each episode, the SRP should start to operate from randomly chosen state of the environment. This ensures that the algorithm learns efficiently in the entire state-space. However, in real-life experiments this cannot be provided often. There are two main use-cases of the RL Neural Module:

- **One RL module controls entire agent architecture** (or actions of the rest of the architecture do not interfere with actions produced by the RL). In this case the randomization parameter in the ASM has to be tuned to trade-off between average reward and exploration of the environment. Or, the size of this parameter can be controlled online—e.g. by means of motivation sources [5].
- **Multiple RL modules compete** for the control over agents resulting action (or different subsystems interfere with action selection mechanism). In this case, the randomization of action selection (and therefore exploration) can occur spontaneously, because the agent may not take the greedy action all the time. In this case, the optimal value of randomization has to be determined with respect to the rest of action selection mechanisms.

In architectures, where RL module represents particular behaviour (e.g. "*go for food*") the motivation sources provide efficient way how to weight between competing behaviours (SRPs), the one connected with higher motivation should win and gain control of the architecture. With respect to this principle, we added new parameter to the RL module, called **Importance**. Increasing of this parameter affects two following components in the Neural Module:

- Causes **decrease of** ε **parameter** in the ϵ-Greedy ASM. Therefore, when the behaviour represented by the module has high importance, the exploration is suppressed.
- Causes **increase of value of the selected action**. This ensures that in competition against other RL modules (or other action-selecting sub-systems) has higher chance to win.

Defining Prosperity of Q-Learning Neural Module During the optimization of the agent topology, it is often very suitable to have some notion about performance of particular components of the system, rather than only one value evaluating the overall behaviour. This heuristics (called Prosperity) should represent the performance as accurately as possible, but also remain as general as the algorithm is. The value of prosperity should be dependent on the Neural Modules configuration.

The Q-Learning algorithm has two main objectives: to be able to reach to reach the reward efficiently and be able to operate from any state of the state space. In case of architecture controlled by one RL module in non-episodical experiments, these objectives are antagonistic. This optimal configuration of the node could be found by multi-objective optimization technique, for example by means of Evolutionary Multi-Objective Optimization [2]. In order to keep the complexity of optimization low, we tried to find one-valued representation of prosperity for this algorithm. The resulting selected method is described in the following section.

4 Selected Experiment

For simplicity, the RL module was tested on discrete grid map of size 20×20 with obstacles and one attractor. The agent was equipped with 4 actions (moving in four directions) and the reward was received after reaching the position containing the reward. The presented values are averaged from 5 non-episodical experiments, each started from the same initial state and lasted 80,000 discrete steps. The RL algorithm was configured with the following empirically-estimated constant parameters: $\alpha = 0.5$, $\gamma = 0.3$, $\lambda = 0.04$.

The graphs in Fig. 5 show dependency of two variables on the value of importance, selected from the interval $I \in \langle 0, 1 \rangle$. We can see that combining Mean Cumulative Reward and Mean Coverage together almost eliminates the influence of Importance variable to estimated prosperity of the node, which was our goal.

5 Conclusion

The newly created open-source[5] RL module is implemented in Java language and can be used as a standalone library, ROS node or Neural Module in the Nengoros simulator. The Q(Lambda) algorithm represented as domain-independent neural module can be directly incorporated in new architectures of autonomous agents, such as architectures proposed in [5] or [9]. Interfacing the algorithm with more widely used framework enables user to test its performance in new domains or in

[5] With the BenchMark simulator available at: https://github.com/jvitku/rl

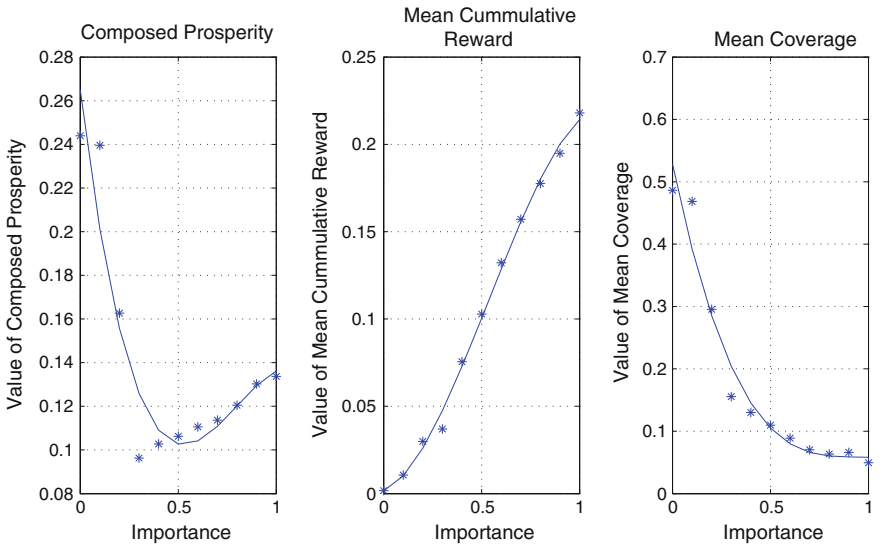


Fig. 5 Dependency of prosperity value composed of two identically weighted antagonistic objectives: mean cumulative reward and mean coverage. Mean coverage represents mean value of states containing at least one non-zero action utility

new combinations with other modules. For instance, by simple connecting it to Neural Modules implementing Fuzzy operations, a simple fuzzy-RL algorithms can be obtained. Furthermore, such definition of algorithm enables us to automatically design novel architectures containing the module, e.g. by means of Evolutionary Algorithms.

Acknowledgments This research has been funded by the Department of Cybernetics, Faculty of Electrical Engineering, Czech Technical University in Prague, partially under the following SGS Projects SGS12/146/OHK3/2T/13 and OHK3-013/14.

References

1. Auda, G., Kamel, M.: Modular neural networks: a survey. Int. J. Neural Syst. **9**(2), 129–151 (1999)
2. Deb, K.: Multi-objective optimisation using evolutionary algorithms: an introduction. In: Wang, L., Ng, A.H.C., Deb, K. (eds.) Multi-objective Evolutionary Optimisation for Product Design and Manufacturing, pp. 3–34. Springer, London (2011)
3. Gu, D., Hu, H.: Reinforcement learning of fuzzy logic controllers for quadruped walking robots (2002)
4. Izhikevich, E.M.: Simple model of spiking neurons. IEEE Trans. Neural Netw. **14**, 1569–1572 (2003)

5. Kadlecek, D.: Motivation driven reinforcement learning and automatic creation of behavior hierarchies. PhD thesis, Czech Technical University in Prague, Faculty of Electrical Engineering (2008)
6. Maass, W.: Networks of spiking neurons: the third generation of neural network models. J. Neural Netw. **10**, 1659–1671 (1996)
7. Murre, J.M.J., Phaf, R.H., Wolters, G.: Calm networks: a modular approach to supervised and unsupervised learning. In: Proceedings of International Joint Conference on Neural Networks, IJCNN, pp. 649–656 (1989)
8. Thomas, D.B., Luk, W.: FPGA accelerated simulation of biologically plausible spiking neural networks. In: Proceedings of the IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM), April 2009
9. Vitku, J.: An artificial creature capable of learning from experience in order to fulfill more complex tasks. Diploma thesis, Czech Technical University in Prague, Faculty of Electrical Engineering, Department of Cybernetics (2011). Supervisor: Doc. Ing. Nahodil Pavel CSc. (in English)

A Probabilistic Neural Network Approach for Prediction of Movement and Its Laterality from Deep Brain Local Field Potential

Mohammad S. Islam, Khondaker A. Mamun, Muhammad S. Khan and Hai Deng

Abstract Prediction of neural activity relating to movement is essential to understanding and treatment of neurodegenerative diseases and cybernetic interfaces. Here we had shown that it is possible to decode deep brain local field potentials (LFPs) related to movements and its laterality, left or right sided visually cued movements using Probabilistic Neural Network (PNN) classifier. The frequency related components of LFPs were extracted using the wavelet packet transform (WPT). Then the signal features were computed as the instantaneous power of each band using the Hilbert Transform (HT) with defined windows for motor response. Based on the extracted feature, PNN classifier was designed and evaluated using 10-fold cross validation method to identify the robustness for predicting movements. The Classification accuracy 82.72 ± 7.2 % achieved for distinguishing movement condition from the rest. While for subsequent discrimination of left and right movement, the accuracy reached up to 74.96 ± 10.5 %. Considering the classification performance (accuracy, sensitivity, specificity and the area under the Receiver Operating Characteristic (AUC) curve), PNN classifier

M. S. Islam (✉) · M. S. Khan · H. Deng
Electrical and Computer Engineering (ECE), Florida International University (FIU), Miami, FL, USA
e-mail: misla004@fiu.edu

M. S. Khan
e-mail: mkhan055@fiu.edu

H. Deng
e-mail: Hai.deng@fiu.edu

K. A. Mamun
Institute of Biomaterials and Biomedical Engineering, University of Toronto, Toronto, Canada
e-mail: k.mamun@utoronto.ca

K. A. Mamun
Department of CSE, Ahsanullah University of Science and Technology, Dhaka, Bangladesh

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_12,

successfully achieved better than chance level. The proposed modality and computational process may promisingly effective and powerful method to open up several possibilities for improving BMI applications, diagnosis of chronic neurological disorders and robust monitoring system with propitious result.

Keywords PNN · DBS · Artificial intelligence-AI · Hilbert transform-HT · LFP · Brain machine interface-BMI

1 Introduction

The primary motor cortex or interchangeably M1 is a valuable part of brain region in human located in the frontal lobe. It works in association with other motor areas including premotor cortex, the supplementary motor area, posterior parietal cortex, and several subcortical brain regions by generating neural impulses. It is main area in the human brain for controlling uncoerced movements [1]. Several neurological disorders can damage it in a massive scale. Therefore, investigations are going on to understand how other deeper areas of brain i.e. Thalamus, Cortical network and Basal Ganglia (BG) involved in originating complex command to perform specific task. These investigations can help us to understand involvement of Basal Ganglia to perform voluntary, self-paced, imagining and involuntary movement paradigm as well as development of neural interface systems [2].

Deep brain stimulation (DBS) [3] is a surgical treatment targeted to globus pallidus or sub-thalamic nucleus (STN) to improve motor function of PD patients and aimed to reduce neuronal abnormalities and other medication-related motor side effects (dyskinesias). Successful stimulation in the deeper areas of brain allows patients to decrease disease related symptoms while improving their ability to perform daily necessary activities.

STN's local field potential (LFP) can be recorded by means of externalization of electrode leads of the wire in time interval between surgery for placement of the electrodes and connection with neuro-stimulation device [3]. LFP is indispensable for comprehending cortical function involved in carrying state of cortical network and local intracortical processing including excitatory or inhibitory interneurons activity and effect of neuromodulatory pathways [4]. Neural information processing systems (NIPS) as well as cortical organizations need to study with great care to understand LFP signal since it is partly ambiguous by nature due to multiple neuronal process has contribution to form it [4]. LFP's real time application with human primary motor cortex (M1) can potentially be important, if it is used in conjunction or combination with primary motor cortex and basal ganglia activities.

Human self-initiated movement is characterized with changes in neuronal or cortical oscillatory activity [3]. Nevertheless, during clicking and continuous voluntary movement frequency dependent de-synchronization and synchronization

can be found in the STN. In addition, STN's LFP signal contains contra and ipsilateral gamma band synchronization during wrist extensions [3]. Online self-paced hand-movement's onset can be predicted form STN's activity using spectral features via wavelet transform and using LVQ network with 95 % sensitivity and 77 % specificity which concludes LFP activity directly or indirectly involve with the process of motor preparation [5]. Prediction of these movement related synchronization and de-synchronization in real time may provide opportunity for treatment of numerous neurological disorders as well open up multiple ways to develop new generation neuro-prosthetic devices (NPD).

Reliable decoding of LFPs oscillatory characteristics during movement intention using signal transformation may provide substantial or additional information about motor control and bilateral coordination system in human.

In this study, LFP's recorded were investigated to recognize sequential occurrence of movement and subsequent laterality using popular signal processing method and probabilistic neural network (PNN) classifier. Proposed PNN architecture for prediction of movement and its corresponding laterality may provide alternative ways for medical professionals and bioinformatics practitioner for pervasive assessment, monitoring and treatment of movement disorders.

The paper organized as follows. Section 2 describes the framework of experimental design and data acquisition system (DAQ). Section 3 discussed with methodology of this experimental work. Classification of movement and its laterality using PNN are presented in Sect. 4. Experimental results and discussions are presented in Sect. 5. Finally with future directions of this study, Sect. 6 concludes our work.

2 Experimental Framework and Data Acquisition System

2.1 Data Recordings with DBS Implantation and Patient's Activity

In this study, three patients with Parkinson's were selected to take part in bilateral implantation of deep brain stimulation electrodes in the subthalamic nucleus (STN) of Basal Ganglia (BG). Local research ethics committee has provided the required permission for experiment and consent from patient had been taken prior the operation. The DBS macroelectrode (Model 3387-Manufacturer: Medtronic Neurological Division, Minneapolis, USA) was implanted bilaterally in the left and right STN's for treatment of the Parkinson's patient. The macro electrode consists of four platinum–iridium cylindrical surfaces (Diameter: 1.27 mm and length: 1.5 mm, center to center spacing: 2 mm, Contact-0 is the most caudal and contact-3 most rostral). Macroelectrodes were inserted after STN had been identified by ventriculography and pre-operative magnetic resonance imaging (MRI) [6]. MRI had confirmed the contact status connected in STN to record LFP signal in both sides. Three adjacent pairs consisting of 4 contacts (positions are 0–1, 1–2 and 2–3)

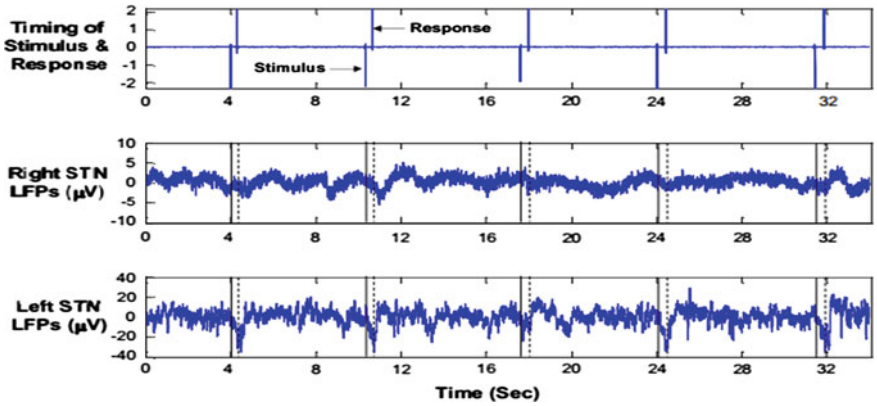


Fig. 1 *Left* and *Right* STN LFPs recording with variation of stimulus and response

were used to record LFPs in bipolar configuration. Bilateral recording of deep brain neuronal activity was performed. While recording LFP signals, CED 1902 amplifiers (×10,000) were employed for amplification of the signal. As an initial artifact removal process, signals were low pass filtered with the range of 0.5–500 Hz and digitized using 12-bit CED 1401 mark II sampling rate at 2,000 Hz. After that, SPIKE 2 (Cambridge Electronic Design-CED, Cambridge, UK) software was used for recording, online monitoring and storing the converted digitized data in the hard drive [6].

During LFP recording (Fig. 1) all subjects were instructed to do finger pressing task. The patients were 60 cm (approx.) far from the computer screen. After that, they were instructed to look at a 10 mm cross located in the center of the screen. The letter A (Height: 8 mm and Width: 7 mm) had appeared on the screen for the duration of 400 ms instantly to the left or right central cross. This was the indicated signal to the patient for ordering of the finger movement. Interval of cues and laterality were provided randomly in the experiment.

3 Experimental Method

3.1 Preprocessing

Raw data of STN's LFP signal were contaminated with high frequency oscillations of as well as movement based surface EMGs. To remove high frequency components, low phase type I Chebyshev filter (zero phase shifting and cut off frequency 90 Hz) were used. Nevertheless, a notch filter of 50 Hz was implemented to remove noise associated with power line. However, to reduce computational complexity and memory space, datasets were digitally re-sampled at 256 Hz for further analysis.

3.2 Feature Extraction Process of STN LFP's

3.2.1 Wavelet Packet Transform

Wavelet transform is a computational tool which transforms sequential data in time axis to time-frequency domain. Wavelet packet transform (WPT) method [7] is a generalization of wavelet packet decomposition that offers excellent multi-resolution time-frequency representation [8] of non-stationary signals. For non-stationary signals like bio signals, wavelet packet transform is a better alternative than short-time Fourier transform (STFT) in terms of multiresolution decomposition capability. In wavelet packet analysis, signal is expressed as a linear combination of time frequency resolution which overcomes some extent of the barriers of Fourier analysis as well as wavelet analysis [9, 10]. Recently WPT can be applied with different biomedical signal detection, classification, compression and noise reduction with desired level of success [11, 12].

With recursive splitting of vector spaces, WPT offers both approximation and details spaces in a binary tree in lieu of dividing only the approximation spaces.

Let $Wm,n(k)$, $n = 0,\ldots, 2m-1$ to represent the WPT coefficients at level m. Below provided equations used to compute the wavelet packet coefficients.

$$W_{m,2n}(k) = \sum_{l=0}^{L-1} h(l) W_{m-1,n}(2k + 1 - lmodN_{n-1}) \tag{1}$$

$$W_{m,2n+1}(k) = \sum_{l=0}^{L-1} g(l) W_{m-1,n}(2k + 1 - lmodN_{n-1}) \tag{2}$$

where $k = 1\ldots N$ and $N_n = N/2_n$. $h(l)$ and $g(l)$ are the impulse responses of low-pass and high-pass filters of the wavelet packets respectively. In this research, discrete Meyer wavelet was used to compute wavelet packet coefficients. This method used here due to the fact that it shows more appropriateness during analysis of event related potentials (ERP), and broadly matches with oscillatory characteristics of STN LFP activity [6]. Using wavelet packet transform (WPT) with discrete Meyer wavelet (demy) at decomposition scale of 5, different frequency band components were extracted as $\delta = 0$–4 Hz, $\theta = 4$–8 Hz, $\alpha = 8$–12 Hz, low $\beta = 12$–20 Hz, high $\beta = 20$–32 Hz, low $\gamma = 32$–60 Hz and high $\gamma = 60$–90 Hz.

3.2.2 Hilbert Transform

The envelope of each frequency component of the reconstructed signal using WPT was computed by applying the Hilbert Transform (HT) [2, 13]. Hilbert Transform is a common and useful tool for the analysis of oscillatory time varying biosignals.

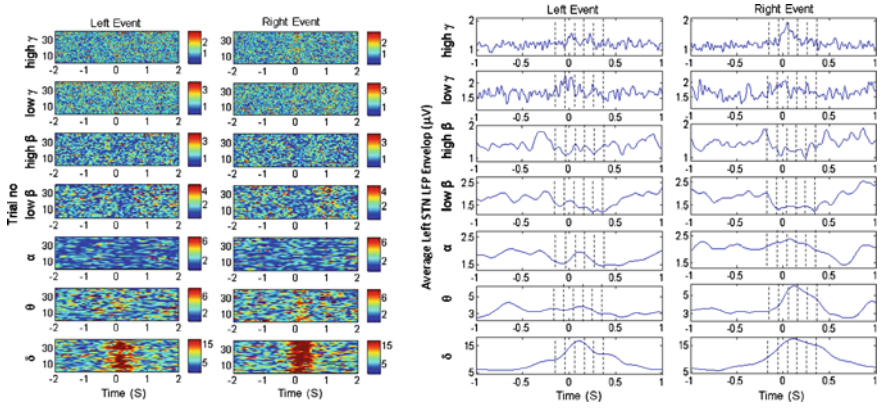


Fig. 2 Spectrogram of instantaneous amplitude of each frequency component computed by Hilbert Transform with 4-s window centered at the time of response of all trails for subject-1 only (both cued *left* and *right* finger clicking events). Average instantaneous amplitude in 2-s window around the time of response for subject-1 only (*right*) [6]

HT is used to form a complex analytic signal composed of the real narrow band time-series and the imaginary part of that HT [6]. The magnitude of the complex analytic signal represents the amplitude envelope of that time series biosignals. If $y_a(l)$ is the computed analytic signal from a complex time varying signal and it can be expressed as $y_a(l) = A(l)exp(i\varphi(l))$. Here, $A(l)$ is the instantaneous amplitude of the signal and $\varphi(l)$ is the phase of that complex signal. Recent research suggested that HT can provide less distortion for getting the envelope of the signal as compared to full wave or half wave rectification of that particular signal [6].

The time-frequency representation of modulated wave for subject-1 is presented in Fig. 2 for each frequency band over all trials during left and right clicking recorded from STN's. It was found that an amplitude decrement had happened in beta (β) frequency band whereas significant increment of amplitude were observed in all other frequency (α, θ and γ bands), most significantly in delta (δ) band. Based on left and right hand cued movement, oscillatory characteristics of LFP signals due to main energy increment(synchronization) or energy reduction (de synchronization) average amplitude of five consecutive windows of length 100 ms were taken as features for classification purpose. Five windows (window size: 100 ms) from −750 to −250 ms were ran for resting condition (before stimulus present) of the patients and five windows were ran from −150 to 350 ms (total length : 0.5 s) for clicking events of the patients (Fig. 2). The five consecutive segments corresponding to each frequency band while clicking of left and right events, STN's LFP signal were defined as—Segment-1 (from left): −150 to −50 ms, Segment-2: −50 to 50 ms, Segment-3: 50–150 ms, Segment-4: 150–250 ms and Segment-5: 250–350 ms. Finally, from each subject and each frequency band a pattern of total seventy (70) features (2 sides × 35 features) from contra and ipsi-lateral (left STN or GPi LFPs and right STN or GPi LFPs) were extracted for classification task with PNN classifier.

4 Classification Using Probabilistic Neural Network

Movement due to LFP activity and its corresponding laterality classification has been carried out using probabilistic neural network (PNN).

4.1 Probabilistic Neural Network

Probabilistic Neural Network (PNN) was first proposed in 1990 by Specht which is capable of classification task of various multiclass problems [14]. PNN is a kind of distance based intelligent neural network algorithm which is composed of numerous interconnected information processing unit called neurons in the successive layers. It is more suitable decision maker used for the detection of human neurological disorders as compared to traditional back-propagation (BP) based neural network since it uses Bayesian strategy [15].

The complete architecture of typical PNN is shown in Fig. 3. Here, R represents the total number features and Q represents the total number of instances used for training and testing. Besides this, K belong the number of classes to be classified in the network. The PNN network consists of three processing and non-processing layers. These are input layer (P), radial basis function (Q) layer and competitive (C) layer consecutively. The input layer does not involve in computation without distributing the inputs to the neurons of the pattern layer. On the other hand RBF layer measure distance between input vector and rows of the weight matrix. Nevertheless, maximum probability of correctness during classification process used to decide in competitive (C) layer.

Details information about probabilistic neural network (PNN) can be obtained from the references [14, 16].

4.2 Design of PNN Classifier

The present study used probabilistic neural network (PNN) with supervised learning scheme implemented for prediction movement and its laterality. The complete classification process was performed in MATLAB (The Mathworks Inc.) R2012b on Intel Pentium Core i7, 3.40 GHz environment.

The prime design consideration for designing PNN are network architecture, dataset (design and test set) and algorithm of training. Figure 4 represents flow diagram of PNN classifier training and testing using 10-fold CV method.

The complete dataset for three subjects were randomly divided into design (training set) and test (evaluation) set. Typically the training set contains more sample vectors as compared to test set. In this study, maximum 279 sample vectors and minimum 162 vectors of both classes considered to train the PNN network for

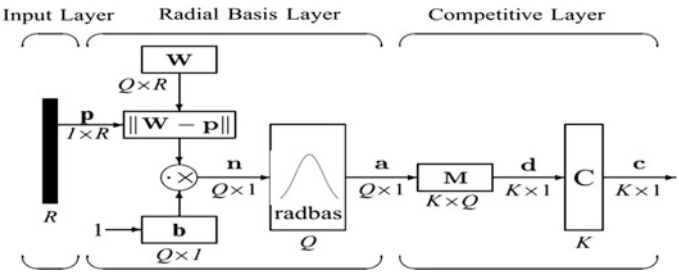


Fig. 3 Illustration of probabilistic neural network (PNN) architecture [15] used for decoding movement and its laterality using LFP signal

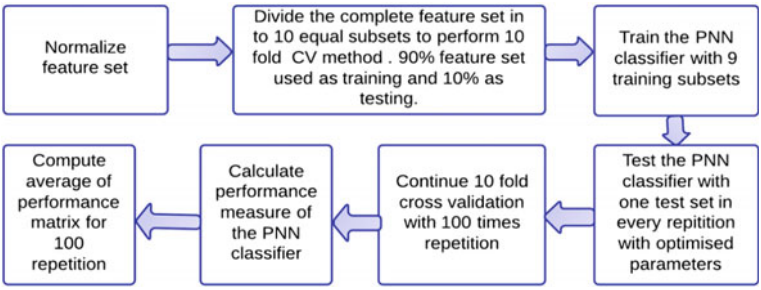


Fig. 4 Flow diagram of proposed PNN classifier training and testing

movement (event and rest) discrimination and maximum 135 vectors and minimum 80 vectors of both classes had taken for training in laterality (left and right) discrimination. The whole classification process was repeated with different step sizes of spread to determine the optimal range. It has been seen that due to sample variation in different subjects, variation of spread was required to train the PNN network to obtain desired performances.

A popular cross validation method (10-fold CV) was used to evaluate the classification accuracy of PNN classifier for each subject to get an unbiased estimation of final outcome. In this method complete dataset were divided into ten subsamples randomly partitioned for each subject. The training, test and validation sets are independent and more reliable results were achieved from the classifier. In this 10-fold CV procedure, at every iteration one of the ten subsets (10 % of whole dataset) is used as the test set and the remaining nine sub-sets (90 %) are used as a training set for learning and generalization of the classifier. After finalizing 10-fold cross validation, average mean square (MSE) error was computed. To evaluate accurate performance of the data set, 100 times repeated 10-fold CV results were recorded for each subject (Event vs. Rest or Left vs. Right) and averaged accordingly to get unbiased estimation of final outcome.

Performance of the proposed PNN classifier was evaluated like other machine learning algorithm using commonly used evaluation parameters (i.e. metrics) such

as total overall accuracy, sensitivity, specificity and the receiver operating characteristics (ROC) curve. Measures of the confusion matrix (CM) are defined as follows:

$$\text{Total overall accuracy} = \frac{(\text{TP} + \text{TN})}{(\text{TP} + \text{TN} + \text{FP} + \text{FN})} \tag{3}$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{5}$$

In the confusion matrix, TP = Number of true positives, FN = Number of false negatives, TN = Number of true negatives, FP = Number of false positives.

5 Experimental Results and Discussions

The complete paradigm of the movement decoding performance measures of all subjects and their corresponding average values are depicted in Fig. 5.

It is clear that subject-1 has more accuracy, sensitivity and specificity as compared to other two subjects. During decoding movement, performance measures for subject-1 seem to generate almost same values. This may happen due to the fact that detection rate of TP nearly same as TN detection rate. Out of 180 test samples, 170 samples have been correctly classified for subject-1. Test MSE has less value compared to other subjects.

After movement decoding, events were passed for laterality decoding using PNN. Classification performance of movement laterality has shown in Fig. 6 (left). After training the classifier with optimized parameter, correct classification rate achieved 74.96, 85.22 % sensitivity and 66.30 % specificity with testing feature set. Based on the information provided in Fig. 6, it can be seen that average classification performance measures for subject-1 is higher than the other subjects. Subject-1 probably contains a lot of features with high predictive power and less overlapping of classes which might good impact to improve performance compared to other subjects. However, we did not estimate the predictive power of features and to rank the features using any feature ranking method. Nevertheless, due to less number of register in each class, we did not perform any statistical significance test of the results to get conclusive approach about error rate and accuracy. More generally speaking, average value of sensitivity is greater than average accuracy and specificity for all subjects. Using same features during classification, it was also illustrated that the performance of the laterality decoding is less conclusive than the performance of movement decoding.

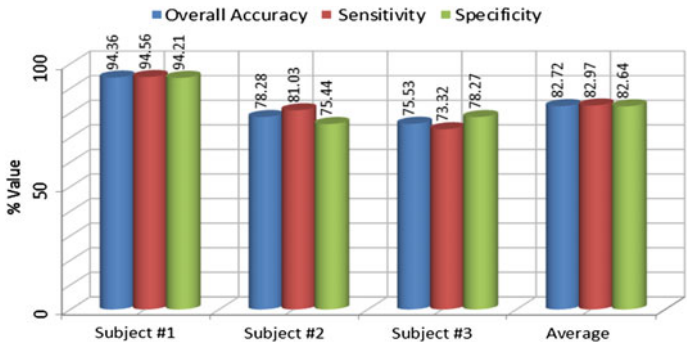


Fig. 5 3-D chart plot of performance outcome for decoding movement using PNN classifier

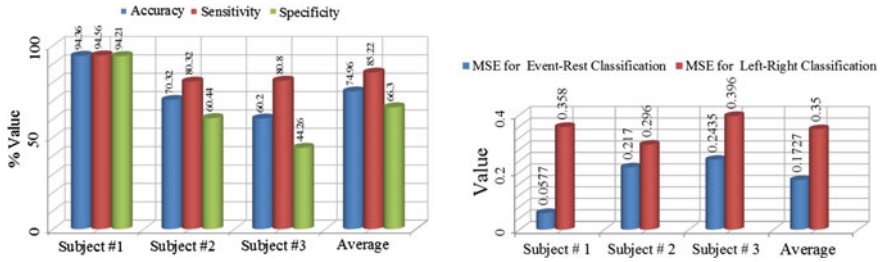


Fig. 6 3-D chart plot of overall accuracy, sensitivity and specificity (*left*) and MSE (*Right*) of PNN network for decoding laterality

The extracted features due movement and its laterality had provided very good decoding performance in terms of mean square error (MSE). Considering cross validated error as shown in Fig. 6 (right), it can be found that maximum average test MSE is achieved 0.2435 for decoding movement and 0.396 for decoding its laterality. High test MSE observed for both movement and laterality in subject-3 as compared to other subjects. Besides this, during experiment it was found that training MSE was very less as compared to test MSE.

Although movement classification has shown very good accuracy as compared to laterality, but still standard deviation (SD) of test error for laterality is 0.050 which is less as compared to movement classification (SD 0.100). However, while discriminating the left and right movement activity, test MSE a bit high but still it has shown good classification rate and sensitivity.

Receiver operating characteristics (ROC) is a very common tool used in data mining to justify classifier performance which is a graphical plot of sensitivity (TPR) and 1-specificity (FPR) of the classifier. From ROC curves (Fig. 7) it was observed that for each subject event vs rest has high AUC value with smooth transition between TPR and FPR. Compared to movement decoding, laterality classification has less AUC value in the figure. This is perhaps due to less training

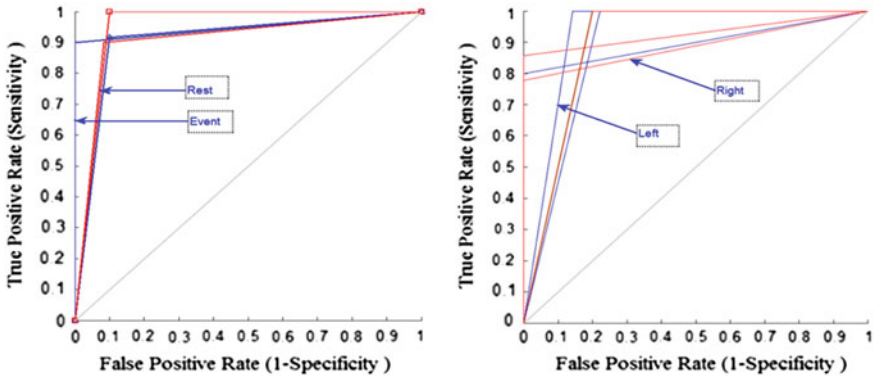


Fig. 7 Area under the receiver operating characteristic (AUC) plot for decoding movement activity and its corresponding laterality. *Left side*—event versus rest activity and *right side*—*left* versus *right* movement activity

sample used to train the PNN network. However, considering AUC value, it can be concluded that proposed PNN classifier has shown good discrimination ability (since average AUC > 0.8 for each subject). High value of AUC substantiated that the implemented classifier is quite robust and accurate.

The extracted features of movement and its laterality activity have very good discrimination capability and have provided excellent decoding performance. However, few disagreement of performance due to less number of subjects, unbalanced and small datasets were encountered for both movement and its laterality recognition.

6 Conclusion and Future Work

In this study, classification using PNN of STN's LFP activity of PD patients during movement and its laterality were addressed. Experimental results and optimum values of statistical parameters substantiated that the proposed PNN approach performs considerably well to achieve distinguishing generalization performance in terms of accuracy, sensitivity, specificity, and receiver operating characteristics which could serve as an alternative approach for movement predictions. With optimized parameters of PNN, average correct classification accuracy reached 82.72 ± 7.2 % for event versus rest and of decoding laterality 74.96 ± 10.5 %. This result further substantiate that deep brain signals contain necessary movement information and it reflects the possibility to use deep LFPs and its decoding methods for developing alternative brain machine interfaces [9]. Should such a study is envisioned that the neural activity in Basal Ganglia can offer unique way to control brain machine interfaces and keep a vast and promising window of opportunity for future generation to explore the development of clinically viable &

cost effective human machine interfaces. Current work exploring the potential of dual recording from the STN and pedunculopontine nucleus (PPN) to create an integrated, wearable system that will detect, fuse, and transmit sensor data from neural (brain) activity, muscle action, and physical motion for pervasive monitoring, diagnostic assessment, and treatment of patients with neurological and/or movement disorders. The proposed PNN approach can be used for discriminating neuronal activity using other type of brain signal. We would pay further attention to investigate the improvement of methods for robust and user friendly Brain-Machine and bidirectional (read out and write in) Brain-Machine-Brain interfaces-BMBI using demand driven deep brain stimulator [17]. We will investigate other robust feature extraction and dynamic feature selection strategy to compare the performance. Finally, we will carry experiment with more subjects and efficient machine learning techniques to make the comparisons more interpretable and meaningful.

Acknowledgements The authors would really grateful to Oxford Functional Neurosurgery Group, University of Oxford, UK to get the dataset for successful completion of this research work.

References

1. Mamun, K.A.; Huda, M.N.; Mace, M.; Lutman, M.E.; Stein, J.; Liu, X.; Aziz, T.; Vaidyanathan, R.; Wang, S., : Pattern classification of deep brain local field potentials for brain computer interfaces. In: Proceedings of International Conference on Computer and Information Technology (ICCIT) pp. 518–523, (2012)
2. Ince, N.F., Gupta, R., Arica, S., Tewfik, A.H., Ashe, J., et al.: High accuracy decoding of movement target direction in non-human primates based on common spatial patterns of local field potentials. PLoS One **5**(12), e14384 (2010). doi:10.1371/journal.pone.0014384
3. Alegre, M., Alonso-Frech, F., Rodríguez-Oroz, M.C., Guridi, J., Zamarbide, I., Valencia, M., Manrique, M., Obeso, J.A., Artieda, J.: Movement-related changes in oscillatory activity in the human subthalamic nucleus: ipsilateral vs. contralateral movements. Eur. J. Neurosci. **22**(9), 2315–2324 (2005)
4. Mazzoni, A., Logothetis N.K., Panzeri, S.: The information content of local field potentials: experiments and models. arXiv preprint arXiv:1206. p. 0560, (2012)
5. Loukas, C., Brown, P.: Online prediction of self-paced hand-movements from subthalamic activity using neural networks in Parkinson's disease. J. Neurosci. Methods. **137**, 193–205 (2004)
6. Mamun, K.A.; Mace, M.; Lutman, M.E.; Stein, J.; Liu, X.; Aziz, T.; Vaidyanathan, R.; Wang, S., : A robust strategy for decoding movements from deep brain local field potentials to facilitate brain machine interfaces. Biomed. Robot. Biomechatron. (BioRob), pp. 320–325, 24–27, (2012)
7. Leman, H., Marque, C.: Rejection of the maternal electrocardiogram in the electrohysterogram signal. IEEE Trans. Biomed. Eng. **47**, 1010–1017 (2000)
8. Xie, H.-B., Zheng, Y.-P., Guo, J.-Y.: Classification of the mechanomyogram signal using a wavelet packet transform and singular value decomposition for multifunction prosthesis control. Physiol. Meas. **30**(5), 441–457 (2009)
9. Mamun, K.A.; Vaidyanathan, R.; Lutman, M.E.; Stein, J.; Liu, X.; Aziz, T.Wang, S., :Decoding movement and laterality from local field potentials in the subthalamic nucleus. In

2011 5th International IEEE/EMBS Conference on Neural Engineering (NER), pp. 128–131, (2011)
10. Amiri, G.G., Asadi, A.: Comparison of different methods of wavelet and wavelet packet transform in processing ground motion records. Int. J. Civil Eng. **7**(4), 248 (2009)
11. Behroozmand, R., Almasganj, F.: Optimal selection of wavelet-packet-based features using genetic algorithm in pathological assessment of patients' speech signal with unilateral vocal fold paralysis. Comput. Med. Biol. **37**, 474–485 (2007)
12. Martinez-Alajarinm, J., Ruiz-Merino, R.: Wavelet and wavelet packet compression of phonocardiograms. Electron. Lett. **40**, 1040–1041 (2004)
13. Marple, S.L.: Computing the discrete-time "analytic" signal via FFT. IEEE Trans. Signal Process. **47**(9), 2600–2603 (1999)
14. Specht, D.: Probabilistic neural networks for classification mapping, or associative memory, In Proceedings of IEEE International Conference on Neural Networks, vol. 1, (1988)
15. Bao, F.S., Lie, D.Y.-C, Zhang Y.: A new approach to automated epileptic diagnosis using eeg and probabilistic neural network. In Proceedings of 20th IEEE International Conference on Tools with Artificial Intelligence (ICTAI '08), vol. 2, pp. 482–486, (2008)
16. Burrascano, P.: Learning vector quantization for the probabilistic neural network. IEEE Trans. Neural Network **2**(4), 458–461 (1991)
17. Gasson, M.N., Wang, S,. Aziz, T.Z., Stein, J.F., Warwick K.: Towards a demand driven deep-brain stimulator for the treatment of movement disorders. In Proceedings of 3rd IEE International Seminar on Medical Applications of Signal Processing, London, pp. 83–86, (2005)

Patterns and Trends in the Concept of Green Economy: A Text Mining Approach

Eric Afful-Dadzie, Stephen Nabareseh and Zuzana Komínková Oplatková

Abstract The term 'green economy' has recently become a topical issue that has engaged the attention of Governments, International bodies and the media. The understanding of this concept and policy concentration is carved in various ways depending on the body that is engaged. There exist varied definitions of the 'green economy' with many associating it directly to agriculture since it has the 'green' connotation. However, despite the varied definitions, one principle that stands out most is the term "Sustainable development" or simply "sustainability. It has 3 pillars namely; social sustainability, economic and environment sustainability. Based on the in-depth of knowledge of the concept of green economy and the commitment of Governments and other international organizations, several policy documents and articles have been published on the web for global consumption. This paper uses the web mining algorithms in-built in the R programming language to mine over 402 English articles on the internet on green economy. It identifies relevant terms and patterns, reveals frequent associative words and gives a conglomerate understanding of the concept. It also brings out the most active participants in the green economic drive and sought to find if by chance any of the three pillars of sustainability would be found in the most frequent terms.

Keywords Green economy · Green growth · Web mining · Text mining · Sustainable development · Economic development

E. Afful-Dadzie (✉) · Z. K. Oplatková
Faculty of Applied Informatics, Tomas Bata University in Zlin, T.G. Masaryka 5555, 760 01 Zlin, Czech Republic
e-mail: afful@fai.utb.cz

Z. K. Oplatková
e-mail: kominkovaoplatkova@fai.utb.cz

S. Nabareseh
Faculty of Management and Economics, Tomas Bata University in Zlin, T.G. Masaryka 5555, 760 01 Zlin, Czech Republic
e-mail: nabareseh@fame.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_13,

1 Introduction and Research Questions

Over the last 2 years, there has been amplification of ideas of a green economy soaring out of its unique place in environmental economics to the mainstream of policy discourse and formulation [1]. The concept of green economy is increasingly sung in the corridors of Governments and international bodies. The discourse in these sectors mostly centres on green economy for sustainable development, economic development and poverty eradication. The current toehold for a green economy concept is supported by prevalent disenchantment [1] with the recent economic crunch.

A green economy is one that leads to an advanced human well-being and social equity which clearly decreases environmental risks and scarcities in ecology [2]. There are various regulatory frameworks that have been outlined by the United Nations Environmental Programme (UNEP) [1] for the transition of Governments to a green economy. Various publications and research reports advocate sound economic growth, social justification and investment while increasing environmental quality and social inclusiveness [1] on transitioning to a green economy.

Various stakeholders also have diverse opinions on the policies of a green economy [2]. These varied views permeate into the level of commitment of various institutions in initiating green economic policies. These opinions also frequently find their way into research articles, web articles, reports and conference proceedings.

The website is loaded with countless articles and publications on the green economy on policy areas, understanding of the concept, governmental involvement and implementation of green economic policies, employment opportunities [1] based on green economy, and the adverse effects of a green economy. The mining of these ideas using a web mining strategy helps to conglomerate the various ideas from the diverse numerous articles flooding the internet on green economy.

Web mining has been a concept that has gripped the attention of many data analyst in recent times. It congregates various methods, techniques, and algorithms to pull out information from the internet (web) on a particular subject matter [3]. Web mining has proven to be a technique that analyses web data that conventional and classic statistical techniques tend to be inefficient in addressing [4].

The web mining technique is used in this paper to mine URLs with reports, stories or articles on green economy. A query is carried out by the use of the R Data Mining tool on the URLs with various results that contain “green economy” for analysis [5].

The paper uses R data mining tool to preprocess the data, stem, and identify relevant terms and patterns [6] in green economy reports and articles on the web. These patterns are analyzed to identify the peculiar issue(s) on green economy mostly talked about for policy focus.

According to [7], trend analysis across document subsets is often employed to derive answers to certain types of questions. Especially in relation to corpus of news stories, [8] proposes that trend analysis be guided by the following questions:

- What is the general trend of the news topics between two periods (as represented by two different document subsets)?
- Are the news topics nearly the same or are they widely divergent across the two periods?
- Can emerging and disappearing topics be identified?
- Did any topics maintain the same level of occurrence during the two periods?

In line with the guiding questions in [8], the paper seeks to ascertain relevant patterns in green economy as reported online to find out the relevant issues in green economy embraced by authors. The study examined online news articles reporting on green economy and applied text mining to analyze text content from the news sources using R-programming language's special text mining plugin for the analysis. The study attempts to answer the following questions:

- What patterns can be found from news sources?
- What are some of the frequent words in the world of green economy?
- Are there some associations among some key words?

The theme at the World Environment Day in 2012 was "*Green Economy: Does it include you?"—Calling for people's engagement.* The key objective was to mobilize and raise awareness about some of the positive environmental actions that are being implemented around the world. This paper therefore attempts to find out whether news articles on the subject of green economy reflect the theme for the 2012 World Environment Day, especially whether online media reports some of the positive actions implemented by UN and other countries [9, 10] or whether the stories are still negative.

2 Methodology

The authors used the text mining and its associated web mining plugin utilities in R-programming language to "scrape" from online news articles, information related to the subject of "green economy". The web scraping was done from google news from the 22 December 2012.

2.1 Text Mining

Text mining is defined as a knowledge-intensive process in which a user interacts with a document collection (corpus) over time by using a set of analytical tools [7]. The evolution of text mining techniques and approaches have made it an inter-disciplinary research field utilizing novel methods from computer science, natural

language, statistics, linguistics among others. The technology is aimed at extracting meaningful information largely from unstructured data mostly by the use of computer software [11, 12]. Unlike traditional content analysis, text mining depends mainly on large sets of textual data such as HTML files, news feeds, blog messages and chat messages, emails and general text files [13]. Text mining approach is often similar to data mining as both seek to identify hidden patterns or trends in data, interpret the results by explaining the patterns and trends in the data [7, 14, 15]. They both apply pre-processing techniques, visualization tools and general pattern-discovery algorithms to automatically identify, extract, manage, and interpret knowledge from data [7, 16]. Text mining techniques and methods have been successfully applied to numerous areas such as in medicine [17, 18], education [11, 19, 20], and consumer research [16, 21].

2.2 Data Collection

The sources for text mining continue to explode especially with tools that are able to scrape text from the internet. Some of the notable tools and programming languages are Weka, RapidMiner, R-programming language, Python among others. The vast amount of textual data in machine readable comes from various sources such as content of websites (Google, Yahoo, etc.), scientific articles, abstracts, books, from CiteSeerX Project, Epub Repositories, Gutenberg Project, etc. There are also news feeds (Reuters and other news agencies), Memos, letters, blogs, forums, mailing lists, Facebook, Twitter, Youtube etc. The research extracted 402 English Google news articles on the subject of "green economy" from all over the world for analysis.

2.3 Corpus Preprocessing

In Fig. 1, the 3 steps that guided the research are illustrated. The ultimate aim was to represent the extracted textual information as term vector of weighted frequencies and to apply useful text mining methods to extract some intelligent information from the news articles.

Step 1a: Corpus Cleaning.
In a typical text preparation process, the raw extracted text is cleaned. This is done by deleting images, the html and xml-tags, some specific characters such as (/, @) and the scripting code. Further on, white spaces, numbers and punctuations are also removed. Depending on the focus of the research work, sparse terms in the corpus may also be removed. Following is tokenization [7, 22] where the character sequences in the corpus is broken into pieces and letters are converted into their lower case forms. Stop word filtering is essential in identifying terms which have

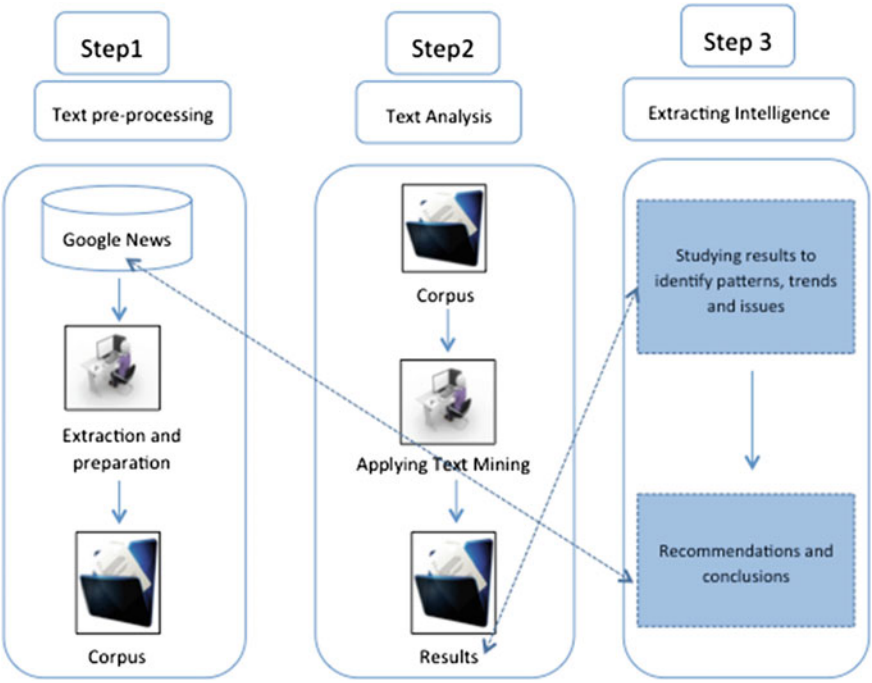


Fig. 1 Text mining process for online news on green economy

little or no content information [23]. Finally word stemming [22] becomes important to reduce the variant form of a word to its common form. For instance, the words "example" and "examples" are stemmed to "example". However, after stemming, the stemmed words are corrected to their original forms to appear "normal".

Step 1b: Vector Space Model.
The next step involves creating a Vector Space Model to begin the analysis. The Vector Space Model (VSM), an algebraic model for representing a document with a vector of index terms was developed in 1975 by Salton et al. for the SMART information retrieval system [24, 25] The basic idea of the VSM is to represent each document in a corpus as a point in a vector space where points that are close together in the space are semantically similar and points that are far apart are semantically distant [26]. The vector of index terms contains only the words that belong to the document and their frequency meaning that a document is represented by the words that it contains. The Vector Space Model ignores punctuations and breaks sentences into tokens thereby losing the order and the grammatical structure.

$$\emptyset : d \rightarrow \emptyset : (d) = (tf(t_1, d), tf(t_2, d), \ldots, tf(t_N, d)) \in \mathbb{R}^N \quad (1)$$

In Eq. (1) $tf(t_i, d)$ is the frequency of the term t_i, *in* d If the dictionary contains N terms, a document is mapped into an N dimensional space [27]. Most often, N happens to be quite large, around a hundred thousand words, and produces a sparse Vector Space Model representation of the document, where few $tf\ (t_i, d)$ are non-zero [28].

A *corpus* is defined as a set of documents, and a *dictionary,* the set of words that appear into the corpus. A document can therefore be seen as a bag of terms [28, 29] represented as a vector, where each component is associated with one term from the dictionary. A corpus of α *documents* can be represented as a *document-term* matrix whose rows are indexed by the documents and whose columns are indexed by the terms. Each entry in position (i, j) is the term frequency of the term t_j in document i.

$$D = \begin{pmatrix} tf(t_1,d_1) & \cdots & tf(t_N,d_1) \\ \vdots & \ddots & \vdots \\ tf(t_1,d_\alpha) & \cdots & tf(t_N,d_1) \end{pmatrix} \quad (2)$$

In Eq. (2), Matrix D can further be used to produce the following:

- the term-document matrix: D'
- the term-term matrix: $D'D$
- the document-document matrix: $D\ D'$,

for the generation of the corpus.

In Eq. (3), document i is represented by a set of terms (n terms in each case).

$$V_i = \begin{bmatrix} Term_1 \\ Term_2 \\ Term_3 \\ \cdots \\ Term_n \end{bmatrix} \quad (3)$$

TF-IDF.

Tf-idf which stands for term frequency-inverse document frequency, is used to determine what words in a corpus of documents might be more favorable to use in a query. It is used to determine the importance of a word to a document in a documents collection or corpus. The tf-idf weight is used as a statistical measure in text mining to evaluate the degree of importance a word is to a document in a corpus. The degree of importance of a word increases proportionally to the number of times a word appears in the document but is offset by the frequency of the word in the corpus. The tf-idf weighting scheme is often what is used by search engines in scoring and ranking a document's relevance given a user query [25]. The formulae for TF and IDF are given in Eqs. (4) and (5) respectively.

Term frequency (TF) is calculated according to (4).

$$tf(t, D) = \frac{F(t, D)}{\max\{F(w, D) : w \in D'\}} \tag{4}$$

where
F() some frequency function
t a specific term
w all the terms in the document
tf term frequency
D collection of documents (corpus)

Inverse Document Frequency (IDF) is given in Eq. (5).

$$idf(\mathrm{t}, \mathrm{D}) = log\left(\frac{|t|}{|D \in D : t \in D|}\right) \tag{5}$$

where the variables mean the same as in Eq. (4).

To produce a composite weight for each term t in each document d, the definitions of the *TF* and the *IDF* are combined as shown in Eq. (6) [30].

$$tfidf(t, D, D) = tf(t, D) * idf(t, D) \tag{6}$$

The example follows. If a document containing 1,000 words has the word *fun* appearing 6 times, the term frequency (i.e., tf) for *fun* is (6/1,000) = 0.006. Now, assuming in 10 million documents the word *fun* appears in 1,000 of these. Then, the inverse document frequency (i.e., idf) is calculated as log(10,000,000/1,000) = 4. The Tf-idf weight, which is the product of these two quantities would result in: 0.006 × 4 = 0.024.

Step 2: Text Analysis.
To seek answers to the research questions of finding patterns and what is trending as far as the development of green economy is concerned, we conducted a text analysis on the 402 corpus of online news articles we extracted. We mainly looked for word frequencies, word associations, document similarity and created word cloud which have foundations with the definitions and principles of green economy.

3 Finding Patterns

Though the concept of green economy seems to have varied definitions, one principle that stands out most is the term "Sustainable development" or simply "sustainability". According to [31] "Sustainable development is development that

Fig. 2 Pillars of sustainability (*Source* http://johngerber.world.edu)

Table 1 Terms that appeared ten times in the corpus

Business	Climate	News	Uae	Global
City	Council	People	World	Hours
Conference	*Economic*	Solar	Carbon	National
Dubai	Environment	Will	Company	Report
Energy	General	Zealand	Countries	Science
Green	Growth	Mining	Economy	*Sustainable*
Investment	Jobs	Suicide	*Environmental*	Volgograd
News	Mining	Security	Today	Year
People	Suicide	Will	Zealand	World
Solar	Uae	Billion	Change	Concerns
Development	Electricity	Government	International	Pathway

meets the needs of the present without compromising the ability of future generations to meet their own needs". The three constituent domains of sustainable [32] development are: social sustainability, economic and environment sustainability as shown in Fig. 2. The paper looked to find if by chance any of the three pillars of sustainability would be found in the most frequent terms. All the three terms including sustainability occur severally in the corpus.

We however, looked for terms that occur at least 10 times to see if all the three terms would occur to give a strong indication of the consistency in the use of the words in online news articles. The result of the terms occurring more than 10 times are shown in Table 1. All the words except the term social occurred in the search.

In word cloud, the importance of each word is shown with a unique font size or colour. The size shows the number of times a word has been applied to a single item and is useful in knowing at a glance the most prominent terms in the corpus. Colours are also used to group concepts into *themes*. Concepts that share the same colour belong to the same theme and are closely related. In Fig. 3, concepts such as *environmental*, *economic*, *climate*, *people*, *change* among others belong to the same theme and therefore are closely related.

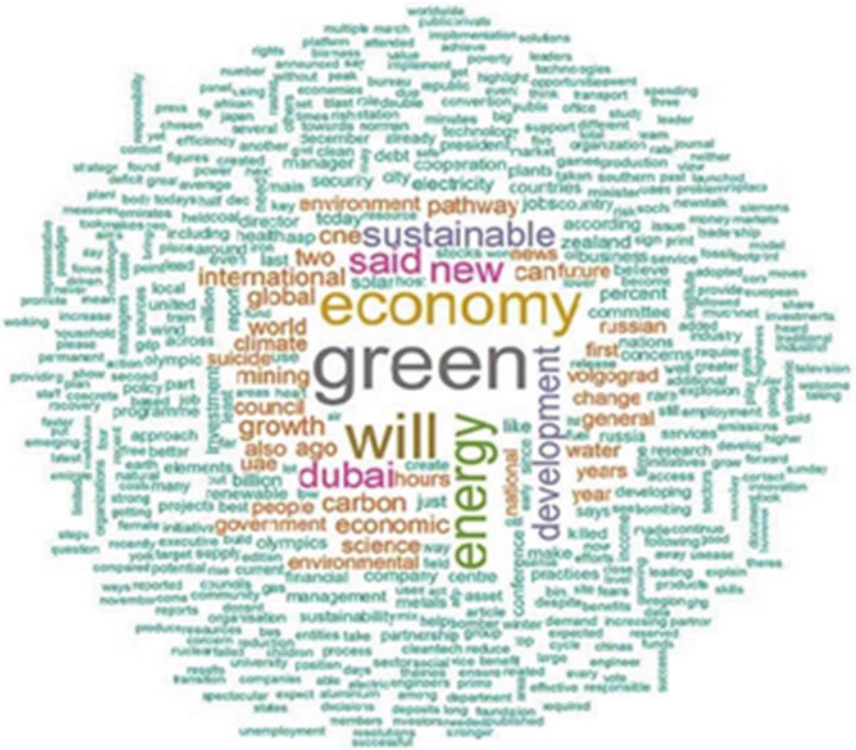


Fig. 3 Word cloud of the concept of green economy as reported online

Table 2 Some interesting terms associated with "sustainable"

Term	Correlation limit	Correlation
Partnership	0.6	0.92
Highness	0.6	0.91
Dubai	0.6	0.90
Energy	0.6	0.80
Economy	0.6	0.74
Environment	0.6	0.60
Reduce	0.6	0.82
Carbon	0.6	0.84
Efficiency	0.6	0.75
Gas	0.6	0.83
Industry	0.6	0.75

We also looked for the association a specific term has with other terms using correlation. For example, with a correlation limit of 0.6 in the term-document matrix, the term *sustainable* has an association with the terms in Table 2.

Table 3 Some interesting terms associated with "environmental"

Term	Correlation limit	Correlation
Social	0.6	0.88
Power	0.6	0.83
Clean	0.6	0.91
Financing	0.6	0.90
Biomass	0.6	0.80
Engaging	0.6	0.74
Electricity	0.6	0.60
Reduce	0.6	0.82
Renewable	0.6	0.84
Natural	0.6	0.75
Battery	0.6	0.88
Risk	0.6	0.75

Respectively in Table 3, a correlation of 0.6 in the term-document matrix shows that the term *environmental* is associated with the terms in the table.

In Table 1 and Fig. 3, three countries Russia, United Arab Emirates and New Zealand feature prominently as far as the concept of green economy is concerned. This could be explained to mean either negative or positive reportage about these countries.

In Tables 2 and 3, a correlation limit was set at 0.6 to mine words that are correlated to "sustainable" and "environmental". This correlation matrix was also carried out to ascertain whether the three pillars of the word "sustainable" correlate with it at 0.6. The results show that two of the pillars of the term "sustainability" namely economy and environment correlate often with the term as shown in Table 2. Similarly, words associated with the term "environmental" are shown in Table 3.

4 Conclusion

The paper used text mining techniques in R-programming language to identify patterns, trends and associations in terms or words that are associated with the concept of green economy. Through the steps of pre-processing, text-analysis and intelligence extraction, we were able to find some interesting patterns and trends. The results show that the term "sustainability" often used in the discussion of "green economy" feature prominently in majority of news articles on the web. The pillars of the term sustainability were also found frequently in the corpus especially the most occurring were "economy" and "environment". The result also further showed that countries currently linked the more to the concept of green economy are Russia, United Arab Emirates and New Zealand. We however cannot conclude for now whether green economy news on these countries is positive or

negative. The research is particularly useful in understanding terms and concepts associated with a topical issue and how it trends on the web. It can also be used to predict the occurrence of certain words in relation to definitions and understandings behind certain concepts.

Acknowledgements This work was supported by Internal Grant Agency of Tomas Bata University IGA/FAI/2014/037 and by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089.

References

1. UNEP.: Towards a green economy: Pathways to sustainable development and poverty eradication, www.unep.org/greeneconomy. ISBN: 978-92-807-3143-9, (2011)
2. Victor, P.A., Jackson, T.: A commentary on UNEP's green economy scenarios. Ecol. Econ. **77**(2012), 11–15 (2012)
3. Velásquez, D.J.: Web mining and privacy concerns: Some important legal issues to be consider before applying any data and information extraction technique in web-based environments. Expert Syst. Appl. **40**(13), 5228–5239 (2013)
4. Markov, Z., Larose, D.T.: Data Mining the Web: Uncovering Patterns in Web Content, Structure, and Usage. Wiley, Hoboken (2007)
5. Thorleuchter, D., Van den Poel, D.: Web mining based extraction of problem solution ideas. Expert Syst. Appl. **40**(2013), 3961–3969 (2013)
6. Feinerer, I.: A text mining framework in R and its applications. Doctoral thesis, WU Vienna, University of Economics and Business. Available at: http://epub.wu.ac.at/1923/, (2008)
7. Feldman, R., Sanger, J.: The Text Mining Handbook—Advanced Approaches in Analyzing Unstructured Data. Cambridge University Press, Cambridge (2006)
8. Montes-y-Gomez, M., Gelbukh, A., Lopez-Lopez, A.: Discovering association rules in semi-structured data sets. In Proceedings of the Workshop on Knowledge Discovery from Distributed, Dynamic, Heterogeneous, Autonomous Data and Knowledge Source at 17th International Joint Conference on Artificial Intelligence (IJCAI'2001). Seattle, AAAI Press, Menlo Park, CA: 26–31, (2001)
9. The United Nations Non-Governmental Liaison Service, UN-NGLS.: World Environment Day 2012. http://www.unngls.org/spip.php?page=article_s&id_article=3928 Accessed 20 Dec 2013
10. United Nations Environment Programme, UNEP.: Global Green New Deal—Environmentally-Focused Investment Historic Opportunity for 21st Century Prosperity and Job Generation. http://www.unep.org/Documents.Multilingual/Default.asp?DocumentID=548&ArticleID=5957&l=en, (2013)
11. He, W.: Examining students' online interaction in a live video streaming environment using data mining and text mining. Comput. Hum. Behav. **29**(1), 90–102 (2013)
12. Liu, B., Cao, S.G., He, W.: Distributed data mining for e-business. Inf. Technol. Manage. **12**(2), 67–79 (2011)
13. Tsantis, L., Castellani, J.: Enhancing learning environments through solution-based knowledge discovery tools. J Spec. Educ. Technol. **16**(4), 1–35 (2001)
14. Guo, J., Xu, L., Xiao, G., Gong, Z.: Improving multilingual semantic interoperation in cross-organizational enterprise systems through concept disambiguation. IEEE Trans. Ind. Inf. **8**(3), 647–658 (2012)
15. Romero, C., Ventura, S.: Educational data mining: A review of the state of the art. IEEE Trans. Syst. Man Cybern. Part C: Appl. **40**(6), 601–618 (2010)

16. He, W., Zha, S., Ling, L.: Social media competitive analysis and text mining: A case study in the pizza industry. Int. J. Inf. Manage. **33**(2013), 464–472 (2013)
17. Li, L., Ge, R., Zhou, S., Valerdi, R.: Guest editorial integrated healthcare information systems. IEEE Trans. Inf Technol. Biomed. **16**(4), 515–517 (2012)
18. Huh, J., Yetisgen-Yildiz, M., Pratt, W.: Text classification for assisting moderators in online health communities. J. Biomed. Inform. **46**(6), 998–1005 (2013)
19. Abdous, M., He, W., Yen, C.J.: Using data mining for predicting relationships between online question theme and final grade. Edu. Technol. Soc. **15**(3), 77–88 (2012)
20. Leong, C.K., Lee, Y.H., Mak, W.K.: Mining sentiments in SMS texts for teaching evaluation. Expert Syst. Appl. **39**(3), 2584–2589 (2012)
21. Mostafa, M.M.: More than words: Social networks' text mining for consumer brand sentiments. Expert Syst. Appl. **40**(10), 4241–4251 (2013)
22. Thorleuchter, D., Van den Poel, D., Prinzie, A.: Mining ideas from textual information. Expert Syst. Appl. **37**(10), 7182–7188 (2010)
23. Thorleuchter, D., Van den Poel, D.: Predicting e-commerce company success by mining the text of its publicly-accessible website. Expert Syst. Appl. **39**(17), 13026–13034 (2012)
24. Salton, G.: The SMART retrieval system: Experiments in automatic document pro-cessing. Prentice-Hall, Upper Saddle River (1971)
25. Salton, G., Wong, A., Yang, C.-S.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975)
26. Turney, P.D., Pantel, P.: From frequency to meaning: Vector space models of semantics. J. Artif. Intell. Res. **37**(2010), 141–188 (2010)
27. Kumar, V.: Text Mining Classification, Clustering, and Applications Data Mining and Knowledge Discovery Series. Chapman Hall/CRC press, Boca Raton (2009)
28. Fan, H., Li, H.: Retrieving similar cases for alternative dispute resolution in construction accidents using text mining techniques. Autom. Constr. **34**(2013), 85–91 (2013)
29. Volna, E., Kotyrba, M., Jarusek, R.: Multi-classifier based on Elliott wave's recognition. Comput. Math. Appl. **66**(2), 213–225 (2013)
30. Stanford University.: TF_IDF. http://nlp.stanford.edu/IR-book/html/htmledition/tf-idf-weighting-1.html Accessed on 29 Dec 2013)
31. WCED: Our Common Future: World Commission on Environment and Development. Oxford University Press, Oxford (1987)
32. Lorek, S., Spangenberg, J.: Sustainable consumption within a sustainable economy—beyond green growth and green economies. J. Clean. Prod. **63**(1), 33–44 (2014)

Utilization of the Discrete Chaotic Systems as the Pseudo Random Number Generators

Roman Senkerik, Michal Pluhacek, Ivan Zelinka and Zuzana Kominkova Oplatkova

Abstract This paper investigates the utilization of the discrete dissipative chaotic system as the chaotic pseudo random number generators. (CPRNGs) Several discrete chaotic maps are simulated, statistically analyzed and compared within this initial research study.

Keywords Chaos · Dissipative systems · Discrete maps · Pseudo random number generators

1 Introduction

Generally speaking, the term "chaos" can denote anything that cannot be predicted deterministically. In the case that the word "chaos" is combined with an attribute such as "deterministic", then a specific type of chaotic phenomena is involved, having their specific laws, mathematical apparatus and a physical origin. The deterministic chaos is a phenomenon that—as its name suggests—is not based on the presence of a random or any stochastic effects. It is clear from the

R. Senkerik (✉) · M. Pluhacek · Z. K. Oplatkova
Faculty of Applied Informatics, Tomas Bata University in Zlin, Nam T.G. Masaryka 5555, 760 01 Zlin, Czech Republic
e-mail: senkerik@fai.utb.cz

M. Pluhacek
e-mail: pluhacek@fai.utb.cz

Z. K. Oplatkova
e-mail: oplatkova@fai.utb.cz

I. Zelinka
Faculty of Electrical Engineering and Computer Science, Technical University of Ostrava, 17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic
e-mail: ivan.zelinka@vsb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_14,

structure of the equations (see the Sect. 4), that no mathematical term expressing randomness is present. The seeming randomness in deterministic chaos is related to the extreme sensitivity to the initial conditions [1].

Till now, the chaos has been observed in many of various systems (including evolutionary one). Systems exhibiting deterministic chaos include, for instance, weather, biological systems, many electronic circuits (Chua's circuit), mechanical systems, such as double pendulum, magnetic pendulum, or so called billiard problem.

The idea of using chaotic systems instead of random processes (pseudo-number generators—PRNGs) has been presented in several research fields and in many applications with promising results [2, 3].

Another research joining deterministic chaos and pseudorandom number generator has been done for example in [4]. Possibility of generation of random or pseudorandom numbers by use of the ultra weak multidimensional coupling of p 1-dimensional dynamical systems is discussed there. Another paper [5] deeply investigate logistic map as a possible pseudorandom number generator and is compared with contemporary pseudo-random number generators. A comparison of logistic map results is made with conventional methods of generating pseudo-random numbers. The approach used to determine the number, delay, and period of the orbits of the logistic map at varying degrees of precision (3–23 bits). Another paper [6] proposed an algorithm of generating pseudorandom number generator, which is called (couple map lattice based on discrete chaotic iteration) and combine the couple map lattice and chaotic iteration. Authors also tested this algorithm in NIST 800-22 statistical test suits and for future utilization in image encryption. In [7] authors exploit interesting properties of chaotic systems to design a random bit generator, called CCCBG, in which two chaotic systems are cross-coupled with each other. A new binary stream-cipher algorithm based on dual one-dimensional chaotic maps is proposed in [8] with statistic proprieties showing that the sequence is of high randomness. Similar studies are also done in [9–11].

2 Motivation

Till now the chaos was observed in many of various systems (including evolutionary one) and in the last few years is also used to replace pseudo-number generators (PRGNs) in evolutionary algorithms (EAs).

Recent research in chaos driven heuristics has been fueled with the predisposition that unlike stochastic approaches, a chaotic approach is able to bypass local optima stagnation. This one clause is of deep importance to evolutionary algorithms. A chaotic approach generally uses the chaotic map in the place of a pseudo random number generator [12]. This causes the heuristic to map unique regions, since the chaotic map iterates to new regions. The task is then to select a very good chaotic map as the pseudo random number generator.

The initial concept of embedding chaotic dynamics into the evolutionary algorithms is given in [13]. Later, the initial study [14] was focused on the simple embedding of chaotic systems in the form of chaos pseudo random number generator (CPRNG) for DE and SOMA [15] in the task of optimal PID tuning.

Several papers have been recently focused on the connection of heuristic and chaotic dynamics either in the form of hybridizing of DE with chaotic searching algorithm [16] or in the form of chaotic mutation factor and dynamically changing weighting and crossover factor in self-adaptive chaos differential evolution (SACDE) [17]. Also the PSO (Particle Swarm Optimization) algorithm with elements of chaos was introduced as CPSO [18] or CPSO combined with chaotic local search [19].

The focus of our research is the pure embedding of chaotic systems in the form of chaos pseudo random number generator for evolutionary algorithms.

This idea was later extended with the successful experiments with chaos driven DE (ChaosDE) [20, 21] with both and complex simple test functions and in the task of chemical reactor geometry optimization [22].

The concept of Chaos DE has proved itself to be a powerful heuristic also in combinatorial problems domain [23].

At the same time the chaos embedded PSO with inertia weigh strategy was closely investigated [24], followed by the introduction of a PSO strategy driven alternately by two chaotic systems [25] and novel chaotic Multiple Choice PSO strategy (Chaos MC-PSO) [26].

The primary aim of this work is not to develop a new type of pseudo random number generator, which should pass many statistical tests, but to try to test, analyze and compare the implementation of different natural chaotic dynamics as the CPRNGs, thus to analyze and highlight the different influences to the system, which utilizes the selected CPRNG (including the evolutionary computational techniques).

3 The Concept of CPRNG

The general idea of CPRNG is to replace the default PRNG with the discrete chaotic map. As the discrete chaotic map is a set of equations with a static start position, we created a random start position of the map, in order to have different start position for different experiments. This random position is initialized with the default PRNG, as a one-off randomizer. Once the start position of the chaotic map has been obtained, the map generates the next sequence using its current position.

The first possible way is to generate and store a long data sequence (approx. 50–500,000 numbers) during the evolutionary process initialization and keep the pointer to the actual used value in the memory. In case of the using up of the whole sequence, the new one will be generated with the last known value as the new initial one.

The second approach is that the chaotic map is not re-initialized during the experiment and no long data series is stored, thus it is imperative to keep the current state of the map in memory to obtain the new output values.

As two different types of numbers are required in computer science; real and integers, the modulo operators is used to obtain values between the specified ranges, as given in the following Eqs. (1) and (2):

$$rndreal = \mathrm{mod}(\mathrm{abs}(rndChaos),\ 1.0) \tag{1}$$

$$rndint = \mathrm{mod}(\mathrm{abs}(rndChaos),\ 1.0) \times Range + 1 \tag{2}$$

where *abs* refers to the absolute portion of the chaotic map generated number *rndChaos*, and *mod* is the modulo operator. *Range* specifies the value (inclusive) till where the number is to be scaled.

Nevertheless there exist many other approaches as to how to deal with the negative numbers as well as with the scaling of the wide range of the numbers given by the chaotic maps into the typical range 0–1:

- Finding of the maximum value of the pre-generated long discrete sequence and dividing of all the values in the sequence with such a maxval number.
- Shifting of all values to the positive numbers (avoiding of ABS command) and scaling.

4 Chaotic Maps

This section contains the description of discrete dissipative chaotic maps, which was used as the chaotic pseudo random generators. In this research, direct output iterations of the chaotic maps were used for the generation of real numbers scaled into the typical range ⟨0–1⟩. Following chaotic maps were used: Burgers (3), Dissipative standard map (4), Lozi map (5) and Tinkerbell map (6).

The *x*, *y* plots of the chaotic maps are depicted in Fig. 1—left (Burgers map), Fig. 2—left (Dissipative standard map), Fig. 3—left (Lozi map), and finally Fig. 4—left (Tinkerbell map). The typical chaotic behavior of the utilized maps, represented by the examples of direct output iterations is depicted in Figs. 1, 2, 3 and 4—right.

The illustrative histograms of the distribution of real numbers transferred into the range ⟨0–1⟩ generated by means of studied chaotic maps are in Figs. 5, 6, 7 and 8.

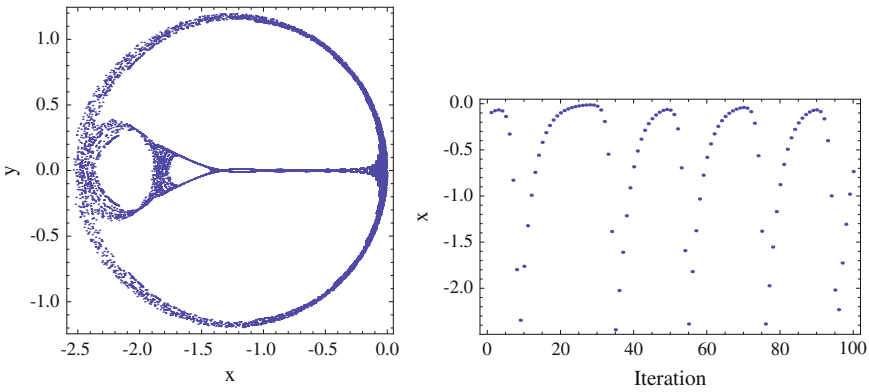


Fig. 1 *x*, *y* plot of the Burgers map (*left*); Iterations of the Burgers map (variable *x*) (*right*)

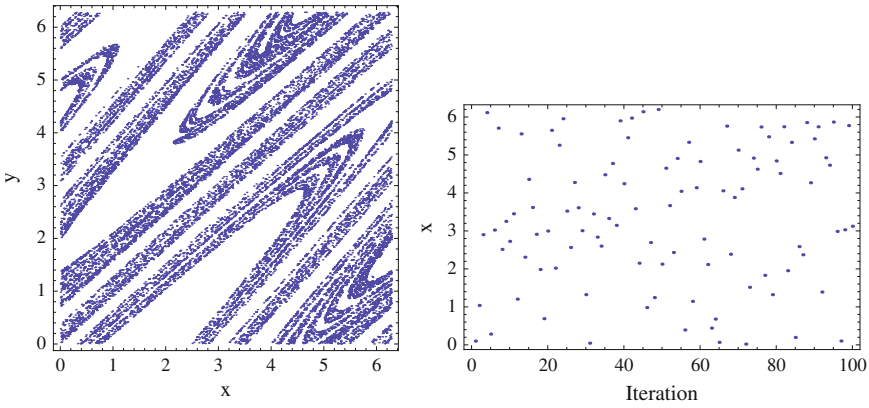


Fig. 2 *x*, *y* plot of the dissipative standard map (*left*); Iterations of the dissipative standard map (variable *x*) (*right*)

4.1 Burgers Map

The Burgers mapping is a discretization of a pair of coupled differential equations which were used by Burgers [27] to illustrate the relevance of the concept of bifurcation to the study of hydrodynamics flows. The map equations are given in (3) with control parameters $a = 0.75$ and $b = 1.75$ as suggested in [28].

$$\begin{aligned} X_{n+1} &= aX_n - Y_n^2 \\ Y_{n+1} &= bY_n + X_n Y_n \end{aligned} \tag{3}$$

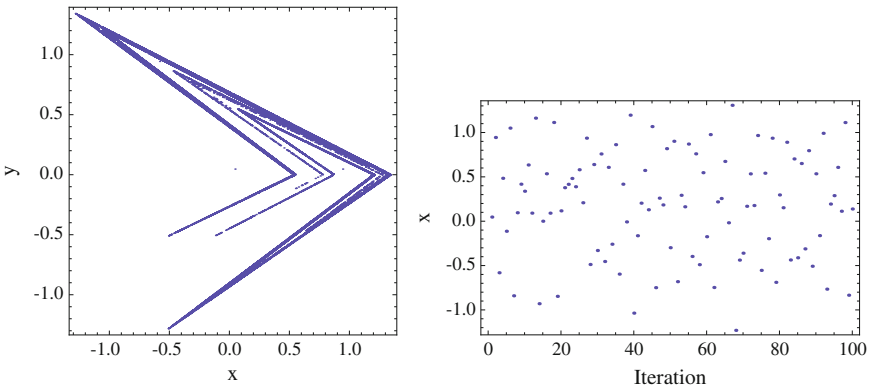


Fig. 3 *x*, *y* plot of the Lozi map (*left*); Iterations of the Lozi map (variable *x*) (*right*)

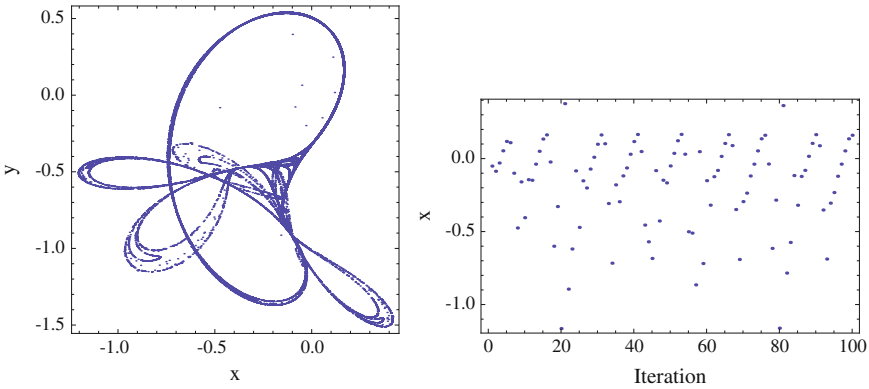


Fig. 4 *x*, *y* plot of the Tinkerbell map (*left*); Iterations of the Tinkerbell map (variable *x*) (*right*)

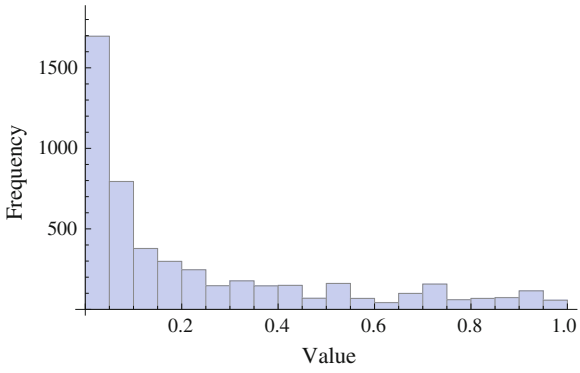


Fig. 5 Histogram of the distribution of real numbers transferred into the range ⟨0–1⟩ generated by means of the chaotic Burgers map—5,000 samples

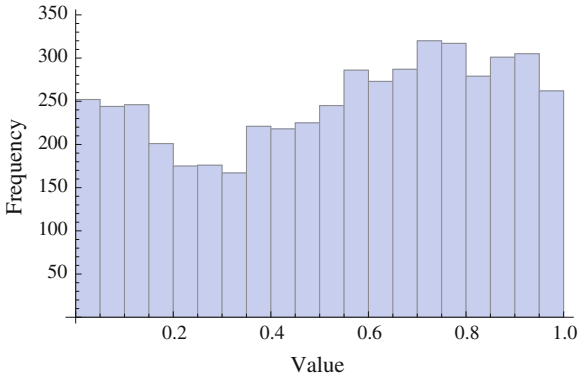


Fig. 6 Histogram of the distribution of real numbers transferred into the range ⟨0–1⟩ generated by means of the chaotic dissipative standard map—5,000 samples

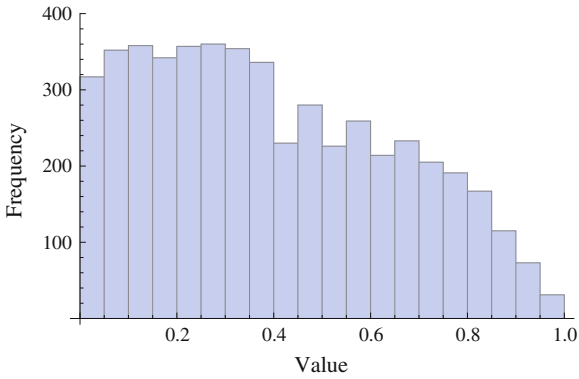


Fig. 7 Histogram of the distribution of real numbers transferred into the range ⟨0–1⟩ generated by means of the chaotic Lozi map—5,000 samples

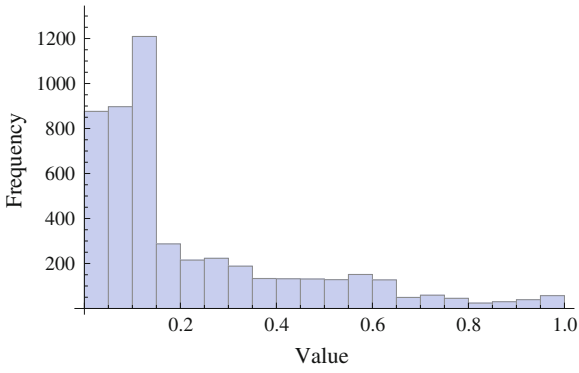


Fig. 8 Histogram of the distribution of real numbers transferred into the range ⟨0–1⟩ generated by means of the chaotic Tinkerbell map—5,000 samples

4.2 Dissipative Standard Map

The Dissipative Standard map is a two-dimensional chaotic map. The parameters used in this work are $b = 0.1$ and $k = 8.8$ as suggested in [28]. For these values, the system exhibits typical chaotic behavior and with this parameter setting it is used in the most research papers and other literature sources. The map equations are given in (4).

$$\begin{aligned} X_{n+1} &= X_n + Y_{n+1} (\mathrm{mod}\ 2\pi) \\ Y_{n+1} &= bY_n + k \sin X_n (\mathrm{mod}\ 2\pi) \end{aligned} \tag{4}$$

4.3 Lozi Map

The Lozi map is a discrete two-dimensional chaotic map. The map equations are given in (5). The parameters used in this work are: $a = 1.7$ and $b = 0.5$ as suggested in [28]. For these values, the system exhibits typical chaotic behavior and with this parameter setting it is used in the most research papers and other literature sources.

$$\begin{aligned} X_{n+1} &= 1 - a|X_n| + bY_n \\ Y_{n+1} &= X_n \end{aligned} \tag{5}$$

4.4 Tinkerbell Map

The Tinkerbell map is a two-dimensional complex discrete-time dynamical system given by (6) with following control parameters: $a = 0.9$, $b = -0.6$, $c = 2$ and $d = 0.5$ [28].

$$\begin{aligned} X_{n+1} &= X_n^2 - Y_n^2 + aX_n + bY_n \\ Y_{n+1} &= 2X_nY_n + cX_n + dY_n \end{aligned} \tag{6}$$

5 Conclusion

This paper was investigating the utilization of the discrete dissipative chaotic system as the chaotic pseudo random number generators. (CPRNGs) Totally four different discrete chaotic maps were simulated, statistically analyzed and compared within this initial research study.

From the graphical comparisons, it follows that through the utilization of different chaotic maps; entirely different statistical characteristics of CPRNGs can be achieved. Thus the different influence to the system, which utilizes the selected CPRNG, can be chosen through the implementation of particular inner chaotic dynamics given by the particular discrete chaotic map.

Furthermore chaotic systems have additional parameters, which can by tuned. This issue opens up the possibility of examining the impact of these parameters to generation of random numbers, and thus influence on the results obtained by means of either evolutionary techniques or different systems from the soft computing/computational intelligence field.

Acknowledgments This work was supported by: Grant Agency of the Czech Republic—GACR P103/13/08195S, is partially supported by Grant of SGS No. SP2014/159, VŠB—Technical University of Ostrava, Czech Republic, by the Development of human resources in research and development of latest soft computing methods and their application in practice project, reg. no. CZ.1.07/2.3.00/20.0072 funded by Operational Programme Education for Competitiveness, co-financed by ESF and state budget of the Czech Republic, further was supported by European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089 and by Internal Grant Agency of Tomas Bata University under the project No. IGA/FAI/2014/010.

References

1. Celikovsky, S., Zelinka, I.: Chaos theory for evolutionary algorithms researchers. In: Zelinka, I., Celikovsky, S., Richter, H., Chen, G. (eds.) Evolutionary Algorithms and Chaotic Systems. Studies in Computational Intelligence, vol. 267, pp. 89–143. Springer, Berlin Heidelberg (2010)
2. Lee, J.S., Chang, K.S.: Applications of chaos and fractals in process systems engineering. J. Process. Control **6**(2–3), 71–87 (1996)
3. Wu, J., Lu, J., Wang, J.: Application of chaos and fractal models to water quality time series prediction. Environ. Model Softw. **24**(5), 632–636 (2009)
4. Lozi, R.: Emergence of randomness from Chaos. Int. J. Bifurcat. Chaos **22**(02), 1250021 (2012)
5. Persohn, K.J., Povinelli, R.J.: Analyzing logistic map pseudorandom number generators for periodicity induced by finite precision floating-point representation. Chaos, Solitons Fractals **45**(3), 238–245 (2012)
6. Wang, X.-Y., Qin, X.: A new pseudo-random number generator based on CML and chaotic iteration. Nonlinear Dyn. **70**(2), 1589–1592 (2012)
7. Narendra, K.P., Vinod, P., Krishan, K.S.: A random bit generator using chaotic maps. Int. J. Netw. Secur. **10**, 32–38 (2010)
8. Yang, L., Wang, X.-Y.: Design of pseudo-random bit generator based on chaotic maps. Int. J. Mod. Phys. B **26**(32), 1250208 (2012)
9. Bucolo, M., Caponetto, R., Fortuna, L., Frasca, M., Rizzo, A.: Does chaos work better than noise? Circuits Syst. Mag., IEEE **2**(3), 4–19 (2002)
10. Hu, H., Liu, L., Ding, N.: Pseudorandom sequence generator based on the Chen chaotic system. Comput. Phys. Commun. **184**(3), 765–768 (2013)
11. Pluchino, A., Rapisarda, A., Tsallis, C.: Noise, synchrony, and correlations at the edge of chaos. Phys. Rev. E **87**(2), 022910 (2013)

12. Aydin, I., Karakose, M., Akin, E.: Chaotic-based hybrid negative selection algorithm and its applications in fault and anomaly detection. Expert Syst. Appl. **37**(7), 5285–5294 (2010)
13. Caponetto, R., Fortuna, L., Fazzino, S., Xibilia, M.G.: Chaotic sequences to improve the performance of evolutionary algorithms. IEEE Trans. Evol. Comput. **7**(3), 289–304 (2003)
14. Davendra, D., Zelinka, I., Senkerik, R.: Chaos driven evolutionary algorithms for the task of PID control. Comput. Math. Appl. **60**(4), 1088–1104 (2010)
15. Zelinka, I.: SOMA—self-organizing migrating algorithm. New Optimization Techniques in Engineering. Studies in Fuzziness and Soft Computing, vol. 141, pp. 167–217. Springer, Berlin Heidelberg (2004)
16. Liang, W., Zhang, L., Wang, M.: The chaos differential evolution optimization algorithm and its application to support vector regression machine. J. Softw. **6**(7), 1297–1304 (2011)
17. Zhenyu, G., Bo, C., Min, Y., Binggang, C.: Self-adaptive chaos differential evolution. In: Jiao, L., Wang, L., Gao, X.-b., Liu, J., Wu, F. (eds.) Advances in Natural Computation, vol. 4221, pp. 972–975. Lecture Notes in Computer Science. Springer, Berlin Heidelberg (2006)
18. LdS, Coelho, Mariani, V.C.: A novel chaotic particle swarm optimization approach using Hénon map and implicit filtering local search for economic load dispatch. Chaos, Solitons Fractals **39**(2), 510–518 (2009)
19. Hong, W.-C.: Chaotic particle swarm optimization algorithm in a support vector regression electric load forecasting model. Energy Convers. Manag. **50**(1), 105–117 (2009)
20. Senkerik, R., Pluhacek, M., Zelinka, I., Oplatkova, Z., Vala, R., Jasek, R.: Performance of chaos driven differential evolution on shifted benchmark functions set. In: Herrero, Á., Baruque, B., Klett, F. et al. (eds.) International Joint Conference SOCO'13-CISIS'13-ICEUTE'13, vol. 239, pp. 41–50. Advances in Intelligent Systems and Computing. Springer International Publishing (2014)
21. Senkerik, R., Davendra, D., Zelinka, I., Pluhacek, M., Kominkova Oplatkova, Z.: On the differential evolution Drivan by selected discrete chaotic systems: Extended study. In: 19th International conference on soft computing, MENDEL 2013, pp. 137–144 (2013)
22. Senkerik, R., Pluhacek, M., Oplatkova, Z.K., Davendra, D., Zelinka, I.: Investigation on the differential evolution driven by selected six chaotic systems in the task of reactor geometry optimization. In: 2013 IEEE Congress on Evolutionary Computation (CEC), 20–23 June 2013, pp. 3087–3094 (2013)
23. Davendra, D., Bialic-Davendra, M., Senkerik, R.: Scheduling the lot-streaming flowshop scheduling problem with setup time with the chaos-induced enhanced differential evolution. In: 2013 IEEE Symposium on Differential Evolution (SDE), 16–19 April 2013, pp. 119–126 (2013)
24. Pluhacek, M., Senkerik, R., Davendra, D., Kominkova Oplatkova, Z., Zelinka, I.: On the behavior and performance of chaos driven PSO algorithm with inertia weight. Comput. Math. Appl. **66**(2), 122–134 (2013)
25. Pluhacek, M., Senkerik, R., Zelinka. I., Davendra, D.: Chaos PSO algorithm driven alternately by two different chaotic maps—an initial study. In: 2013 IEEE Congress on Evolutionary Computation (CEC), 20–23 June 2013, pp 2444–2449 (2013)
26. Pluhacek, M., Senkerik, R., Zelinka, I.: Multiple choice strategy based PSO algorithm with chaotic decision making—a preliminary study. In: Herrero, Á., Baruque, B., Klett, F., et al. (eds.) International Joint Conference SOCO'13-CISIS'13-ICEUTE'13, vol. 239, pp. 21–30. Advances in Intelligent Systems and Computing. Springer International Publishing (2014)
27. ELabbasy, E., Agiza, H., EL-Metwally, H., Elsadany, A.: Bifurcation analysis, chaos and control in the Burgers mapping. Int. J. Nonlinear Sci. **4**(3), 171–185 (2007)
28. Sprott, J.C.: Chaos and Time-Series Analysis. Oxford University Press, Oxford (2003)

MIMO Pseudo Neural Networks for Iris Data Classification

Zuzana Kominkova Oplatkova and Roman Senkerik

Abstract This research deals with a novel approach to classification. This paper deals with a synthesis of a complex structure which serves as a classifier. Compared to previous research, this paper synthesizes multi-input–multi-output (MIMO) classifiers. Classical artificial neural networks (ANN) were an inspiration for this work. The proposed technique creates a relation between inputs and outputs as a whole structure together with numerical values which could be observed as weights in ANN. The Analytic Programming (AP) was utilized as the tool of synthesis by means of the evolutionary symbolic regression. Iris data (a known benchmark for classifiers) was used for testing of the proposed method. For experimentation, Differential Evolution for the main procedure and also for meta-evolution version of analytic programming was used.

Keywords Pseudo neural networks · Symbolic regression · Classification

1 Introduction

This paper deals with a new method for classification problems, which is based on evolutionary symbolic regression. The symbolic regression is able to synthesize a complex structure which is optimized by means of evolutionary computation. Such a structure can be used as a classifier because it can simulate the behavior of the Artificial Neural Networks (ANN) [1–4], where the inspiration for this work came from.

Z. K. Oplatkova (✉) · R. Senkerik
Faculty of Applied Informatics, Tomas Bata University in Zlin, Nam. T. G. Masaryka 5555, 760 01 Zlin, Czech Republic
e-mail: oplatkova@fai.utb.cz

R. Senkerik
e-mail: senkerik@fai.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_15,

Artificial neural networks are based on some relation between inputs and output(s), which utilizes mathematical transfer functions and optimized weights from training process. To optimize of the structure is a time-demanding process and it requires experiences of the user. Setting of a number of layers and nodes in the layers is followed usually by a deterministic training algorithm as Backpropagation or Levenberg-Marquardt [1–4]. There exist also stochastic approaches for training and settings of the artificial neural networks like genetic algorithms and others [5, 6]. On account of this fact, the novelty approach using symbolic regression with evolutionary computation is proposed in this paper. But the approach in this case is different.

Symbolic regression in the context of evolutionary computation means to build a complex formula from basic operators defined by users. The basic case represents a process in which the measured data is fitted and a suitable mathematical formula is obtained in an analytical way. This process is widely known for mathematicians. They use this process when a need arises for mathematical model of unknown data, i.e. relation between input and output values. The symbolic regression can be used also for design of electronic circuits or optimal trajectory for robots and within other applications [7–13]. Everything depends on the user-defined set of operators. The proposed technique is similar to synthesis of analytical form of mathematical model between input and output(s) in training set used in neural networks. Therefore we can call this technique Pseudo Neural Networks. There is no optimization of number of nodes, connections or transfer function in nodes. This technique synthesizes a structure between inputs and output which transfer input values from training set items to output. The training is done by means of optimization procedure in evolutionary symbolic regression on the basis of output error function. The obtained structure is not possible to redraw to a pure ANN structure of nodes and connections.

This paper uses Analytic Programming (AP) [10–13] for evolutionary symbolic regression procedure. Besides AP, other techniques for symbolic regression computation can be found in literature, e.g. Genetic Programming (GP) introduced by Koza [7, 8] or Grammatical Evolution (GE) developed by O'Neill and Ryan [9].

The above-described tools were recently commonly used for synthesis of artificial neural networks but in a different manner than is presented here. One possibility is the usage of evolutionary algorithms for optimization of weights to obtain the ANN training process with a small or no training error result. Some other approaches represent the special ways of encoding the structure of the ANN either into the individuals of evolutionary algorithms or into the tools like Genetic Programming. But all of these methods are still working with the classical terminology and separation of ANN to neurons and their transfer functions [5].

In this paper, iris plant dataset [14, 15] was used as a benchmark case for classification, which has been introduced by Fisher [14] for the first time. It is a well known dataset with 4 features and 3 classes. The attributes consist of sepal length, sepal width, petal length and petal width, which divides the plants into *Iris virginica*, *Iris versicolor* and *Iris sentosa*. The data set was analyzed in a lot of

papers by means of supervised and unsupervised neural networks [16–19], variations like distribution based ANN [20], piecewise linear classifier [21] or rough sets [22]. Not only pure ANN were used for classification but also evolutionary algorithms connected with fuzzy theory were employed [23, 24]. The tool from symbolic regression called Gene expression programming was used for classification too [25]. The last mentioned tool was used as a classifier, which contain procedures if-then rules in the evolutionary process. The basic components consist of greater then, less then, equal to, etc. and pointers to attributes.

The proposed technique in this paper is different. It synthesizes the structure without a prior knowledge of transfer functions and inner potentials. It synthesizes the relation between inputs and output of training set items used in neural networks so that the items of each group are correctly classified according the rules for cost function value.

In previous research, an Analytic Programming (AP) was used only for multi input–single-output (MISO). This paper introduces an approach with more outputs—MIMO approach within usage of AP similarly as the artificial neural networks use. It synthesizes serially expressions with relation between input and each of outputs. Also in the case of artificial neural networks, it can be obtained more expressions between inputs and each of outputs.

Firstly, Analytic Programming used as a symbolic regression tool is described. Subsequently Differential Evolution used for main optimization procedure within Analytic Programming and also as a second algorithm within metaevolution purposes is mentioned. After that a brief description of artificial neural network (ANN) follows. Afterwards, the proposed experiment with differences compared to classical ANN is explained. The result section and conclusion finish the paper.

2 Analytic Programming

This tool was used for the synthesis of a complex structure which can behave similarly to supervised ANN and classify items from the training set into specified groups. Basic principles of the AP were developed in 2001 [10–13]. Until that time only genetic programming (GP) [7, 8] and grammatical evolution (GE) [9] had existed. GP uses genetic algorithms while AP can be used with any evolutionary algorithm, independently on individual representation.

The core of AP is based on a special set of mathematical objects and operations. The set of mathematical objects is set of functions, operators and so-called terminals (as well as in GP), which are usually constants or independent variables. This set of variables is usually mixed together and consists of functions with different number of arguments. Because of a variability of the content of this set, it is called here "general functional set"—GFS. The structure of GFS is created by subsets of functions according to the number of their arguments. For example GFS_{all} is a set of all functions, operators and terminals, GFS_{3arg} is a subset containing functions with only three arguments, GFS_{0arg} represents only terminals,

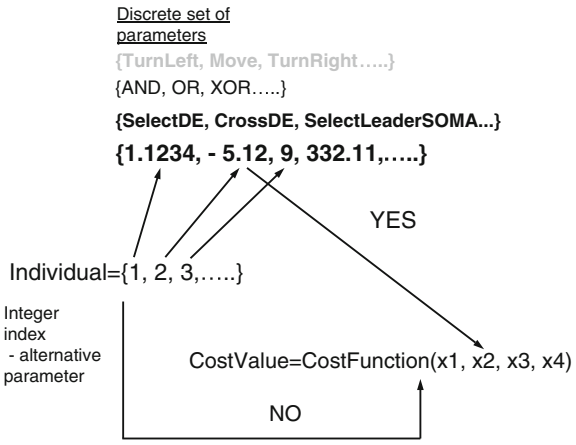


Fig. 1 Discrete set handling

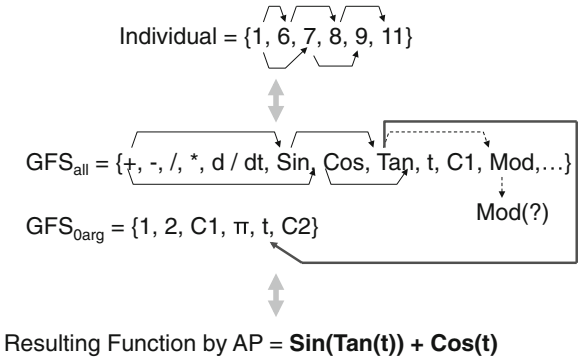


Fig. 2 Main principles of AP

etc. The subset structure presence in GFS is vitally important for AP. It is used to avoid synthesis of pathological programs, i.e. programs containing functions without arguments, etc. The content of GFS is dependent only on the user. Various functions and terminals can be mixed together [10].

The second part of the AP core is a sequence of mathematical operations, which are used for the program synthesis. These operations are used to transform an individual of a population into a suitable program. Mathematically stated, it is a mapping from an individual domain into a program domain. This mapping consists of two main parts. The first part is called discrete set handling (DSH) [10–13] and the second one stands for security procedures which do not allow synthesizing pathological programs. The method of DSH, when used, allows handling arbitrary objects including nonnumeric objects like linguistic terms {hot, cold, dark…}, logic terms (True, False) or other user defined functions. In the AP DSH is used to map an individual into GFS and together with security procedures creates the above mentioned mapping which transforms arbitrary individual into a program (Figs. 1 and 2).

AP needs some evolutionary algorithm that consists of population of individuals for its run. Individuals in the population consist of integer parameters, i.e. an individual is an integer index pointing into GFS. The individual contains numbers which are indices into GFS. The detailed description is represented in [10, 11].

AP exists in 3 versions—basic without constant estimation, AP_{nf}—estimation by means of nonlinear fitting package in Mathematica environment and AP_{meta}—constant estimation by means of another evolutionary algorithms; meta means meta-evolution.

3 Used Evolutionary Algorithms

This research used one evolutionary algorithm: Differential Evolution (DE) [26, 27] for main process and also meta-evolutionary process in AP.

DE is a population-based optimization method that works on real-number-coded individuals [26]. DE is quite robust, fast, and effective, with global optimization ability. It does not require the objective function to be differentiable, and it works well even with noisy and time-dependent objective functions.

For each individual $\vec{x}_{i,G}$ in the current generation G, DE generates a new trial individual $\vec{x}'_{i,G}$ by adding the weighted difference between two randomly selected individuals $\vec{x}_{r1,G}$ and $\vec{x}_{r2,G}$ to a randomly selected third individual $\vec{x}_{r3,G}$. The resulting individual $\vec{x}'_{i,G}$ is crossed-over with the original individual $\vec{x}_{i,G}$. The fitness of the resulting individual, referred to as a perturbed vector $\vec{u}_{i,G+1}$, is then compared with the fitness of $\vec{x}_{i,G}$. If the fitness of $\vec{u}_{i,G+1}$ is greater than the fitness of $\vec{x}_{i,G}$, then $\vec{x}_{i,G}$ is replaced with $\vec{u}_{i,G+1}$; otherwise, $\vec{x}_{i,G}$ remains in the population as $\vec{x}_{i,G+1}$. Description of used DER and 1Bin strategy is presented in (1). Please refer to [26, 27] for the description of all other strategies.

$$u_{i,G+1} = x_{r1,G} + F \cdot \left(x_{r2,G} - x_{r3,G}\right) \tag{1}$$

4 Artificial Neural Networks

Artificial neural networks are inspired in the biological neural nets and are used for complex and difficult tasks [1–4, 28]. The most often usage is classification of objects as also in this case. ANNs are capable of generalization and hence the classification is natural for them. Some other possibilities are in pattern recognition, control, filtering of signals and also data approximation and others.

There are several kinds of ANN. Simulations were based on similarity with feedforward net with supervision. ANN needs a training set of known solutions to be learned on them. Supervised ANN has to have input and also required output.

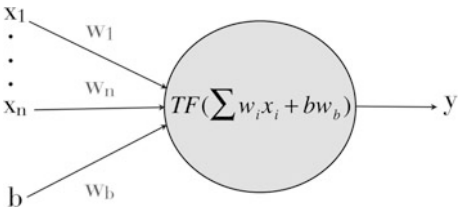


Fig. 3 Neuron model, where *TF* transfer function like sigmoid, x_1–x_n inputs to neural network, b bias (usually equal to 1), w_1–w_n, w_b weights, y output

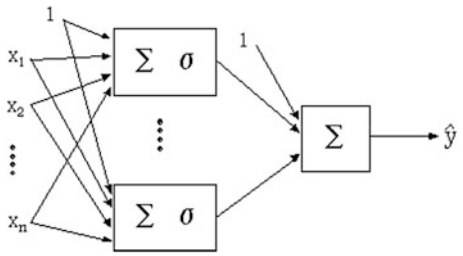


Fig. 4 ANN models with one hidden layer

The neural network works so that suitable inputs in numbers have to be given on the input vector. These inputs are multiplied by weights which are adjusted during the training. In the neuron the sum of inputs multiplied by weights are transferred through mathematical function like sigmoid, linear, hyperbolic tangent etc. Therefore ANN can be used for data approximation [2]—a regression model on measured data, relation between input and required (measured data) output.

These single neuron units (Fig. 3) are connected to different structures to obtain different structures of ANN (e.g. Fig. 4), where $\sum \delta = TF[\sum (w_i x_i + bw_b)]$ and $\sum = TF[\sum (w_i x_i + bw_b)]$; TF is for example logistic sigmoid function.

The example of relation between inputs and output can be shown as a mathematical form (2). It represents the case of only one neuron and logistic sigmoid function as a transfer function.

$$y = \frac{1}{1 + e^{-(x_1 w_1 + x_2 w_2)}}, \tag{2}$$

where

y	Output
x_1, x_2	Inputs
w_1, w_2	Weights.

The aim of the proposed technique is to find similar relation to (2). This relation is completely synthesized by evolutionary symbolic regression—Analytic Programming.

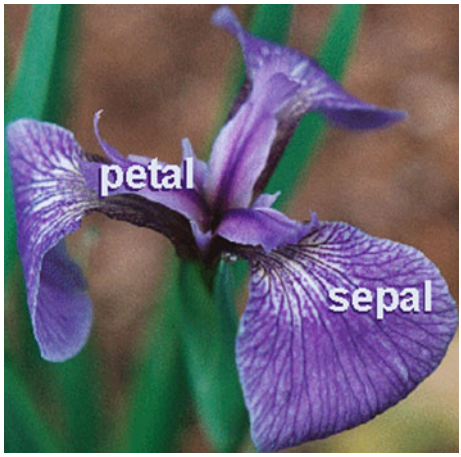


Fig. 5 Iris—petal and sepal

5 Problem and Iris Dataset Description

For this classification problem, iris plant dataset [14, 15] was used as a benchmark case for classification. It is a well known dataset with 4 features and 3 classes. The attributes consist of sepal length, sepal width, petal length and petal width (Fig. 5), which divides the plants into *Iris virginica*, *Iris versicolor* and *Iris sentosa*. This set contains 150 instances. Half amount was used as training data and the second half was used as testing data. The cross validation is planned for future testing. The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant. One class is linearly separable from the other 2; the latter are NOT linearly separable from each other. The attributes were of real values.

Compared to previous research, this paper uses MIMO approach. An expression for each output node was synthesized. The data for each simulation was used as follow: first node—data for *Iris setosa* as value higher than 0.5 which was saturated to 1 and the other two groups as data with saturated zero value. The second node expression divided similarly data for iris versicolor as saturated value 1 and the other two as value zero. The last node was the same for *Iris virginica*. The final combination of the output node values give the same result as in ANN approach.

The cost function value is given in Eq. (3), i.e. if the cv is equal to zero, all n training patterns are classified correctly. The same cost function was performed for all node expressions.

$$cv = \sum_{i=1}^{n} |required\ Output - current\ Output| \tag{3}$$

Table 1 DE settings for main process of AP

PopSize	40
F	0.8
Cr	0.8
Generations	50
Max. CFE	2,000

Table 2 DE settings for meta-evolution

PopSize	40
F	0.8
Cr	0.8
Generations	150
Max. CFE	6,000

6 Results

AP_{meta} version was carried out in simulations. As described above, AP needs an evolutionary algorithm for its run. Here Differential evolution was used for the main process and also meta-evolutionary process. Meta-evolutionary approach means usage of one main evolutionary algorithm for AP process and second algorithm for coefficient estimation, thus to find optimal values of constants in the structure of pseudo neural networks.

Settings of EA parameters for both processes were based on performed numerous experiments with chaotic systems and simulations with AP_{meta} (Tables 1 and 2), where CFE means cost function evaluations.

The set of elementary functions for AP was inspired in the items used in classical artificial neural nets. The components of input vector x contain values of each attribute (x_1, x_2, x_3, x_4). Thus AP dataset consists only of simple functions with two arguments and functions with zero arguments, i.e. terminals. Functions with one argument, e.g. Sin, Cos, etc., were not applied in the case of this paper.

Basic set of elementary functions for AP:

$GFS_{2arg} = +, -, /, *, \hat{}, \exp$
$GFS_{0arg} = x_1, x_2, x_3, x_4, K$

Total number of cost function evaluations for AP was 1,000, for the second EA it was 6,000, together 6 millions per each simulation.

From carried simulations, several notations of input dependency were obtained. The advantage is that equations for each output node can be mixed. Therefore the best shapes for each output node can be selected and used together for correct behaviour. Equations (4–6) are just examples of the found synthesized expressions. The training error for the example is equal to 2 misclassified items (it means

2.66 % error, 97.34 % success from all 75 training patterns). One mistake was in the second group and one in the last group. As said above, the expression can be mixed. The suitable node outputs should be selected also on the basis of the testing error. In some cases the error was equal to 7, the proposed output here has the error equal to 5 misclassified items (8 % error, 92 % success). The expressions are too long therefore the typesetting is not in a standard form.

$$y_1 = \frac{\exp\left(-138.195 - \mathrm{x}_2 + \frac{698.64}{x_4}\left(-1.123695526343 \times 10^{-387} + \frac{1}{\left(\frac{x_1}{x_2}\right)^{1012.09}}\right)\right)}{-291.996\mathrm{x}_3 + \mathrm{x}_4} \tag{4}$$

$$y_2 = -454.721 - \frac{\mathrm{x}_2}{\left(-2.61862159472 \times 10^{431}\right)^{19.1899\mathrm{x}_1}\exp\left(-6.08872 \times 10^{79^{\mathrm{x}_4}} - (247.913 - x_1)^{0.0096173\mathrm{x}_4}\right)} + \exp\left((-764.718)^{904.124-\mathrm{x}_4}(247.913 - x_1)^{0.0096173\mathrm{x}_4}\left(425.262 + \mathrm{x}_2^{\mathrm{x}_3}\right)\right) \tag{5}$$

$$y_3 = -135.338 + \exp(x_3) + x_4 \tag{6}$$

The obtained training and testing errors are comparable with errors obtained within other approaches as artificial neural networks and others.

The following output serves as an example of ANN classification for comparison with AP approach. ANN was more successful during the training. The training error was equal to zero. The testing error was equal to 7 when 4 hidden nodes were used with sigmoid function in hidden nodes and also in the output. The toolbox of Neural Networks for Mathematica environment was used. The output was prepared the same as in AP case. Figure 6 shows root mean square error during training of ANN.

Examples of the input dependency obtained from ANN are depicted in (7–9).

$$y_1 = \frac{1}{1 + \exp\left(\begin{array}{l} 2.1349 + \frac{20.3401}{1 + \exp(0.617561 + 0.724599\mathrm{x}_1 + 2.80507\mathrm{x}_2 - 4.37061\mathrm{x}_3 - 1.07415\mathrm{x}_4)} \\ + \frac{0.114025}{1 + \exp(-0.00358697 + 0.696284\mathrm{x}_1 + 0.963376\mathrm{x}_2 + 0.558429\mathrm{x}_3 + 0.45666\mathrm{x}_4)} \\ + \frac{21.2425}{1 + \exp(-0.0444825 - 1.23679\mathrm{x}_1 - 2.91472\mathrm{x}_2 + 5.16414\mathrm{x}_3 + 1.74949\mathrm{x}_4)} \\ + \frac{1.67864}{1 + \exp(-24.3389 - 7.31299\mathrm{x}_1 - 15.8329\mathrm{x}_2 + 10.8511\mathrm{x}_3 + 35.0774\mathrm{x}_4)} \end{array}\right)} \tag{7}$$

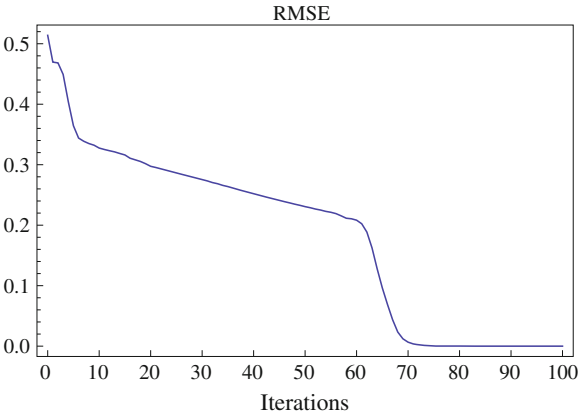


Fig. 6 RMSE of sigmoid transfer functions used in hidden and output nodes

$$y_2 = \frac{1}{1+\exp\left(\begin{array}{l} 26.7957 + \dfrac{5.83714}{1+\exp(0.617561 + 0.724599x_1 + 2.80507x_2 - 4.37061x_3 - 1.07415x_4)} \\ \quad + \dfrac{0.776899}{1+\exp(-0.00358697 + 0.696284x_1 + 0.963376x_2 + 0.558429x_3 + 0.45666x_4)} \\ \quad + \dfrac{35.6293}{1+\exp(-0.0444825 - 1.23679x_1 - 2.91472x_2 + 5.16414x_3 + 1.74949x_4)} \\ \quad + \dfrac{41.8935}{1+\exp(-24.3389 - 7.31299x_1 - 15.8329x_2 + 10.8511x_3 + 35.0774x_4)} \end{array}\right)} \quad (8)$$

$$y_3 = \frac{1}{1+\exp\left(\begin{array}{l} -9.71972 + \dfrac{11.4094}{1+\exp(0.617561 + 0.724599x_1 + 2.80507x_2 - 4.37061x_3 - 1.07415x_4)} \\ \quad + \dfrac{0.740345}{1+\exp(-0.00358697 + 0.696284x_1 + 0.963376x_2 + 0.558429x_3 + 0.45666x_4)} \\ \quad + \dfrac{2.18052}{1+\exp(-0.0444825 - 1.23679x_1 - 2.91472x_2 + 5.16414x_3 + 1.74949x_4)} \\ \quad + \dfrac{42.3753}{1+\exp(-24.3389 - 7.31299x_1 - 15.8329x_2 + 10.8511x_3 + 35.0774x_4)} \end{array}\right)} \quad (9)$$

where

y_1, y_2, y_3	Outputs from each output node
x_1, x_2, x_3, x_4	Inputs (attributes of iris flowers).

Both results (4–6) and (7–9) work as a transfer of inputs to output, which classify iris flower into three groups. The ANN were more successful during training phase when training error (RMSE) was 10^{-8} to 10^{-17} in all performed simulations. The AP technique was less successful. The final solution was synthesized on the RMSE over a whole training set not one by one as it is in ANN. This is one reason why AP technique takes longer time. The complexity and number of function members is higher in the case of ANN for the same quality result.

7 Conclusion

This paper deals with a novel approach—pseudo neural networks. Within this approach, classical optimization of the structure or weights was not performed. The proposed technique is based on symbolic regression with evolutionary computation. It synthesizes a whole structure in symbolic form without a prior knowledge of the ANN structure or transfer functions. It means that the relation between inputs and output(s) is synthesized. In the case of this paper, expressions for each output node were synthesized separately and independently. It means that structures can be mixed according the best shape and the combinations from outputs serve similarly as in ANN. As can be seen from the result section, such approach is promising. The complexity of structure from AP is less than in the case of ANN. On the other hand, it is not possible to draw the pure scheme of the network for the pseudo neural networks. For further tests some observed critical points have to be taken into consideration. Future plans will be focused on further tests and more complicated cases with more attributions.

Acknowledgments This work was supported by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089.

References

1. Gurney, K.: An Introduction to Neural Networks. CRC Press, Boca Raton (1997). ISBN 1857285034
2. Hertz, J., Kogh, A., Palmer, R.G.: Introduction to the Theory of Neural Computation. Addison-Wesley, Reading (1991)
3. Wasserman, P.D.: Neural Computing: Theory and Practice. Coriolis Group (1980). ISBN 0442207433
4. Fausett, L.V.: Fundamentals of Neural Networks: Architectures, Algorithms and Applications. Prentice Hall, Englewood Cliffs (1993). ISBN 9780133341867
5. Fekiac, J., Zelinka, I., Burguillo, J.C.: A review of methods for encoding neural network topologies in evolutionary computation. In: ECMS 2011, Krakow, Poland. ISBN 978-0-9564944-3-6
6. Back, T., Fogel, D.B., Michalewicz, Z.: Handbook of evolutionary algorithms. Oxford University Press, Oxford (1997). ISBN 0750303921
7. Koza, J.R., et al.: Genetic Programming III; Darwinian Invention and Problem Solving. Morgan Kaufmann Publisher, Los Altos (1999). ISBN 1-55860-543-6
8. Koza, J.R.: Genetic Programming. MIT Press, Cambridge (1998). ISBN 0-262-11189-6
9. O'Neill, M., Ryan, C.: Grammatical Evolution. Evolutionary Automatic Programming in an Arbitrary Language. Kluwer Academic Publishers, Dordrecht (2003). ISBN 1402074441
10. Zelinka, I., et al.: Analytical programming—A novel approach for evolutionary synthesis of symbolic structures. In: Kita, E. (ed.) Evolutionary Algorithms, InTech (2011). ISBN 978-953-307-171-8
11. Oplatkova, Z.: Metaevolution: Synthesis of Optimization Algorithms by means of Symbolic Regression and Evolutionary Algorithms. Lambert Academic Publishing, Saarbrücken (2009). ISBN 978-3-8383-1808-0

12. Zelinka, I., Varacha, P., Oplatkova, Z.: Evolutionary synthesis of neural network. In: 12th International Conference on Softcomputing (Mendel 2006), Brno, Czech Republic, 31 May–2 June 2006, pp. 25–31. ISBN 80-214-3195-4
13. Zelinka, I., Oplatkova, Z., Nolle, L.: Boolean symmetry function synthesis by means of arbitrary evolutionary algorithms-comparative study. Int. J. Simul. Syst. Sci. Technol. **6**(9), 44–56 (2005). ISSN 1473-8031
14. Fisher, R.A.: The use of multiple measurements in taxonomic problems. Ann. Eugenics **7**(2), 179–188 (1936). doi:10.1111/j.1469-1809.1936.tb02137.x
15. Machine learning repository with Iris data set, http://archive.ics.uci.edu/ml/datasets/Iris
16. Swain, M., et al.: An approach for iris plant classification using neural network. Int. J. Soft Comput. **3**(1) (2012). doi:10.5121/ijsc.2012.3107
17. Shekhawat, P., Dhande, S.S.: Building and iris plant data classifier using neural network associative classification. Int. J. Adv. Technol. **2**(4), 491–506 (2011). ISSN 0976-4860
18. Avci, M., Yildirim, T.: Microcontroller based neural network realization and iris plant classifier application. In: Proceedings of the Twelfth Turkish Symposium on Artificial Intelligence and Neural Networks (TAINN'03), Canakkale, Turkey, 2–4 July 2003
19. Osselaer, S.: Iris data analysis using back propagation neural networks. J. Manuf. Syst. **13**(4), 262 (2003)
20. Chen, S., Fang, Y.: A new approach for handling iris data classification problem. Int. J. Appl. Sci. Eng. (2005). ISSN 1727-2394
21. Kostin, A.: A simple and fast multi-class piecewise linear pattern classifier. Pattern Recogn. **39**(11), 1949–1962 (2006). ISSN 0031-3203. doi:10.1016/j.patcog.2006.04.022
22. Kim, D.: Data classification based on tolerant rough set. Pattern Recogn. **34**(8), 1613–1624 (2001). ISSN 0031-3203. doi:10.1016/S0031-3203(00)00057-1
23. Agustín-Blas, L.E., et al.: A new grouping genetic algorithm for clustering problems. Expert Syst. Appl. **39**(10), 9695–9703 (2002). ISSN 0957-4174. doi:10.1016/j.eswa.2012.02.149
24. Zhou, E., Khotanzad, A.: Fuzzy classifier design using genetic algorithms. Pattern Recogn. **40**(12), 3401–3414 (2007). ISSN 0031-3203. doi:10.1016/j.patcog.2007.03.028
25. Ferreira, C.: Gene Expression Programming: Mathematical Modelling by an Artificial Intelligence (2006). ISBN 9729589054
26. Lampinen, J., Zelinka, I.: New Ideas in Optimization—Mechanical Engineering Design Optimization by Differential Evolution, vol. 1, 20 p. McGraw-hill, London (1999). ISBN 007-709506-5
27. Price, K., Storn, R.M., Lampinen, J.A.: Differential Evolution: A Practical Approach to Global Optimization, 1st edn. Natural Computing Series. Springer, Berlin (2005)
28. Volna, E., Kotyrba, M., Jarusek, R.: Multiclassifier based on Elliott wave's recognition. Comput. Math. Appl. **66** (2013). ISSN 0898-1221. doi:10.1016/j.camwa.2013.01.012
29. Oplatkova, Z., Senkerik, R.: Evolutionary synthesis of complex structures—Pseudo neural networks for the task of iris dataset classification. In: Zelinka, I., Chen, G., Rössler, O.E., Snasel, V., Abraham, A. (eds.) Nostradamus 2013: Prediction, Modeling and Analysis of Complex Systems, vol. 210, pp. 211–220. Advances in Intelligent Systems and Computing. Springer International Publishing, Switzerland (2013). doi:10.1007/978-3-319-00542-3_22

Part II
Computer Science

Compliance Management Model for Interoperability Faults Towards Governance Enhancement Technology

Kanchana Natarajan and Sarala Subramani

Abstract The objective of the research is to propose a software compliance management model for interoperability faults of regulatory non-compliances in IT industries. The enterprise software is exercised to minimize the risks on different types of non-compliances. The framework activities and procedures are kept in adherence to the guidelines and regulatory laws of the information related business or industries. The entities that are non-adherence to the standards and failed to follow the enumerated regulations are analyzed for the non-compliances. The non-compliances in procedure-oriented processes and coding are mapped with the risks associated with severity and impact on the chosen applications. The interoperability fault is tolerated by the customized rules based on criticality of the applications. The conformance to the requirement specifications pertaining to process, people, product and its quality are verified as a distributed system to manage the non-compliances. The existing information governance can be improvised by the proposed GET technique.

Keywords Business risks · COBIT · Compliance · Risk · Interoperability fault

1 Introduction

The Governance Risk and Compliance (GRC) management is a holistic approach focuses on the systematic decision making process in order to meet the issues and challenges with enterprise IT governance. The research focuses on the

K. Natarajan (✉) · S. Subramani
Department of Information Technology, Bharathiar University, Coimbatore, India
e-mail: kanchananatarajan@live.com

S. Subramani
e-mail: sarala.bu@gmail.com

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_16,

interoperability issues that may occur at various levels of processes and resources of the enterprise software. To identify the exact interoperability faults in any software the development and management teams have to be trained with the existing standards and compliances across the domain of interest. The various issues and challenges in the governance of non-compliances can solve through the structured management technique called Regulatory Compliance Management (RCM). It ensures that the data, processes and organization are structured in accordance with the regulations of the guidelines which are specified in the regulations [1].

Software compliance can be defined as a state of conformance to the standards based on requirements of the respective domain. The scope of compliance become increasingly complex due to the large number of regulations and standards are introduced by the local or global policy makers. The documentation and maintenance agreement comes under regulatory activities which are vulnerable if not having met the enforcement acts [2]. The three different disciplines like technical wing, legal wing and administration wing have to coordinate to procure any software related products such as intellectual properties to satisfy the customers. To identify the exact interoperability faults in any software the development and management teams have to be trained with the existing standards and compliances across the domain of interest. The cost of non-compliances is more expensive which can be reduced when the organization initially spends a higher proportion of IT budgets on compliance activities especially on the factors of global privacy, regulatory constraints and legal obligations [3].

The necessity of the process model should have a control to monitor the compliance constraints through reviews and audits to avoid the risk of non-compliance and financial penalty's [4]. The non-compliance may also lead to risks of legal sanctions or customer trust loss due to inadequate services of the software product. The issues may evolve in terms of the failure of compliance features which includes complexity, reusability, understandability and maintainability [5]. One among the quality sub-attributes of ISO/IEC 25010-1 standard for software quality model is interoperability, means that regulations to handle the capability of the software product to interact with one or more specified systems. The interoperability faults in the software processes may begin in the lower level of implementation were each and every quality feature has to be ascertained in all possible combinations to assure the expected system behavior [6]. Especially to fulfill the control objectives it is needed to have a correct and timely composition of services which depends on the quality features and associated quality attributes [7, 8]. The analysis of non-observed measurement factors through best practices may solve the issues of non-compliances [9]. The corporate fraud includes 2G Scam, WorldCom and Enron [13] are based on auditing applications due to the failed applicable regulations and accountability acts. The best practices like COBIT, HIPAA [1, 2, 12], SOX [2, 4, 12], PCI DSS [3] and BASEL III [5] may ensures the compliance of IT sectors, health sectors, banking sectors and so on by ensuring do the information security standards are focused [10].

2 Related Works

The non-compliant indicators do not depend on the software development method but are related to the human resources strategy and project management strategy on the organizational level [11]. The problem of achieving interoperability is closely related to standards of the applicable domain. Existing research on interoperability models ensures the compatibility and integration level of large scale systems through Capability Maturity Model Integration (CMMI), Government Interoperability Maturity Matrix (GIMM) and Business Interoperability Quotient Measurement Model (BIQMM) [12]. The lack of applicable domain-specific models for small scale systems arise several non-compliances thereby the research inter-relates interoperability and compliance in the context of standards of the domain. There are many solutions for the problem of modeling and checking compliance, as well as violation recovery through meta-model by defining syntax, semantics and notations. The scale and diversity of compliance requirements are changing with respect to many features like application criticality, deployment platform, modes of control and its selectivity of domain specific problems which may change more frequently. Such large and complex problems necessitate a formal representation of control objectives in Formal Contract Language (FCL) or Process Compliance Language (PCL) [13], the languages which are suitable to capture the declarative nature of compliance requirements [14]. The commitment, privilege and right analysis [15] and its effectiveness of requirement engineers involved in the extraction of compliance requirements from privacy policy which results much better in the view of correctness and completeness. The conceptual approach for the regulatory compliance issue [16] has the combination of an organization's business process management on the one hand and a respective accompanying meta-model covering risk and control mechanisms for achieving compliance on the other. The regulatory compliance framework [17] have been integrated with set of software requirements and regulations as input to identify the irregularities there by associating argumentation tree structure in order to capture the arguments to ensure its acceptability. Hence the framework is subjective used only under certain circumstance which shows the evidence for framework's inadequacy. The limitation of this model implies the coverage and failure of pattern to specify compliance requirements at certain instances may increase the compliance risk [18].

3 Software Regulatory Compliance and Management Flow

The Indian IT industries need to focus on control objectives that are directly or indirectly proportional towards interoperability between various components deployed over physical or virtual servers of the enterprise software. The work focuses on the software compliance and its governance in the processes to

minimize the risks using compliance verification as per the currently existing standards of the domain of application and to report the non-compliance of the submitted software to the business client through application interface. The compliance adheres to standards, conventions or regulations in laws and similar prescriptions. Hence a compliance requirement is a constraint or assertion that prescribes a desired result to be achieved by factoring control procedures in processes. It can be prescribed in the form of abstract constraints or control objectives.

The software regulatory compliance and its management are more critical in the case of sensitive information related technologies and more specifically across many geographical boundaries of many nations. Under these conditions, the people, processes and resources are to be kept under strong vigil so as to maintain the business productivity and organization reputation. Such a container of a policy framework to handle information technology related issues and risks are enumerated in the COBIT Framework. The Control Objectives for Information and Related Technology (COBIT) is a large-scale business framework for enterprise IT which provides a set of principles, maturity models and best practices for maximizing the potential benefits and minimizing the business risks and at the same time to accomplish essential criteria's such as effectiveness, efficiency, confidentiality, integrity, availability, compliance and reliability. COBIT is accepted and recognized by the Information Technology Departments in India as the standard for IT governance representing its national e-governance plan. The globally recognized COBIT framework provides a set of guidelines, regulations and modules which are classified accordingly in order to enhance compliance and auditability for better governance. For example, the COBIT modules are grouped such as business focused, process oriented, control based and measurement driven which defines several regulations of the software processes which is shown in the Table 1.

The software regulatory bodies define and declare regulatory acts and laws in order to ensure the process and quality compliance of the application programs. The work group has to follow the business processes with appropriate standards to make sure the people and products are governed and all activities are followed legally throughout the developmental phases. The audit and review processes are performed outside the workspace by the third-party checkers modules.

The non-compliances in suspected code after review process ensures failure of respective standards and its enforcement and governance activities of working group of the organization improves the quality of the legacy code in the application program. The exceptions and errors are reported by the risk reporter component if non-compliances were found. The planning and organization phase determines the scheduling and estimation of the work needed to certify the program as mentioned in the workflow shown in Fig. 1.

Table 1 COBIT frameworks and modules

COBIT framework Regulations (R1–R3)	COBIT modules			
	Business focused	Process oriented	Controls based	Measurement driven
R1	Communicate management aims and direction (PO6)	Acquire and maintain application software (AI2)	Manage third-party services (DS2)	Monitor and evaluate internal control (M2)
R2	Ensure compliance with external requirements (PO8)	Develop and maintain procedures (AI4)	Ensure Systems Security (DS5)	Ensure compliance (M3)
R3	Assess risks (PO9)	Define and manage service levels (DS1)	Monitor and evaluate IT performance (M1)	Provide for independent audit (M4)

4 Computational Risk and Compliance Management Model for Interoperability Faults

The non-availability of safety critical applications and the low maintainability of the industrial process control software will increase the probability of major risks in the production and human loss. Risk exposure element determines the priority and organization of quality requirements in all commercial software. The focus of all these analysis and study is to minimize the impact of risks at all levels. The proposed computational model focuses the detailed association of the control requirements (CR) with all technical issues (TI) which leads to different forms of business risks. Commonly, risk can be expressed as $R = P \times I$, where R is the project risk exposure, P is the probability of the risk factor's occurrence, and I is the impact of the risk factor. Generally risk can be quantitatively expressed as shown in the equation form (1) and business risk in the equation form (2):

$$\text{Risk} = \sum_{i=1}^{n} \text{Prob}(\text{Non} - \text{Compliance}) * [1 - \text{Prob}\ (\text{Defect in Resources} * \text{Hazard level in Processes})] \quad (1)$$

$$\text{Business Risk} = \text{Total Instance of Non} - \text{Compliance} * [1 - \text{Prob}(\text{IOF}(\text{Resources})) \cdot (\text{IOF}(\text{Processes})) * \text{Prob}(\text{Technical Issues})]. \quad (2)$$

The number of instances or occurrences of non-compliance in that domain are due to the control requirements in the processes. The interoperability faults are the hazards in the processes and in the resources and they become the defects in the

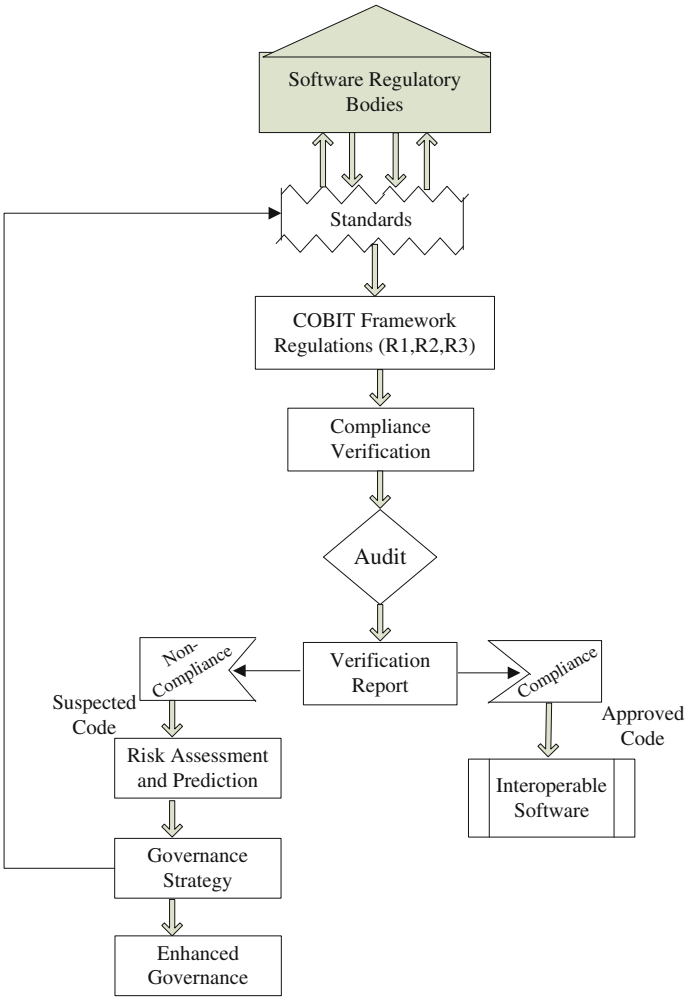


Fig. 1 Software regulatory compliance management flow

case of people and devices depend on the faults and their impact levels. The Fig. 2 indicates the technical issues that are being transformed as control requirements due to all these types of interoperability faults along with the variations in the policies and business guidelines or standards. The compliance management is distributed across all the domains, processes and resources along with the policies. The individual player in the overall governance through compliance management can be illustrated through mathematical relationships and their integration across the framework. The proposed approach, the GET is considered as a net list of functions and their sub functions through functional programming. Instantly, it is quite an evidence in declaring the governance of enhanced technology, let GET_{COBIT} as a

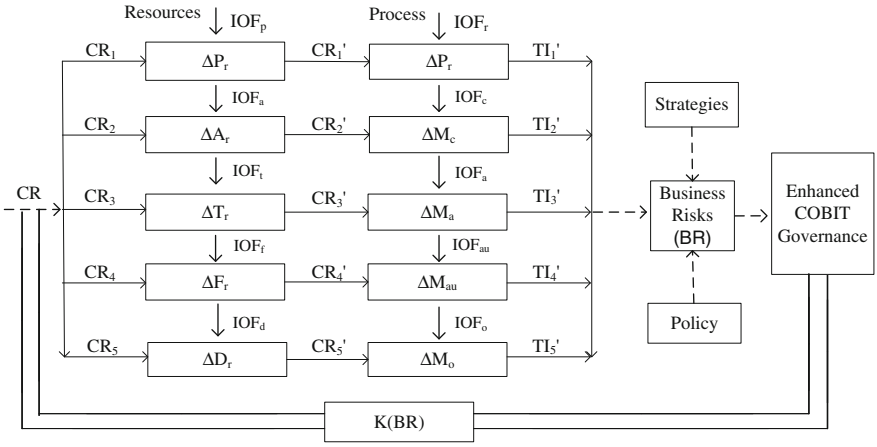


Fig. 2 Flow diagram of business clocked feedback control for enhanced governance

domain of number of processes which needs different information about a variety of resources can be written as *Domain(Process(Information(Resources)))*.

The early detection of non-compliances like the interoperability fault is considered as an requirement fault. It is possible only with the above enhanced model where the governance can be calculated in terms of the integrated value of the ratio of control requirement factors encompassing all resources to the total technical issues towards possible interoperability features and business risks. The quantitative analytical representation can be given as in the equation form of (3) and (4):

$$\mathrm{GET_{COBIT}} = \sum_{i,j}^{m,n} \frac{\mathrm{CR_i(P + A + T + F + D)}}{\mathrm{TI_j(P_r + M_c + M_a + M_{au} + M_o)}} \left(\mathrm{IOF_i + BR_j}\right) \tag{3}$$

$$\mathrm{GET} = \mathrm{CR} - (\mathrm{K} * \mathrm{BR}) = \sum \frac{\mathrm{CR1}' - \mathrm{CR1}}{\mathrm{I0r}} * \Delta\mathrm{R} + \frac{\mathrm{TI1}' - \mathrm{CR1}'}{\mathrm{I0p}} * \Delta\mathrm{P}. \tag{4}$$

Interoperability (IO) Faults are in the processes and resources which are essential to achieve the governance over the framework. The quantification of these faults depends on the individual process handling and resource utilization. The technical issues and business risks are arising due to improper handling of the above mentioned processes and their respective resources.

5 Experimental Results

In the software perspective, the risk control objectives are to be identified and its much needed associations are to be permitted. The proposed distributed compliance management architecture within the COBIT framework determines the

Table 2 Experimental values of governance enhancement technology

CR′	CR1	IOp	ΔR	TI1′	CR1′	IOr	ΔP	GET
0	0	0.1	0	0	0	0.6	0	0
0.25	0.2	0.3	0.2	0.25	0.25	0.2	0.3	0.03
0.5	0.4	0.4	0.3	0.5	0.5	0.1	0.4	0.15
0.75	0.6	0.2	0.6	0.75	0.75	0.4	0.7	0.67
1	1	0.6	0.8	1	1	0.5	0.9	0

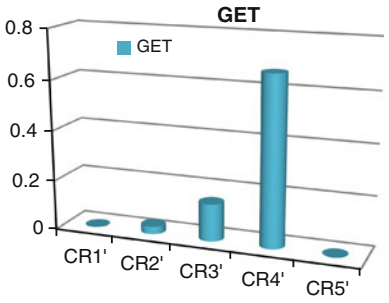


Fig. 3 Governance enhancement technology

possible business risks and their association with the technical issues (TI). The control objectives must be redirected so as to define and declare the set of possible non-compliances found in the resources like either in the processes or in the people. The risk management is made very simple through the compliance management since the possible risks and their reasoning can be done. The impact of such classified risks with their frequency of occurrences is also dealt with. The processes or functions at the business level and the functions at the application and infrastructure levels possibly realize a service or a countermeasure.

The Table 2 shows the experimental values of GET which depicts the interoperability faults involved in the processes and resources. The quantitative perspective makes the multi-set that represents the amount of difference between the actual and the desired or legal parameters, the degree of non compliance or the compliance violation and the amount of risk to be remediated.

There is no governance whereas the interoperability faults with control requirements, technical issues and business risks of resources and processes may varies with policies and guidelines. The triangular issues such as control requirements (CR1′), technical issues (TI1′) and business risks (BR) can be evaluated by keeping the BR as constant. The resources of facilities and processes of audit with respect to CR4′ rises rapidly were the governance improves by reduced business risks. In case of CR1′ and CR5′ there is no governance were the resources of people and processes of review are with business risks as shown in Fig. 3.

6 Conclusion

The distributed software compliance management model focuses the interoperability faults across enterprise software technologies with execution platform is proposed. A mathematical relationship is established between the control objectives of an information technology and associated technical issues occur in the resources and processes, so that the possible business risks can be minimized within the COBIT framework. The existing acts and regulations for the information technology sectors are applied to bring a ubiquitous model where the customer's issues can be reported through different forms of reliable information services using the established, fault-prone mobile and web technologies. The compliance fault or any non-compliance is considered as the logical combination of the respective processes or entities in that phase and the non-utilization of the resources allocated for that process. The workflow model determines the location of the interoperability faults in case of reliable services and generates the audit and review findings. The different types of risks in the IT industries, the frequency of occurrences and the impact also identified and scaled within acceptable limits. The gaps between the possible and realizable objectives are the focus points to minimize the existing regulatory violations and risks by solving the technical issues through proper control strategies.

Acknowledgments The first authors' research is partially funded by the Bharathiar University under University Research Fellowship. The authors would like to thank the unknown reviewers for valuable suggestions that improved the presentation of the paper considerably.

References

1. El Kharbili, M.: Business process regulatory compliance management solution frameworks: a comparative evaluation. In Proceedings of the 8th Asia-Pacific Conference on Conceptual Modeling. Australian Computer Society, vol. 130, pp. 1–10, (2012)
2. Hamdaqa, M., Abdelwahab Hamou-Lhadj.: An approach based on citation analysis to support effective handling of regulatory compliance. J. Future Gener. Comput. Syst. Elsevier Publications, vol. 27, pp. 395–410, (2011)
3. The True Cost of Compliance.: A Benchmark Study of Multinational Organizations. Ponemon Institute, January (2011)
4. Christopher, J., Pavlovski, J.Z.: Non-functional requirements in business process modeling. In Proceedings of the 5th Asia-Pacific Conference on Conceptual Modeling (APCCM), Australian Computer Society, Inc., vol. 79, pp. 103–112, (2008)
5. Tran, H., Zdun, U., Holmes, T., Oberortner, E., Mulo, E., Dustdar, S.: Compliance in service-oriented architectures: A model-driven and view-based approach. J. Inf. Software Technol. Elsevier Publications, vol. 54, pp 531–552, (2012)
6. Mahoney, W., Gandhi, R.A.: An integrated framework for control system simulation and regulatory compliance monitoring. Int. J. Crit. Infrastruct. Prot. Elsevier Publications, vol. 4, pp 41–53, (2011)
7. Kannabiran, G., Sankaran, K.: Determinants of software quality in offshore development: An empirical study of an Indian vendor. J. Inf. Software Technol. Elsevier Publications, vol. 53, pp 1199–1208, (2011)

8. Yuen, K.K.F., Lau, H.C.W.: A fuzzy group analytical hierarchy process approach for software quality assurance management: Fuzzy logarithmic least squares method. J. Expert Syst. Appl. Elsevier Publications, vol. 38, pp. 10292–10302, (2011)
9. Murphy, T., Cormican, K.: An analysis of non-observance of best practice in a software measurement program. Proceedings of the 4th International Conference on ENTERprise Information Systems-aligning technology, organizations and people, Procedia Technology. Elsevier Publications, vol. 5, pp. 50–58, (2012)
10. Saha, P., Mahanti, A., Chakraborty, B.B., Navlani, A.: Development of ontology based framework for information security standards. In Proceedings of the 9th International Conference on Autonomic and Autonomous Systems, pp. 83–89, (2013)
11. Mahnic, V., Zabkar. N,: Assessing scrum-based software development process measurement from COBIT perspective. Proceedings of the 12th WSEAS International Conference on Computers, pp. 589–594, (2008)
12. Zutshi, A., Grilo, A., Jardim-Goncalves R.: The business interoperability quotient measurement model. J. Comput. Ind. Elsevier Publications, **63**(5): 389–404, (2012)
13. Governatori, G., Rotolo. A.: A conceptually rich model of business process compliance. In Proceedings of the 7th Asia-Pacific Conference on Conceptual Modelling (APCCM), vol. 110, pp. 3–12, (2010)
14. Sadiq, S,. Governatori, G., Namiri, K.: Modeling control objectives for business process compliance. In Proceedings of the 5th International Conference on Business Process Management, LNCS, Springer-Verlag, vol. 4714, pp. 149–164, (2007)
15. Schmidt, J.Y., Anton, A.I., Earp, J.B.: Assessing identification of compliance requirements from privacy policies. In Proceedings of the 5th International Workshop on Requirements Engineering and Law (RELAW). pp. 52–61, (2012)
16. Karagiannis, D.: A business process-based modeling extension for regulatory compliance. In Multikonferenz Wirtschaftsinformatik, Munich, pp. 1159–1173, (2008)
17. Ingolfo, S., Siena, A., Mylopoulos, J., Susi, A., Perini, A.: Arguing regulatory compliance of software requirements. J. Data Knowl. Eng. Elsevier Publications, vol. 87, pp. 279–296, (2013)
18. Turetken, O., Elgammal, A., van den Heuvel, W.J., Papazoglou, M.: Enforcing compliance on business processes through the use of patterns. Proceedings of the 19th European Conference on Information Systems, pp. 1–13, (2011)

Reducing Systems Implementation Failure: A conceptual Framework for the Improvement of Financial Systems Implementations within the Financial Services Industries

Derek Hubbard and Raul Valverde

Abstract The financial industry continues to change, become more global, complex and important to economies all around the work. The industry continues to be in flux and the world financial crisis has resulted in changes that have changed the industry for good. The need for agile, accurate and detailed financial systems has never been so important. This research discusses the issues associated with implementing financial systems within financial services companies, a conceptual framework has been built that will help reduce the risk of implementation failure in future financial systems implementations. Financial experts can use the framework to reduce system implementation risk; help deliver projects on time to budget whilst meet the functionality requirements of stakeholders.

Keywords Financial information systems · Risk management · Implementation failure · Risk identification

1 Introduction

There are many challenges faced by finance staff implementing systems within financial service firms. Some of these challenges are listed below:

- System failures cause serious issues for finance departments and can be very costly.

D. Hubbard
UBS, Investment Banking, Singapore, Singapore
e-mail: Dereck.hubbard2@binternet.com

R. Valverde (✉)
John Molson School of Business, Concordia University, Montreal, Canada
e-mail: rvalverde@jmsb.concordia.ca

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_17,

- Finance staff is chosen to be involved in systems implementations due to their functional finance expertise and not according to their skill set to implement systems effectively.
- Finance systems within financial services tend to be specialized and need extensive input and involvement from financial experts to ensure the system works, this is not always the case so increases the implementation risks.
- Simon [1] states that 49 % of implementations have budget overruns, 47 % of implementations have higher maintenance costs and 41 % fail to deliver the expected business value or return on investment.

The research study has the objective of creating a framework for reducing the risk of failure of the implementation of financial systems.

2 Literature Review

A strong financial services industry is an important factor in ensuring that the economies of the world function efficiently. "Financial systems facilitate the transfer of economic resources from one section of the economy to another" [2]. Over recent years we have seen a financial crisis that rocked the world's economies and saw the collapse of some of the industries largest players. Lehman Brothers collapse in 2008 sent shockwaves through the global financial systems industry. We saw emergency consolidations, huge government interventions and nationalization of some banks. The current situation regarding the European banking system is not stable. The financial trilemma indicates that the three objectives of financial stability, cross-border banking and national financial supervision are not compatible [3].

Over recent times, the deregulation of 'financial regulation' coupled with the transforming use of information technology transformed the business models banks used by banks. Online banking, on line brokerage services, and more sophisticated products transformed a highly predictable conservative business into a dynamic one. The increased risk of increasingly large sized banks, internationalization and increased product complexity was made possible through the continuous de-regulation of the industry. The Regale–Neal Act of 1994 reduced the barriers for geographical expansions of firms in the US and allowed interstate banking and The Gramm–Leach–Bliley Act of 1999 expanded the permissible activities of commercial banking as stated by Hendrickson [4]. Both acts led to merger and acquisitions amongst financial institutions and the creation of very large international businesses. The Glass–Seagull Act of 1933 did not allow commercial banking firms to participate in investment banking actives, but the act was repealed partly in 1994 and then the final parts repealed in 1999. The effect of this was to further increase the risk within the industry as people's monetary deposits where then being linked to more risky investment activities. The new

truly 'global financial industry' continued to attract the very best talent which then led to advances and more exotic product innovation.

Following the recent and on-going financial crisis we have seen governments trying to reverse the de-regulations of previous years; a number of laws have been introduced for example; the US House of Representatives passing the Wall Street Reform Act and Consumer Protection Act of 2009 [4]. The success of the measures governments are taking to try and re-regulate banks is questionable. Despite the huge attention and increased focus on audit, sign-offs and disclosures that accompanied the two acts cited, we are still seeing huge trading issues within leading institutions. Examples include the unauthorized rogue trading at UBS costing the firm $2 billion instantly [5], JP Morgan losing $5 billion via incorrect trading losses [6] and Barclays being fined a record amount of $453 million for the manipulation of LIBOR rates [7].

We have seen if an industry is not regulated correctly and at the same time continues to innovate with advances in technology that the successes and benefits of the industry may be out weighted by the problems and costs that can arise. Huge international companies are not easy to audit nor is it simple to get clear transparency of their risk positions. In 2012 there have been number of major regulatory interventions to try to prevent the same type of financial crisis as in 2008. Basel 11/111 will try to ensure that banks are holding enough capital, Wall Street reform and the Consumer Protection Act (Dodd-Frank law) will ban proprietary trading which was one of the main reasons banks become over leveraged and risked their existence [8].

So in summary the financial industry is a critical part of our society whose success can be linked directly to our prosperity. The industry's significance has grown since the 1980's and now banks are huge institutions that span the world selling often-complex products that are often difficult to control. The huge amounts of change impacting the industry will have a knock on impact on systems implementations. Ensuring internal projects are successful is one way a bank can help itself in difficult times.

Software project failures cost companies millions of dollars each year and often prevent key business objectives from being met. Failure estimates, defined primarily by cost and time budgets, overrun as high as 85 % of the original financial target. This is well documented in writings by Jiang [9]. Projects themselves are not just good implementations or bad ones. There are degrees of failure. Failures are too common when implementing financial systems and we will examine the reasons why in more depth.

3 Research Methodology and Data Collection Methods

A questionnaire was designed to collect data for this research. The questionnaire was designed for people that have implemented financial systems projects. The questionnaire required respondents to state their type of involvement in the

implementation and to read a set of systems implementation risks and rank risks from 1 to 13 according to its impact on the success of the overall project. Here ordinal scales have been used. Respondents were also asked to give each risk a second rating score according to how well it was executed. This score here is from 1 to 5. The questionnaire asked the respondents to choose the top 3 risks that could have been improved in the implementations they took part on. The questionnaire included open ended questions for respondents to then elaborate on how improvements could be made in these areas.

The final part of the questionnaire asked about the reasons for implementations and asked for overall judgments. The reason for the implementation question was answered by using a very simple nominal scale where there is no relationship or ordering to the numbers used. The questionnaire was administered electronically by email. Respondents were emailed initially to check their email addresses and give their agreement to participate in the research. A pilot questionnaire was constructed and given to 3 respondents to check that the instruction and meaning of the questions was clear. Feedback was given and taken on board on the layout and format of some of the questions.

The primary data collected in this research has been collected using a judgment sampling method. Remenyi et al. [10] acknowledge that judgment samples are inherently subjective but justify the use of judgment samples explaining how "samples are taken where individuals are selected with a specific purpose in mind, such as their likelihood of representing best practice in a particular issue", this means that the sample was essentially non-probabilistic. From the outset it became clear that statistical tests on this type of 'case study' research would have not been possible.

The sample size here was 40. Whilst this may appear to be a small number it does actually represent a large body of knowledge, experience and expertise in a less explored area of research. Respondents work for one of 11 top tier financial institutions, making in effect, a series of small case studies. Some of the banks include Barclays Bank, UBS, Citi Bank, HSBC, Credit Swiss, Lloyds and Bank of America.

The respondents were questioned from many different countries to represent a geographical spread. There is input from 9 countries but importantly, the key financial hubs around the world have been incorporated. These include London UK, Hong Kong, Singapore, New York US and Zurich Switzerland.

The research was split into 2 key aspects.

- A ranking of the risk categories to establish which is the most important to a successful implementation
- A rating to show which risks are normally well executed and which ones are not.

These aspects need to be analyzed to build the framework needed to help improve the success of future systems implementations in financial service industries.

The data was analyzed and presented by:

1. **By importance ranking**—risk factors were ranked in order of importance by respondents. An average was calculated and the results re-ordered and tabulated. The lower the number the more important the risk factor to an implementation.
2. **By execution rating**—an average was calculated for respondents' scores for execution. Each factor was averaged in turn. The higher the number the worse that factor was executed.
3. **A focus factor was calculated**—The importance ranking data and the execution rating data were combined to create a focus factor. The two data sets were added together and averaged. The focus factor illustrates the combined importance of that factor overall. Some factors are very important and executed well. Some less important factors were executed very badly. The combined position helps the project teams to understand the importance of the combined picture.

A framework for reducing implementation failure was created. The proposed framework uses the importance ranking, execution rating and focus factor results. Data from the questionnaires were combined to create the overall framework, pre-readiness assessment and during the project risk assessments. The framework was reviewed with two post project reviews in order to assess the usefulness of the framework.

4 Data Presentation and Discussion

When questioned about the success of software project implementations; 28 % of responses stated that the project went really well and improved the department. 31 % stated that the project went well but the capability wasn't really improved. 23 % stated that the project was ok but not worth the investment. In this case the respondents would not have started or commissioned the project if they had known the outcome. The most worrying scores where the next two categories. 10 % stated that the project was really poor and actually moved the department backwards. This was due to less functionality, poor reporting and poor processes. 8 % stated that the project was a complete disaster. All respondents were allowed to state the main reasons for issues with the implementations and the majority of responses state that a lack of resources and funding issues resulted in a compromise in the systems execution capability. Poor training or rushed user acceptance testing was also noted.

When asked to state the key things that went wrong the majority of answers fell into the following 6 categories:

1. **Scope Creep**—Project scope kept moving causing re-work, budget issues and productivity loss

2. **Budgets**—Budgets are always tight but due to issues with financial markets budgets are often cut. Scope creep without budget increase can cause lack of delivery
3. **Lack of engagement**—Poor communications resulted in the majority of the team feeling completely disengaged
4. **Poor Requirements**—The project delivered the requirements, but the requirements were incorrect and therefore the project was deemed to have failed
5. **Training**—Lack of UAT or user BAU training results in lack of adoption or resistance
6. **Leaders**—Leaders not resolving issues when problems happen. Conflict resolution or resources allocation then become issues that could then go off track and de-rail the implementation.

An analysis that examines the factors that make a system successful or not was conducted by using a questionnaire. Financial experts ranked 13 factors in order to show the most important and least importance factor in making a project implementation successful overall. This data was then split and cut into sets according to the level of use, knowledge or expertise etc. For example subject matter expert responses can be compared to the responses of people leading the project. This would be useful for example to compare the level of contributions from different roles and grades of staff within the company.

The success of each individual factor within an implementation has been assessed along with how well it was actually executed. So overall importance and execution can be compared.

Figure 1 has been constructed by looking at the overall rankings submitted by the respondents. The results have been generated by adding together and then averaging the ranking ratings. For example for user participation, the sum of the ranking scores is 126 as some respondents ranked it 1st and some ranked it 10th. On average people ranked it 3.9 out of 13 but this score made it the most important out of all the factors after all the factors had been added together and averaged one by one. Top management support's overall score was 150 giving an average score of 4.7.

This next section looks at the execution of each factor. This does not take into account ranking but purely whether the factor was executed well or not. Respondents rated their experience with each factor from 1 (very negative) to 5 (very positive). Scores were then added together and an average was calculated. Essentially the lower the score the least successful that factor was implemented, the higher the score the better that factor was implemented. The results can be seen in Fig. 2. A similar approach has been used with this data; the overall position of the factor has been calculated and then the data has been further organized according to role, use level etc.

Figure 2 shows that the execution factor ranking is very different to the importance ranking discussed earlier. The lowest scores (therefore showing the least effectively executed factor) are team pressure and conflict management

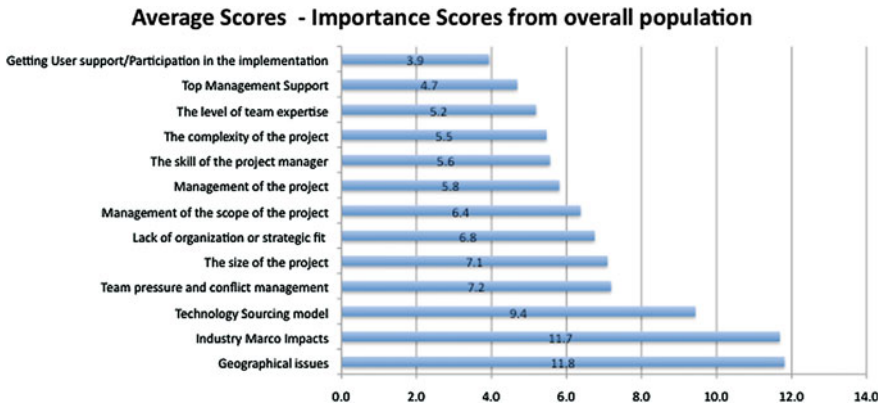


Fig. 1 Importance ranking

Fig. 2 Execution ranking

followed by management of scope, complexity and size of the project. The most successfully executed factors were top management support, team expertise and getting participation from users during the implementation. The latter set of factors were all ranked as the most important factors in the previous discussion.

The execution and importance were combined to create a joint list of important and focus for execution. By combining the two rankings and highlighting the learning points, there is the potential to reduce the negative responses, the like of which has been documented in the table below. This combined ranking puts a different emphasis on what needs to be focused on (Table 1).

Table 1 Areas of focus

Average of combined ranking				Total
Factors	Ranking overall	Ranking 1 and 2	Combined score	
Top management support	2	3	5	1
The complexity of the protect	4	4	8	2
Management of the scope of the project	7	2	9	3
Team pressure and conflict management	10	1	11	4
The level of team expertise	3	8	11	4
Getting user support/participation in the implementation	1	11	12	6
The skill of the project manager	5	8	13	6
The Size of the project	9	4	13	8
Lack of organization or strategic fit	8	6	14	10
Management of the project	6	12	18	11
Industry macro impacts	12	6	18	11
Technology Sourcing model	11	8	19	12
Geographical issues	13	13	26	13

5 Conceptual Framework

From the outset, this research set out to create a user friendly tool that could be used by professionals to better implement financial systems. Current research into the area and primary data has been combined to present a set of documents that can be used with finance teams to improve system implementations.

The framework was constructed using:

1. The importance ranking insight gained from the research
2. The execution rating insight gained from the research
3. The combined focus factor insight gained from the research

The overall framework is documented in Table 2 and starts with the main categories that cause project failures; top management support, scope change management and user participation are all examples here. The framework then explains the main risks and implications of not mitigating the risk. This is to help inform the project team of issues with system implementations. The framework then recommends the actions that need to be completed before and during a project. The use of the framework will not guarantee the success of a system implementation project but will help ensure a project is prepared, learns from basic errors other projects have made and self monitors its own progress.

Table 2 The conceptual framework

General focus areas	Implications and risk mitigating	Actions before you start	Actions during project
Pre-training and readiness training *Stakeholders/Responsible*: Project Leaders/ Subject Matter Experts/Users	It's important that everyone understands why projects go wrong, how to ensure they stay on track and the risks involved	(1) Project sponsor completes training and reviews readiness assessment (2) Subject matter experts and project leader complete delivery and mitigation training (3) A project delivery readiness assessment is completed	Project implementation progress assessment to be completed
Top Management Support *Responsible*: Project Leaders	People want to ensure that senior management support system implementation	(1) Ensure senior management support. Senior management complete the implementation training (2) Ensure there is public recognition of the support of the project (3) Send out communications from Exec sponsor and project lead (4) Ensure top management allocates the correct human and financial resources to make the project a success. Ensure the scope, financial budget and resources are matched together	(1) For large project implementations continuous communications of support will be required by Senior Management (2) Active participation in steering committees to ensure issues are understood and dealt with quickly (3) Scope changes are to be assessed before changes are made
Application Complexity *Responsible*: Subject Matter Experts/Users	Systems are often over complicated, don't use industry standards, don't reuse existing internal software and overly customized	(1) Plan to use standard functionality unless this is impossible (2) Complete a full buying vs build your own assessment before design is completed (3) Assess existing software to see if anything can be reused (4) Ensure there is a plan for updating the system in the future	(1) Document the system development to help reduce time to resolve issues and to help hand over the software to run

(continued)

Table 2 (continued)

General focus areas	Implications and risk mitigating	Actions before you start	Actions during project
Scope Change Control *Responsible*: Stakeholders/Project Leaders/ Subject Matter Experts	Scope creep can have a disproportionate impact on productivity, cost, morale and results in projects not delivering	(1) Ensure the requirements are signed off agreeing the scope of the project (2) Ensure the sponsor, project lead, project team and users understand what success looks like (3) Ensure there is a clear and communicated process to handle scope changes	(1) Ensure detailed design is again signed off and then delivered (2) Impact assessments of any change need to be signed off by senior stakeholders. Additional funding need to be secured before any project plans are changed
Team Expertise *Responsible*: Project Leaders	Without a team that can work together, with the right expertise then project will fail	(1) Get people with the skill and motivation to deliver a change project. The team needs technical and change management skills (2) The leader needs to be able to communicate with all stakeholders and sell the system. The leader needs the ability to say no	(1) Make a team effectiveness assessment to ensure we are getting the best from the team (2) Replace ineffective team members quickly (3) Risks are continuously assessed to ensure the project is delivered (4) Ensure there is an independent review and input regarding project progress
Team Pressure and Conflict Management *Responsible*: Project Leaders/Subject Matter Experts	Conflict needs to be managed carefully or fairly. These will arise so swift resolution is needed for sake of the project	(1) There needs to be a process of raising concerns in an open way to enable resolution (2) Detailed milestone plan, scope, budget and resources will be agreed upon before green light (3) Senior management need to foster open and honest discussions	(1) Complete detailed project reviews and ensure that the team agrees and signs up to schedule (2) Steering Committee will assess the progress of the project and ensure corrective action is made (3) Milestones deliverables will be assessed against original plan (4) Ensure there is an independent review and input regarding project progress

(continued)

Table 2 (continued)

General focus areas	Implications and risk mitigating	Actions before you start	Actions during project
User participation *Responsible*: Project Leaders	It is the users who will make the development work as they ensure the system works when it goes live. Active participation in making this happen is the only way to achieve results	(1) Lock in key personnel participation who are able to deliver this project (2) Spend time completing team building activities to ensure personality team dynamics are understood (3) Ensure subject matter experts sign-off requirements and UAT (4) Create a shared charter explaining how the project, risk management, communications and issues should be managed. Ensure all sign-off to it (5) End user training, UAT testing and continuous consultation needs to be at the heart of the program	(1) It is key to ensure the users have enough time to test and train on the system (2) Documentation of current processes and the new system process documentation are completed (3) Ensure the project team listens and understands the real issues and doesn't get completely absorbed by tasks (4) Test scripts are completed by the teams who will use the system going forward
Project Manager *Responsible*: Subject Matter Experts	A quality project manager is worth every penny. They have done this before, been successful and know what it takes do deliver	Lock in a project manager who knows how to project manage and has experience delivering the size of project required. If large project then you need a project manager who has completed projects before, understands planning, risk management, outstanding communication skills and strong budgeting skill. Ensure that the project manager has the technical understanding of what is required	Ensure the project manager has the authority to deliver the project

(continued)

Table 2 (continued)

General focus areas	Implications and risk mitigating	Actions before you start	Actions during project
Project size *Responsible*: Project Managers	Large projects are more complicated and this increases the risk. It's important that the project size is managed through development techniques and reduces risk	(1) Ensure roles and responsibilities are clear across the sponsor, project team and users (2) Break the project down into phased completions – Helps progress and de-risks the project (3) Bundle developments into releases, plan these in and communicate the future releases	Develop prototypes of components and get buyin before all development has been completed
Organization Fit *Responsible*: Stakeholders	A project that is not strategic is by nature tactical or regulatory	(1) Assessment against the end state architecture (2) Ensure you're clear on why the project is being completed – Technical/Strategic/ Regulatory – Short/Long Term Legacy – Replace/Enhancement (3) Ensure the benefits of the project are clear, calculated and communicated	Continue to explain the need and reason for the project compared to the strategic need of the business
Industry Macro Impacts *Responsible*: Stakeholders Management of the project	The industry is going through huge levels of change and this means short term changing priorities, constrained budgets and distracted leaders/employees	(1) Ensure that the last change in the industry will not impact the project (2) Ensure that there is a process to continue to gain on sponsor support	Ensure people keep focused on the future, the project and the reasons the project is being completed. Change will continue to happen so focus on deliverable by reducing the distractions.
  Responsible: Subject Matter Experts	It's important that plans are kept up to date, expectations are managed and issues raised to senior stakeholders quickly	(1) Complete a project plan that is realistic and has contingency with the plan (2) Ensure that the project plans covers a warranty period when the project goes live (3) Ensure that the project uses SDLC or any other structured implementation methodology	(1) Continue to replan and ensure activities are on track or manage expectations early (2) Communication and participation needs to be high to deliver the project (3) Selling the project and ensuring others really understand the progress is as important as the deliverable itself

(continued)

Table 2 (continued)

General focus areas	Implications and risk mitigating	Actions before you start	Actions during project
Technology Sourcing Model *Responsible*: Project Managers	It's important that complex project vendors delivering part of the project need to be coordinated carefully	(1) Understand how we can de-risk the project by using fix price bundles of work (2) Complete a full assessment of buying products vs build in-house	Continuously ensure that the vendors are delivering to the schedules they have committed to as part of the overall plan
Geographical Concerns *Responsible*: Project Managers	Communication, training and co-ordination are difficult issues during complex projects. These are made harder due to time zone differences, and cultural differences	(1) Need a plan to ensure we keep distance locations up to date with progress (2) Ensure there is a plan in place to gain participation, engagement, testing, training and support from more remote locations	Appreciate and accommodate different time zones and spend the time to engage and motivate more remote locations

6 Conclusions

The financial services industry is going through unprecedented levels of change. Due to the near banking collapse of 2008, banks have reduced earnings; they have greater levels of regulation, and are required to hold greater levels of capital. Leaders who are trying to manage these changes within institutions can lose focus on implementation projects. System implementations continue to be problematic, not delivering the functionality and benefits the projects promised from the outset. With reduced investment funds and distracted leaders a framework to reduce risk that is easy to use and effective will help projects deliver more. Easy to use tools to help educate leaders, subject matter experts and project leaders are needed. It is clear that issues are commonly repeated across organizations and basic to complex mistakes are continuously made. Although tools will help, it is important to note that system implementations are linked to people. People are the key factor in making it work: from senior leadership sponsorship to the expertise of project managers, from experts participating in development and the end users who will use the system, all play a role. It is important to understand that system implementations are huge change projects. Change projects impact people and while people remain flawed with agendas, then projects will continue to fail. The framework produced here is therefore people focused, helping people deliver better systems, de-risking the human role in system implementations.

References

1. Simon, P.: Why new systems fail: An insider's guide to successful IT projects. Course Technology, (2010)
2. Aldammas, A., Al-Mudimigh, A.: Critical success and failure factors of ERP implementations: two cases from the Kingdom of Saudi Arabia. J. Theor. Appl. Inf. Technol. **28**(2), 73–82 (2011)
3. Schoenmaker, D.: Banking supervision and resolution: the European dimension. Law Financ. Markets Rev. **6**(1), 52–60 (2012)
4. Hendrickson, J.M.: Regulation and instability in U.S. commercial banking [Electronic Book]: A history of crises. Palgrave Macmillan, Basingstoke (2010)
5. Morrow, R.R.: UBS sees integration as key after trading loss. Asia Money **22**(9), 18–19 (2011)
6. Lenzner, R.: The games played by JP Morgan Chase, Forbes.Com. p. 7, (2012)
7. Varriale, G.: 'Barclays rate-fixing scandal: Libor alternatives analyzed.' Int. Financ. Law Rev., **31**(6): 68, (2012)
8. Park, C,. IM, G., Keil, M.: Overcoming the mum effect in IT project reporting: the effect of time pressure and blame shifting, In Academy Of Management Annual Meeting Proceedings, pp. F1–F6. (2006)
9. Jiang, J.: Software project risks and development focus. Project Manage. J. **32**(1), 4–9 (2001)
10. Remenyi, D., Williams, B., Money, A., Swartz, E.: Doing Research in Business and Management: An Introduction to Process and Method. Sage Publications, London (1998)

Merging Compilation and Microarchitectural Configuration Spaces for Performance/Power Optimization in VLIW-Based Systems

Davide Patti, Maurizio Palesi and Vincenzo Catania

Abstract The rediscovery of VLIW architecture in the field of embedded multimedia applications introduces new challenges for computing paradigms historically oriented towards Instruction Level Parallelisms and performance optimization. In this work we perform an extensive multi-objective analysis which includes VLIW compiler as part of the configuration space, avoiding any explicit distinction between micro-architectural parameters and compilation strategies. After performing an high-level estimation of power/performance trade-offs by compiling and simulating some common application kernels, we qualitatively and quantitatively analyze of how the design space available can be greatly affected by the interaction of compiler behavior, processor-related features and memory subsystem.

Keywords VLIW · Design space exploration · Energy · Power · Compiler

1 Introduction

In the last years *Very Long Instruction Word* (*VLIW*) architectures [1] have been successfully applied to modern, increasingly complex embedded multimedia applications, given their capacity to exploit high levels of Instruction Level Parallelism (ILP) maintaining a good trade-off with cost and power consumption [2].

D. Patti (✉) · V. Catania
University of Catania, Catania, Italy
e-mail: dpatti@dieei.unict.it

V. Catania
e-mail: vcatania@dieei.unict.it

M. Palesi
Kore University, Enna, Italy
e-mail: maurizio.palesi@unikore.it

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_18,

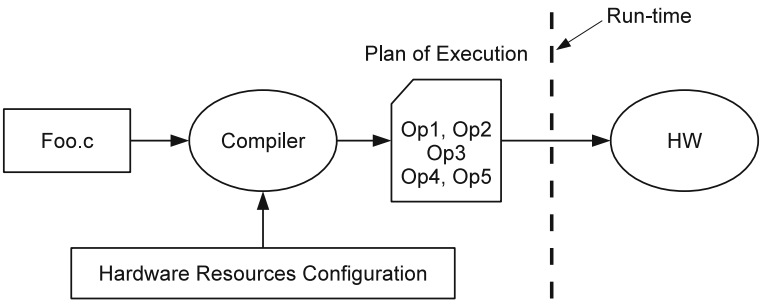


Fig. 1 Static VLIW code scheduling

What distinguishes the VLIW approach from other ILP-oriented architectures (e.g. superscalar) is the role of the compiler, which statically schedules the set of operation to be issued in the same clock cycle. The hardware, in fact, only carries out a *plan of execution* that is previously established at the compilation phase (see Fig. 1).

In embedded systems there are a wide variety of applications (multimedia, microcontrollers, sensors, etc.) with an equally wide variety of types and amounts of data to be processed. An audio/video application, for instance, will consist of transforming large data vectors that can be independently handled in parallel. The possibility of optimizing an architecture ad hoc to achieve high levels of ILP is thus of fundamental importance. However, the aim of maximizing ILP involves a set of aggressive code transformations which may significantly affect magnitudes other than performance, such as dissipated power and/or energy consumption. Energy consumption, for example, could prove to be a decisive factor in battery-powered mobile devices. Power dissipation, on the other hand, which is linked to the amount of heat the system is subjected to, is a fundamental element for aspects such as packaging, which directly affect the final cost of implementing the system.

The main contribution of this work is analyzing the design space of a VLIW architecture considering the compilation parameters as part of the same architecture. In other words, we will not make any distinction between micro-architectural parameters and VLIW compiler settings. Considering the configuration space as a whole, we want to quantitatively and qualitatively analyze how the additional interaction between hardware features (e.g., the number and type of functional units, register files) and code scheduling parameters can impact the design space exploration from a multi-objective perspective.

2 Background and Previous Works

There are a number of contributions in the literature regarding system-level exploration of VLIW-based architectures, basically differentiating by the objectives to be optimized and the architectural elements investigated.

A first area of research regards the design of high performance VLIW application-specific processors. The possibility of introducing application-specific functional units is analyzed in [3, 4]. Lapinskii et al. [5] present a kernel-specific and technology-independent methodology for exploration of the design space of clustered VLIW ASIP data paths. In [6] automatic optimisation of VLIW architectures for the execution of a specific application. Investigation of performance/area trade-off have been presented in [7]. To allow exploration of the extremely large number of configurations, hierarchical evaluation approaches have been adopted, separating exploration of the VLIW core and that of the memory subsystem. A drawback of both approaches is that parameters are explored independently each other, making it difficult to reach some interesting regions of configuration space due inter-dependency between parameters, as reported in [8, 9].

Alongside traditional research aiming at maximizing performance, interest has recently been shown in estimation and architectural exploration from the power and energy perspective [10, 11]. An instruction-level power model for VLIW architectures was proposed by Benini et al. [12]. Estimating the energy associated with a long instruction at the single pipeline stages. In [13] Pokam and Bodin explore the energy-delay tradeoff of ILP enhancing techniques at the compilation level. The impact on power and performance due to code transformation techniques in VLIW architectures is presented by Raghavan et al. [14].

The contribution introduced in this work is a multi-objective design space exploration analysis in which micro-architecture, memory subsystem and compiler parameters are considered as being part of a single system, i.e. regardless the nature of parameter when classified from typical designer perspectives such as “hardware” and “software”. Our idea is that this approach closely matches the VLIW design philosophy, which moves complexity from the microarchitecture to the compiler, so that a different compiler behavior is like having different control circuitry and hardware resources.

3 Simulation Environment

In this section, we briefly describe the parameterized VLIW platform [15] used as testbed for the experiments, the general evaluation flow along with the high-level estimation models used to evaluate the performance indexes to be optimized and the set of applications used as benchmarks.

3.1 Reference Architecture

The parameterized system architecture used in this work is based on HPL-PD [16] which is a parametric processor meta-architecture designed for research in instruction-level parallelism of VLIW architectures. The HPL-PD opcode

Table 1 Compilation parameters

Parameter	Description
tcc_region	Specifies the scope of action of the compiler and the type of code transformation involved (basic block, super block and hyper block)
max_unroll_allowed	The number of unroll iterations allowed
regroup_only	Avoids inlining
do_classic_opti	A set of classical optimization, not VLIW related, such as common expression removal
do_prepass_scalar_scheduling	Performs a schedule before forming regions
do_postpass_scalar_scheduling	Performs a schedule after the region formation
do_modulo_scheduling	Modular scheduling
memvr_profiled	Performs a memory-dependencies profiling

repertoire, at its core, is similar to that of a RISC-like load/store architecture, with standard integer, floating point (including fused multiply-add type operations) and memory operations.

Hardware architectural parameters can be classified in three main categories: *register files*, *functional units* and *memory sub-system*. The first two depend on the implementation of the VLIW core and regard the size of the register files, in terms of the number of registers contained in each of them, and the number of functional units for each type of unit supported. As far as the former are concerned, five different types of register files can be identified: GPR (32-bit registers for integers), FPR (64-bit registers for floating point values) PR (1-bit registers used to store the Boolean values of predicated instructions), BTR (64-bit registers containing information about possible future branches) and CR (32-bit control registers containing information about the internal state of the processor). The functional units involved are: *Integer units*, *floating point units*, *memory units* (associated with load/store operations) and *branch units* (associated with branch operations). With respect to the memory sub-system, the parameters that can be modified are the *size*, *associativity* and *block size* for each of the three caches: First-level data cache (L1D), first-level instruction cache (L1I) and second-level unified cache (L2U).

In order to investigate the effect of ILP-oriented code transformations, a set of compilation parameters, shown in Table 1 has been included in the configuration space.

For sake of simplicity, another set of compilation parameters have been fixed to some reasonable value and have been not included in the design space in this work. The complete list together with their default values is presented in Table 2.

3.2 Evaluation Flow

Together with the configuration of the system, the statistics produced by simulation contain all the information needed to apply the area, performance and power

Table 2 Compiler assumptions

Parameter	Value	Parameter	Value
Issue width	8	Unsafe jsr priority penalty	0.005
Min cb weight	20	Safe jsr priority penalty	0.01
Path max op growth	2.1	Pointer st priority penalty	1.0
Path max dep growth	4.25	Peel enable	$\surd$
Path min exec ratio	0.00075	Peel max ops	36
Path min main exec ratio	0.05	Peel infinity iter	6
Path min priority ratio	0.10	Peel min overall coverage	0.75
Block min weight ratio	0.005	Peel min peelable coverage	0.85
Block min path ratio	0.015	Peel inc peelable coverage	0.10

consumption estimation model implemented in the *Estimator* component of EPIC-Explorer. The results obtained by these models are the input for the *Explorer* component. This component executes an optimization algorithm, the aim of which is to modify the parameters of the configuration so as to minimize the three cost functions (area, execution time and energy/power consumption).

The average power consumed by the processor was estimated using an adaptation of the Cai-Lim model [17] to the VLIW processor. As regards the cache subsystem, a transition-based model was used, according to the equations described in [18]. The main memory energy is based on the model in [19] and assumes a per main memory access energy of 4.95×10^{-9} J based on the data for the Cypress CY7C1326-133 memory chip. The contribution towards power consumption made by the interconnection system was calculated by counting the number of transitions on the bus lines and applying the formula $P_{bus} = 1/2V_{dd}^2 \alpha f C_l$ where V_{dd} is the supply voltage, α is the switching activity, f is the clock frequency and C_l is the capacity of a bus line. For the on-chip buses we also considered the coupling capacitances between bus lines, using the model in [20]. For a detailed description of the models used and their adaption to the case of a VLIW based system see [21].

4 Analysis and Results

In this section we describe the set of experiments carried out in order to collect data useful for the analysis.

4.1 Experimental Setup

Each data set is obtained evaluation a 1,000 randomly chosen point of the configuration space. In particular, two different scenarios have been considered: the

Table 3 Benchmarks

Benchmark	Application
Bmm	Matrices multiplication and elements sum
Fir_int	Finite impulse response
Mm	Floating point matrices multiplication
Sqrt	Newton Raphson numerical analysis
Struct_test	Data structures allocation and access
Wave	Wavefront computation

first one, referred as *Variable Compilation Profile* (*VCP*), includes all compilation parameters described in the Table 1; a second scenario, *Fixed Compilation Profile* (*FCP*), which excludes any of those parameters exploring all the remaining hardware aspects of the architecture as described in Sect. 3.1.

The class of benchmark being considered is representative of some common frequently running application kernels in an embedded multimedia environment. Table 3 shows the set of applications chosen along with a brief description.

4.2 Quantitative Analysis

Figure 2 shows the visited configurations and the Pareto fronts found for both FCP and VCP scenarios for different application benchmarks. How it can be observed, VCP solutions are on average more widely and evenly distributed over the objective space as compared to those found for the FCP case.

Let us now analyse the Pareto fronts from a quantitative viewpoint. We consider two metrics, namely, the *variation range* and the *average normalised absolute dispersion error*. For a given objective, the variation range represents the ratio between the maximum and the minimum value observed for that objective. A comparison between the variation range for different benchmarks between a FCP and a VCP exploration for both power dissipation and execution time is shown in Fig. 3. As it can be observed, the VCP exploration provides solutions which fall on a range that is, on average, 23 and 40 % wider than that provided by a FCP exploration for power dissipation and execution time, respectively.

The average normalised absolute dispersion error measures the average absolute difference between the distribution of points in the objective space and an ideal distribution in which the points are uniformly distributed over the objective space. Formally, let O be the image, in the objective space, of the configurations visited by the design space exploration. The generic element of O (i.e., a solution) is a pair (p, t) where p and t are the average power and execution time, respectively. The two-dimensional objective space is then partitioned by a $M_x \times M_y$ mesh. For each tile T_i, $i = 1, 2, \ldots, M_x M_y$ of the mesh, let N_i be the number of points in O which fall in T_i. The average absolute error, E_i, for T_i is the absolute

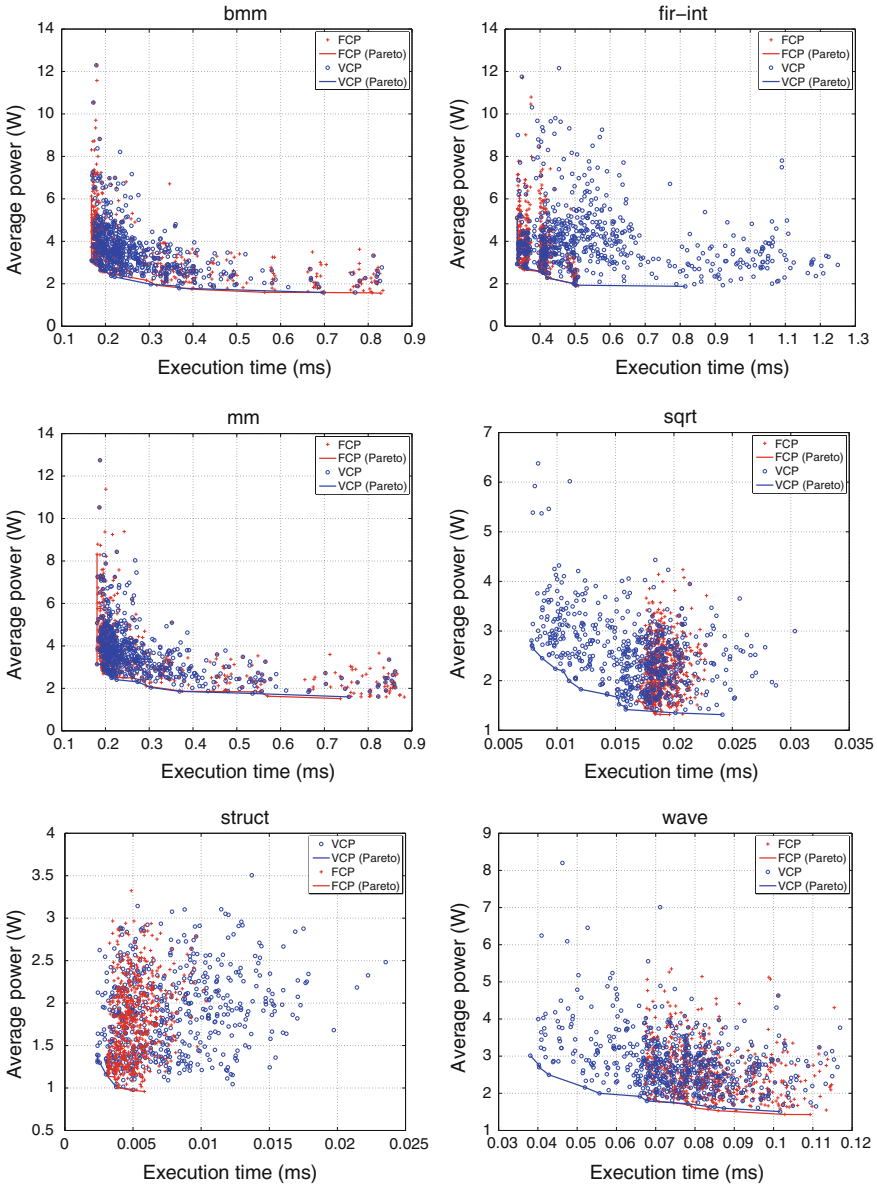


Fig. 2 Visited configurations and pareto fronts found by FCP and VCP exploration

value of the difference between N_i and the ideal number of solutions, $\overline{N}$, which should fall in T_i in case of uniform distribution. Such $\overline{N}$ can be simply computed as the ratio between the cardinality of O and the number of tiles. Thus,

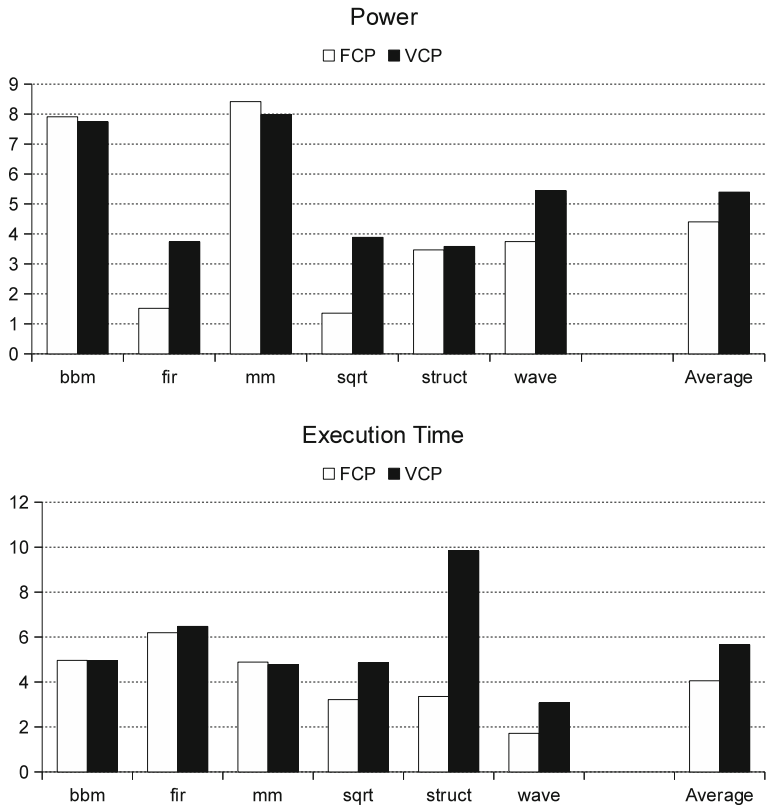


Fig. 3 Variation range for different benchmarks between a FCP and a VCP exploration for power dissipation and execution time

$$E_i = |N_i - \overline{N}|,$$

where $\overline{N} = |O|/(M_x M_y)$. The average normalised absolute dispersion error (*ANADE*) is the average of E_i normalised to the maximum absolute error E_{max}:

$$ANADE = \frac{\sum_{i=1}^{M_x M_y} E_i/(M_x M_y)}{E_{max}},$$

where E_{max} can be computed as the average absolute error in the worst case in which all the solutions fall in a single tile:

$$E_{max} = \frac{(M_x M_y - 1)\overline{N} + |\overline{N} - |O||}{M_x M_y}.$$

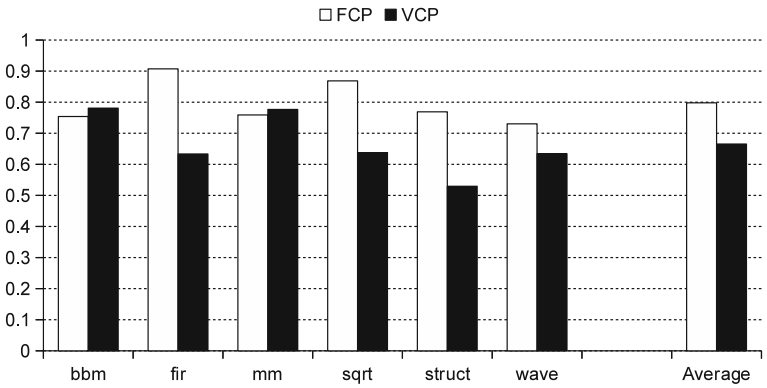


Fig. 4 Average normalised absolute dispersion errors for different benchmarks for FCP and VCP exploration

Figure 4 shows the average normalised absolute dispersion errors for different benchmarks for FCP and VCP exploration. As it can be observed, VCP exploration reduces the dispersion error on average by 20 % as compared to a FCP exporation.

5 Conclusions

In this work we analyzed the impact of ILP oriented compilation strategies on the design space of a VLIW architecture from a multi-objective perspective. After merging both micro-architectural and compilation parameters in a unique configuration space, we evaluated effects on performance, power and energy consumption for a set of representation application kernels. Future works will include a parameter-specific analysis of the interdependencies between compilation profiles, processor and memory subsystem.

References

1. Fisher, J.A.: Very long instruction word architectures and the ELI512. In: 10th Annual International Symposium on Computer Architecture., pp. 140–150 (1983)
2. Fisher, J., Faraboschi, P., Young, C.: Vliw processors: once blue sky, now commonplace. Solid-State Circuits Mag IEEE **1**, 10–17 (2009)
3. Boppu, S., Hannig, F., Teich, J.: Loop program mapping and compact code generation for programmable hardware accelerators. In: 2013 IEEE 24th International Conference on Application-Specific Systems, Architectures and Processors (ASAP), pp. 10–17 (2013)
4. Puppala, V.: Vliw—simd processor based scalable architecure for parallel classifier node computing. In: IEEE 3rd International Advance Computing Conference (IACC), pp. 1496–1502 (2013)

5. Lapinskii, V., Jacome, M., Veciana, G.D.: Application-specific clustered VLIW datapaths: early exploration on a parameterized design space. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **21**, 889–903 (2002)
6. Sabena, D., Reorda, M., Sterpone, L.: On the automatic generation of optimized software-based self-test programs for vliw processors (2013)
7. Najafi, M., Salehi, M.: Exploring the design space for area-efficient embedded vliw packet processing engine. In: 21st Iranian Conference on Electrical Engineering (ICEE), pp. 1–6 (2013)
8. Givargis, T., Vahid, F., Henkel, J.: System-level exploration for Pareto-optimal configurations in parameterized System-on-a-Chip. IEEE Trans. Very Large Scale Integr. Syst. **10**, 416–422 (2002)
9. Catania, V., Di Nuovo, A., Palesi, M., Patti, D., Morales, G.: An effective methodology to multi-objective design of application domain-specific embedded architectures. In: 12th Euromicro Conference on Digital System Design, Architectures, Methods and Tools (DSD '09), pp. 643–650 (2009)
10. Taniguchi, I., Uchida, M., Tomiyama, H., Fukui, M., Raghavan, P., Catthoor, F.: An energy aware design space exploration for vliw agu model with fine grained power gating. In: 14th Euromicro Conference on Digital System Design (DSD), pp. 693–700 (2011)
11. Pillai, A., Zhang, W., Yang, L.: Exploring functional unit design space of vliw processors for optimizing both performance and energy consumption. In: 21st International Conference on.Advanced Information Networking and Applications Workshops (AINAW '07), Vol. 1, pp. 792–797 (2007)
12. Benini, L., Bruni, D., Chinosi, M., Silvano, C., Zaccaria, V., Zafalon, R.: A framework for modeling and estimating the energy dissipation of VLIW-based embedded systems. Design Autom. Embedded Syst. **7**, 183–203 (2002)
13. Pokam, G., Bodin, F.: Understanding the energy-delay tradeoff of ILP-based compilation techniques on a VLIW architecture. In: 11th Workshop on Compilers for Parallel Computers, Chiemsee, Germany (2004)
14. Raghavan, P., Lambrechts, A., Absar, J., Jayapala, M., Catthoor, F., Verkest, D.: Coffee: compiler framework for energy-aware exploration. In: Stenstrm, P., Dubois, M., Katevenis, M., Gupta, R., Ungerer, T. (eds.) High Performance Embedded Architectures and Compilers. Lecture Notes in Computer Science, vol. 4917, pp. 193–208. Springer, Berlin (2008)
15. Ascia, G., Catania, V., Di Nuovo, A.G., Palesi, M., Patti, D.: Performance evaluation of efficient multi-objective evolutionary algorithms for design space exploration of embedded computer systems. Appl. Soft Comput. **11**, 382–398 (2011)
16. Kathail, V., Schlansker, M.S., Rau, B.R.: HPL-PD architecture specification: Version 1.0. Technical report, Compiler and Architecture Research HP Laboratories Palo Alto HPL-93-80 (2000)
17. Cai, G., Lim, C.H.: Architectural level power/performance optimization and dynamic power estimation. In: Cool Chips Tutorial colocated with MICRO32, pp. 90–113 (1999)
18. Kamble, M.B., Ghose, K.: Analytical energy dissipation models for low power caches. In: IEEE International Symposium on Low Power Electronics and Design, pp. 143–148 (1997)
19. Shiue, W.T., Chakrabarti, C.: Memory exploration for low power, embedded systems. In: 36th ACM/IEEE Conference on Design Automation Conference, New Orleans, Louisiana, United States, pp. 140–145 (1999)
20. Henkel, J., Lekatsas, H.: a^2bc: adaptive address bus coding for low power deep sub-micron designs. In: ACM/IEEE Design Automation Conference, Las Vegas, Nevada, USA, pp. 744–749 (2001)
21. Ascia, G., Catania, V., Palesi, M., Patti, D.: EPIC-explorer: a parameterized VLIW-based platform framework for design space exploration. In: First Workshop on Embedded Systems for Real-Time Multimedia (ESTIMedia), Newport Beach, California, USA, pp. 65–72 (2003)

Numerical Solution of Ordinary Differential Equations Using Mathematical Software

Jiri Vojtesek

Abstract The differential equation is mathematical tool widely used for description various linear or nonlinear systems and behaviour in the nature not only in the industry. The numerical solution of the differential equation is basic tool of the modelling and simulation procedure. There are various types of numerical methods, the ones described in this contribution comes from the Taylor's series and big advantage of all of them is in easy programmability or even more some of them are included as a build-in functions in mathematical softwares such as Mathematica or MATLAB. The goal of this contribution is to show how proposed Euler and Runge-Kutta's methods could be programmed and implemented into MATLAB and examine these methods on various examples. The comparable parameters are accuracy and also speed of the computation.

Keywords Differential equation · Numerical solution · Euler's method · Heun's method · Ralston's method · Midpoint method · Runge-Kutta's method

1 Introduction

The task of the modelling is the find appropriate mathematical description of the system which allows making simulation experiments on it. The differential equation is basic mathematical tool which is widely used by engineers for description of the dynamic behavior of the system [1, 2]. The reason why they are used is because of their accuracy in the description.

J. Vojtesek (✉)
Faculty of Applied Informatics, Tomas Bata University in Zlin, Nam. T.G. Masaryka 5555, 760 01 Zlin, Czech Republic
e-mail: vojtesek@fai.utb.cz
URL: http://www.utb.cz/fai

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_19,

There are two basic types of the differential equations (DE) linear and nonlinear. Unfortunately, the major part of systems are described by the nonlinear DE [3]. The numerical methods used for solving of these DE are basically single-step or multi-step [4, 5]. Typical the single-step method is an Euler's method or popular Runge-Kutta's methods. They are very popular because of their simplicity and easy programmability [5, 6].

The multi-step computation can be found in Adams-Bashforth or Predictor-Corrector method [5]. Difference between these methods is that multi-step methods needs for computation k previous steps, single-step methods needs only value in the previous step. Due to the length of the contribution, only single-step methods are mentioned and examined. Although the above mentioned methods are easy programmable, you can find them in various mathematical software used for simulation like Matlab [6, 7], Mathematica [8] etc.

The contribution has five main parts. The second part after this introduction describes theoretical background of the numerical solving of the Ordinary Differential Equations (ODE). The third part is focused on the usage of the mathematical software Matlab for this numerical solving, the next part shows the simulation results for two examples of ODE sets and the last part is conclusion.

All simulations performed in this paper are done in the mathematical software Matlab, version 7.0.1.

2 Numerical Solving of ODE

Numerical methods were tested on the ordinary differential equation in the general form

$$\frac{dy}{dx} = f(x, y) \tag{1}$$

with initial condition $y(0) = y_0$. This equation is called in the literature *Cauchy problem*.

2.1 Euler's Method

The simplest method is Euler's method which uses slope of the curve see Fig. 1. It is clear, that the slope from the curve could be computed as:

$$slope = \tan\phi = \frac{y_{i+1} - y_i}{x_{i+1} - x_i} = f(x_i, y_i) \tag{2}$$

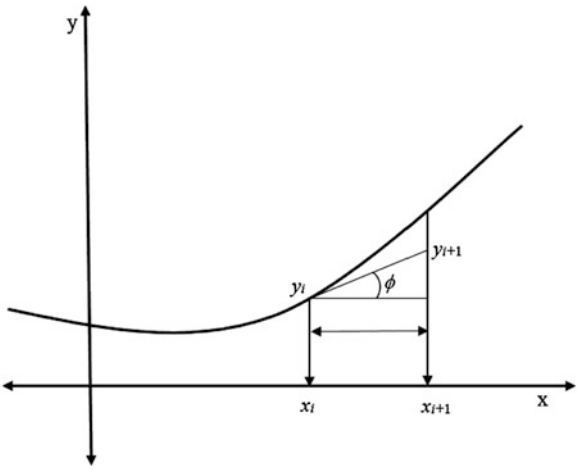


Fig. 1 Graphical interpretation of the Euler's method

which generally means for integration step $h = x_{i+1} - x_i$, that new value of y_i is computed from

$$y_{i+1} = y_i + f(x_i, y_i) \cdot h \tag{3}$$

Disadvantage of this method can be found in the high dependence on the integration step. The computation error grows with the increasing value of the step.

2.2 Runge-Kutta's Methods

All Runge-Kutta's methods mentioned later comes from the Taylor series of the (1):

$$\begin{aligned} y_{i+1} = y_i + \left.\frac{dy}{dx}\right|_{x_i,y_i} (x_{i+1} - x_i) + \left.\frac{1}{2!}\frac{d^2y}{dx^2}\right|_{x_i,y_i} (x_{i+1} - x_i)^2 \\ + \left.\frac{1}{3!}\frac{d^3y}{dx^3}\right|_{x_i,y_i} (x_{i+1} - x_i)^3 + \cdots \end{aligned} \tag{4}$$

which could be rewritten to the form

$$\begin{aligned} y_{i+1} = y_i + f(x_i, y_i)(x_{i+1} - x_i) + \frac{1}{2!} f'(x_i, y_i)(x_{i+1} - x_i)^2 \\ + \frac{1}{3!} f''(x_i, y_i)(x_{i+1} - x_i)^3 + \ldots \end{aligned} \tag{5}$$

As you can see, the first two parts of the Taylor's series (5) is Euler's method (3). Sometimes is this method called also *Runge-Kutta's first order method*.

Runge-Kutta's 2nd order method is more accurate then Euler's method and it uses first three parts of the Taylor's series (5), i.e.

$$y_{i+1} = y_i + f(x_i, y_i)h + \frac{1}{2!} f'(x_i, y_i)\, h^2 \tag{6}$$

which could be rewritten to the well-know relation:

$$y_{i+1} = y_i + (a_1 \cdot k_1 + a_2 \cdot k_2) \cdot h \tag{7}$$

where k_1 and k_2 are computed from:

$$\begin{aligned} k_1 &= f(x_i, y_i) \\ k_2 &= f(x_i + p_1 \cdot h, y_i + q_{11} \cdot k_1 \cdot h) \end{aligned} \tag{8}$$

and parameters a_1, a_2, p_1 and q_{11} are computed from relations [5]:

$$a_1 + a_2 = 1; \quad a_2 \cdot p_1 = \frac{1}{2}; \quad a_2 \cdot q_{11} = \frac{1}{2} \tag{9}$$

It is clear, that we have 4 unknown variables but only 3 equations which means that one of the variables must be set in order to reduce the complexity. Three basic methods are *Heun's method*, *Midpoint method* and *Ralston's method*.

Heun's method is defined for $a_2 = \frac{1}{2}$ which gives $a_1 = \frac{1}{2}$, $p_1 = q_{11} = 1$ and Eqs. (7) and (8) are

$$\begin{aligned} y_{i+1} &= y_i + \left(\frac{1}{2}k_1 + \frac{1}{2}k_2\right) \cdot h \\ k_1 &= f(x_i, y_i) \\ k_1 &= f(x_i + h, y_i + k_1 h) \end{aligned} \tag{10}$$

The second, so called *Midpoint*, method uses $a_2 = 1$ and it means that $a_1 = 0$, $p_1 = q_{11} = \frac{1}{2}$ and Eqs. (7) and (8) are

$$\begin{aligned} y_{i+1} &= y_i + k_2 \cdot h \\ k_1 &= f(x_i, y_i) \\ k_1 &= f\left(x_i + \frac{1}{2}h,\, y_i + \frac{1}{2}k_1 h\right) \end{aligned} \tag{11}$$

And finally the last, *Ralston's method*, have $a_2 = \frac{2}{3}$ and it means that $a_1 = \frac{1}{3}$, $p_1 = q_{11} = \frac{3}{4}$ and Eqs. (7) and (8) are

$$\begin{aligned} y_{i+1} &= y_i + \left(\frac{1}{3}k_1 + \frac{2}{3}k_2\right) \cdot h \\ k_1 &= f(x_i, y_i) \\ k_1 &= f\left(x_i + \frac{3}{4}h,\, y_i + \frac{3}{4}k_1 h\right) \end{aligned} \tag{12}$$

Runge-Kutta's 4th order method is the most commonly used method because of its accuracy. This method uses first five parts of the Taylor's series (5) which is transferred to the well-known form:

$$y_{i+1} = y_i + \frac{1}{6} \cdot (k_1 + 2k_2 + 2k_3 + k_4) \tag{13}$$

where variables k_{1-4} are computed from

$$\begin{aligned} k_1 &= h \cdot f(x_i, y_i) \\ k_2 &= h \cdot f\left(x_i + \frac{1}{2}h,\, y_i + \frac{1}{2}k_1\right) \\ k_3 &= h \cdot f\left(x_i + \frac{1}{2}h,\, y_i + \frac{1}{2}k_2\right) \\ k_4 &= h \cdot f(x_i + h,\, y_i + k_3). \end{aligned} \tag{14}$$

3 Numerical Methods in Mathematical Software

The numerical methods mentioned in the previous section are relatively old which means that they have strong background and support in the mathematical software. The most commonly used *Matlab* and *Wolfram's Mathematica* have even some of these methods as a build-in functions.

On the other hand, they could be easily programmed with the use of cycles like for, if etc. even in the C or C++ language. Matlab's programming language is very close to the C-language and the next chapters will show, how we can easily program our own numerical functions.

3.1 Euler's Method

The Euler's method has not its own build-in function but as it is clear from Eq. (3), this method is easily programmable. Let us make new Matlab function called `euler.m`:

Matlab's function euler.m

```
function [T,Y] = euler(odefun,h,t0,th,y0)
% Euler's numerical method
% inputs: odefun... solved function
%         h... integration step
%         t0... starting time
%         th... final time
%         y0... initial conditions
% output: T... time vector
%         Y... computed outputs
tspan = t0:h:th;
nv = length(y0);
N = length(tspan);
Y = zeros(nv,N);
Y(:,1) = y0;
for i = 1:N-1
   Y(:,i+1)=Y(:,i)+h*feval(odefun,tspan(i),Y(:,i));
end
Y = Y.';
T = tspan';
```

This function can be called from Matlab for example by command

```
[T,Y] = euler(@function,h,t0,th,y0)
```

where `@function` is solved function, h is integration step, `t0` and `th` are used for the start and final time and `y0` denotes initial values. Length of this vector is equal to the number of variables (number of equations).

3.2 Runge-Kutta's 2nd Order Method

Unlike previous method, the Runge-Kutta's second order method has its own build-in fuctions in Matlab `ode23` for ordinary `R-K` methods and `ode23s` for stiff differential equations. These functions are called from Matlab in the simplest way by command

```
[T,Y] = ode23(@function,[t0 th],y0)
```

with variables defined above in the previous chapter. If we have call this function with the above command, the integration step is variable depending on the actual computation error.

The RK 2nd order method generally defined by Eqs. (7) and (8) is easily programmable in Matlab too. The function `rk23.m` has form:

Matlab's function rk23.m

```
function [T,Y]=rk23(odefun,h,t0,th,y0,met)
% Runge-Kutta's 2nd order numerical method
% inputs: odefun... solved function
%         h... integration step
%         t0... starting time
%         th... final time
%         y0... initial conditions
%         met... method(1=Heun's;2=Ralston's;3=Midpoint)
% output: T... time vector
%         Y... computed outputs
%         a2 = 0.5 Heun's Method
%            = 2/3 Ralston's Method
%            = 1.0 Midpoint Method
switch met
    case 1 % Heun's Method
        a2 = 0.5 ;
    case 2 % Ralston's Method
        a2 = 2/3 ;
    case 3 % Midpoint Method
        a2 = 1 ;
end
a1=1-a2 ;
p1=1/2/a2 ;
q11=p1 ;
y0 = y0(:);
tspan = t0:h:th;
nv = lenght(y0); N = length(tspan);
Y = zeros(nv,N); F = zeros(nv,2);
Y(:,1) = y0;
for i = 2:N
  ti = tspan(i-1);
  hi = h;
  yi = Y(:,i-1);
 F(:,1) = feval(odefun,ti,yi);
 F(:,2)=feval(odefun,ti+hi*p1,yi+hi*q11*F(:,1));
  Y(:,i) = yi + hi*(F(:,1)*a1 + F(:,2)*a2);
end
Y = Y.';
T = tspan';
```

This function contains all three methods (Heun's, Ralston's and Midpoint method) mentioned theoretically in the part 2.2. The function is called by

```
[T,Y] = rk23(@function,h,t0,th,y0,met)
```

where met indicates computation method.

3.3 Runge-Kutta's 4rd Order Method

This method is the most used and it has build-in function in Matlab called `ode45` with Matlab's syntax

`[T,Y] = ode45(@function,[t0 th],y0)` With relation to Eqs. (7) and (8), the Runge-Kutta's method with fixed step could be programmed in function `rk45.m`:

Matlab's function rk45.m

```
function Y = rk45(odefun,h,t0,th,y0)
% Runge-Kutta's 4th order numerical method
% inputs: odefun... solved function
%         h... integration step
%         t0... starting time
%         th... final time
%         y0... initial conditions
% output: T... time vector
%         Y... computed outputs
y0 = y0(:);
tspan = t0:h:th;
nv = lenghth(y0);
N = length(tspan);
Y = zeros(nv,N);
F = zeros(nv,4);
Y(:,1) = y0;
for i = 2:N
ti = tspan(i-1);
hi = h;
yi = Y(:,i-1);
F(:,1) = feval(odefun,ti,yi);
F(:,2)=feval(odefun,ti+0.5*hi,yi+0.5*hi*F(:,1));
F(:,3)=feval(odefun,ti+0.5*hi,yi+0.5*hi*F(:,2));
F(:,4)=feval(odefun,tspan(i),yi+hi*F(:,3));
Y(:,i)=yi+(hi/6)*(F(:,1)+2*F(:,2)+2*F(:,3)+F(:,4));
end
Y = Y.';
T = tspan';
```

And it can be called from workspace by command

`[T,Y] = rk45(@function,h,t0,th,y0)` with the parameters defined above.

4 Verification of Numerical Methods

The numerical methods defined above were tested on two example functions. Results are compared at first visually, by the number of computation steps and also by the computation speed which was measured in Matlab with the use of functions `tic...toc`.

This time is relative and of course it could vary dependent on the hardware of the computer or version of the Matlab. The values mentioned here are measured on the same computer with the same condition.

4.1 Single Ordinary Differential Equation

The first example examines results of numerical methods mentioned above for simple ordinary differential equation

$$\frac{dx}{dt} = e^{-x} \tag{15}$$

with initial condition $y(0) = 3$.

One ODE can be in Matlab set simply with the use of function `inline` used for symbolic definition of the function, in this case:

```
f = inline('exp(-x)') and numerical solutions are then called:
[T,Y] = euler(f,h,t0,th,y0);
[T,Y] = rk23(f,h,t0,th,y0,3);
[T,Y] = rk45(f,h,t0,th,y0);
[T,Y] = ode23(f,[t0 th],y0);
[T,Y] = ode45(f,[t0 th],y0);
```

for starting time `t0 = 0`, final time `th = 9` and initial condition `y0 = 3`. The computation step is `h = 0.18` because the computation interval $<$ `t0, th` $>$ is divided into 50 parts.

Results in Fig. 2 are comparable for both manually programmed Runge-Kutta's methods `rk23` and `rk45` and Matlab's build-in functions `ode23` and `ode45`. On the other hand, the Euler's method is the less accurate which is clear even visually from the figure.

If we want to express computation error mathematically, we can introduce new variables Error, E, and Absolute Relative Error in Percentage, $AREP$, computed as

$$\begin{aligned} E &= \left| y_e(t_f) - y_c(t_f) \right| \\ AREP &= \frac{y_e(t_f) - y_c(t_f)}{y_e(t_f)} \cdot 100 \end{aligned} \tag{16}$$

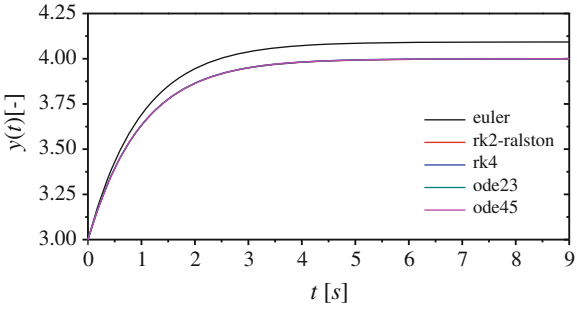


Fig. 2 Results of numerical solution of single ODE for 50 steps in Euler's method

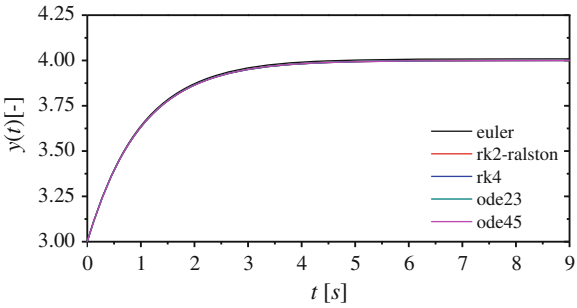


Fig. 3 Results of numerical solution of single ODE for 500 steps in Euler's method

where t_f is final time, in this case $t_f = 9$, $y_e(t_f)$ is exact value in final time and $y_c(t_f)$ is computed value in final time for Euler's method. Exact solution is in this case results of Runge-Kutta's 4th order method.

The computation error in Euler's methods is highly dependent on the number of steps. Results presented in Fig. 2 are for number of steps equal to 50. If we increase the number of steps ten times to 500, the results are much better see Fig. 3.

Dependence on the number of steps in Euler's method is presented in Table 1 and Fig. 4. It is clear that AREP error decreases exponentially with increasing number of computation steps. Disadvantage of the high number of computation steps is of course the computation time.

4.2 Matlab's Function Rigid

This function is defined in the Matlab's help and it has following form:

$$\begin{aligned} \frac{dy_1}{dt} &= y_2 \cdot y_3 \\ \frac{dy_2}{dt} &= -y_1 \cdot y_3 \\ \frac{dy_3}{dt} &= -0.51 \cdot y_1 \cdot y_2 \end{aligned} \tag{17}$$

Table 1 Values of errors for various number of steps

Steps (−)	E (−)	AREP (%)	Computation time (s)
5	1.1563	28.91	0.015
10	0.5165	12.91	0.016
50	0.0927	2.32	0.016
100	0.0457	1.14	0.031
500	0.0090	0.23	0.094
1,000	0.0045	0.11	0.218

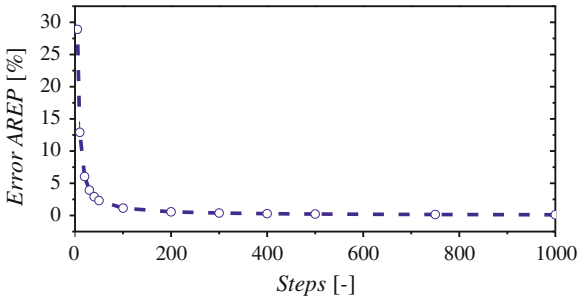


Fig. 4 Computation error AREP for various number of steps

with three variables y_{1-3} and t denoting time. The initial values are $y_1(0) = 0$ and $y_2(0) = y_3(0) = 1$.

There were defined function rigid in Matlab with the following form:

Matlab's function rigid.m

```
function dy = rigid(t,y)
% function RIGID
dy = zeros(3,1);
dy(1) = y(2) * y(3);
dy(2) = -y(1) * y(3);\
dy(3) = -0.51 * y(1) * y(2);
```

The simulation time was 20 s, 200 steps were chosen for programmed functions `euler`, `rk23` (Ralston modification) and `rk45`. The Ralston's Runge-Kutta's 2nd order method was mentioned because other two has very similar results. The function rigid can be then called with the M-file:

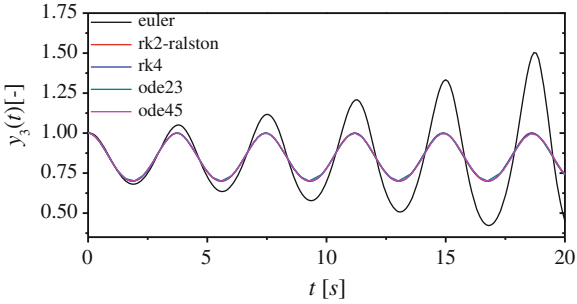


Fig. 5 Results of numerical solution of the function rigid output $y_3(t)$

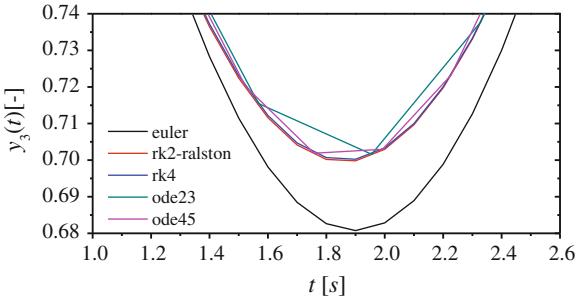


Fig. 6 Zoomed values of the numerical solution of the output $y_3(t)$

```
% simulation parameters
t0 = 0;      % starting time
th = 20;     % final time
h = .1;      % computation step
y0 = [0 1 1];   % initial conditions
% computation - choose one line only!
[T,Y] = euler(@rigid,h,t0,th,y0);
[T,Y] = rk23(@rigid,h,t0,th,y0,3);
[T,Y] = rk45(@rigid,h,t0,th,y0);
[T,Y] = ode23(@rigid,[t0 th],y0);
[T,Y] = ode45(@rigid,[t0 th],y0);
```

Figure 5 clearly shows that Euler's method has problem with the numerical solution similarly as in previous case. Here the output y_3 was mentioned as an example, but both other outputs y_1 and y_2 has similar results. This method could achieve better results with lower value of the integration step h but it causes also higher computation times as it produces more steps. Here, all methods have the same integration step $h = 0.1$ s.

Interesting thing can be found in Fig. 6 which shows zoomed values from Fig. 5. Here you can find that programmed functions `rk23` and `rk45` has smooth courses unlike build-in functions `ode23` and `ode45` which are not so accurate in all time vector. This is caused mainly by the bonus feature of these build-in

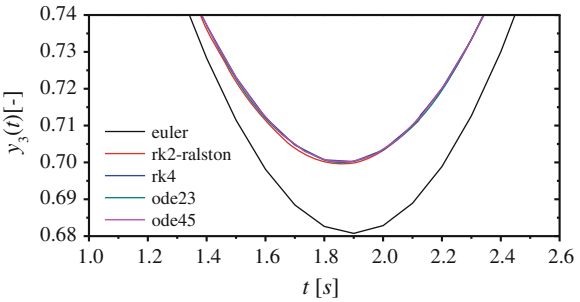


Fig. 7 Zoomed values of the numerical solution of the output $y_3(t)$—new computation

Table 2 Results the function rigid

Function	Steps (−)	Computation time (s)
euler	200	0.016
`rk23`—Ralston	200	0.016
`rk45`	200	0.047
`ode23`	88	0.156
`ode45`	113	0.094
`ode23` with options	592	0.172
`ode45` with options	333	0.093

function variable integration step. This property could sometimes speed-up the computation as it can be seen in Table, but it could result in inaccurate results.

This inaccuracy in build-in functions could be overcome if include option in the `ode23` or `ode45` functions

```
[T,Y] = ode23(@rigid,[t0 th],y0,option)
[T,Y] = ode45(@rigid,[t0 th],y0,option)
```

where option is

```
option = odeset('RelTol',1e-6)
```

and `'RelTol'` means relative toleration 1×10^{-6}. The results are then much better as it can be seen from Fig. 7. Table 2 also shows that with this option, computation took 592 or 333 steps respectively which is much higher than for other methods.

5 Conclusion

The paper describes the numerical solution of ODE using mathematical software Matlab. There were introduced basic numerical methods from the simplest Euler's method which is also the less accurate through the second order Rnge-Kutta's

methods with three modifications to the most accurate Runge-Kutta's method. The theoretical part shows how these methods are easy to program in the Matlab and also introduce the implementation of these methods via build-in functions `ode23` and `ode45`.

The practical part applies the proposed numerical methods on two examples with two main results. At first it shows that the accuracy of the Euler's method is highly dependent on the number of computation steps a higher number of steps produces more accurate results. The second result from the practical part is that it is not recommended to rely on the results from the build-in functions. These functions have implemented numerical improvements which changes the computation step adaptively according to the actual computation error which could, in some cases, provide inaccurate results. This disadvantage could be overcome with the use of parameter '`option`' which define relative toleration or with the use of the functions `rk23` and `rk45` described in the theoretical part which have fixed step during the whole computation.

References

1. Ingham, J., Dunn, I.J., Heinzle, E., Penosil, J.E.: Chemical Engineering Dynamics. An Introduction to Modelling and Computer. Simulation. Second, Completely Revised Edition. VCH Verlagsgesellshaft, Weinheim (2000)
2. Maria, A.: Introduction to modeling and simulation. In: Proceedings of the 1997 Winter Simulation Conference, pp 7–13 (1997)
3. Saad, Y.: Iterative Methods for Sparse Linear Systems. Society for Industrial and Applied (2003)
4. Johnston, R.L.: Numerical Methods. Wiley, New York (1982)
5. Kaw, K., Nguyen, C., Snyder, L.: Holistic Numerical Methods. http://mathforcollege.com/nm/
6. Mathews, J.H., Fink, K.K.: Numerical Methods Using Matlab. Prentice-Hall, Englewood Cliffs (2004)
7. Matlab's help to function ode23. http://www.mathworks.com/help/matlab/ref/ode23.html
8. Advanced Numerical Differential Equation Solving in Mathematica. Webpages of Wolframs Mathematica. http://reference.wolfram.com/mathematica/tutorial/NDSolveOverview.html

Global Dynamic Window Approach for Autonomous Underwater Vehicle Navigation in 3D Space

Inara Tusseyeva and Yong-Gi Kim

Abstract The marine world becomes more narrow and full of different objects that move unpredictably in the ocean space. The problem of increasing the capacity of the systems management in any kind of underwater robots is highly relevant based on the development of new methods for the dynamic analysis, pattern recognition, artificial intelligence and adaptation. Among the huge number of navigation methods, Dynamic Window Approach is worth noting. It was originally presented by Fox et al. and implemented into indoor office robots. In this paper Dynamic Window Approach was developed for marine world and extended to manipulate the vehicle in 3D environment. This algorithm is provided to avoid obstacles and reach targets in efficient way. It was tested using MATLAB environment and assessed as an effective obstacle avoidance approach for marine vehicles.

Keywords Dynamic window approach · Autonomous unmanned underwater vehicle · 3D environment · Obstacle avoidance

1 Introduction

Autonomous unmanned underwater vehicle (AUV) is a marine robot which moves under water in order to collect helpful information about different conditions of ocean bottom, the structure of the upper sediment layer or the presence of objects and obstacles. The main challenge related to all existing mobile devices that move independently without any control by the human remains the navigation. One of

I. Tusseyeva · Y.-G. Kim (✉)
Department of Computer Science and Engineering Research Institute (ERI), Gyeongsang National University, Jinju, Republic of Korea
e-mail: ygkim@gnu.ac.kr

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_20,

the works done toward solving the issue of multiple vessels navigation using Fuzzy logic was shown in [1].

For successful sailing in open space the onboard robot system should be able to build the route, control the motion parameters, and keep a track of its own position in a real time mode. One of the solutions to this issue is the integration of the novel algorithm named Dynamic Window Approach to control the motion of the vehicle. Fox, Burgard and Thrun were the first scientists who proposed it in 1997 [2]. That event led to the changing of automatic control notion in non-permanent environment and enabling the vehicle to move at high speeds. Thereby this Dynamic Window can be specified as the area of obstacle detection which depends on the speed of the vessel and can be changed dynamically.

As well as all nautical algorithms DWA chooses and constructs the most appropriate trajectories from initial to destination positions. The major difference from other approaches is that it controls the speed of a vehicle in order to avoid obstacles, for instance, when the sensors detect an obstacle, the robot will decrease the velocity or even stop. Additionally the algorithm allows the AUV to go on a maximal speed if there are no blockages on its path. It is obvious that all the calculations and decision making process must be handled dynamically [3].

Still there is no research work done toward the integration of DWA into autonomous unmanned vehicle navigation system. We made an attempt to develop this algorithm and proposed Global DWA with the enhancement toward the ability to move in narrow 3D ocean environment.

In Sect. 2, we will list some previous works related to the topic and describe their basic ideas. Then in Sect. 3 we will provide the details of the proposed approach. And finally the results of the experiments will be shown.

2 Related Work

There are a big number of methods and approaches applied in robotics that were presented by researchers and designed for rarely changing surroundings. One of the main problems of these algorithms is that they are not able to manage and make decisions when facing with dynamic circumstances.

For this purposes, the Fox, Burgard and Thrun first suggested DWA in 1997 [2]. They changed the notion of automatic control of objects in unstable conditions, while providing the ability to move robots at high speeds. Dynamic Window is presented as the area of obstacle detection which is changing dynamically depending on the speed of the robot [4].

Initially the technology was assumed to be used in machines, which are serving the staff inside the office building [2, 5]. Another practical example is shown in [6] where the robot with integrated system was tested during its exploitation in museum in order to conduct various excursions for visitors.

The number of scientists interested in this method increased during several decades. This fact is the evidence of its effectiveness. Among these researchers are

Brock and Khatib whose work [4] differs from Fox and many others in a way of developing the DWA algorithm for holonomic robots whereas Fox initially applied this method for synchro-drive robots [5]. And the search space in Fox's approach had the square shape when in Brock's approach it was a circle.

The researchers started to improve the equations of DWA after noticing the fact that the robot would likely to go far away from goal and increase its speed in some cases while maximizing the objective function. That is why the approach in [7] is supposed to be more efficient, compared to the original DWA, because the robot with proposed motion planning was able to decrease its speed before changing the direction due to obstacle detection.

The authors Seder and Petrovi [8] highlighted the differences between global and local path planning and described the idea of combining these two methods into one algorithm for more safe motion which is free of collisions. The research was based on their previous work [9] but it was improved with some changes which gave robots the ability to avoid collisions even with dynamically moving objects.

Another attempt to improve the Global DWA was made by the researchers in [10] who tried to use clothoid curves instead of circle that made the process of motion planning more real and close to machine moving simulation. Proposed approach was compared with Vector Field Histogram. The Virtual Force Field algorithm has been modified and expended to MVFF by the researchers in another article [11]. The development has been done by adding fuzzy logic in order to provide safe tracking without collisions. This approach has a big similarity with proposed in this article in a way of detecting obstacles: both our approaches uses a circle (in 3D) as a space to control and monitor and detect any obstacles around the vessel on a distance equal to the radius of the circle far from it.

3 Novel Navigation Approach

3.1 Dynamic Window Approach

The limitations of the autonomous planning methods has led researchers to study real-time planning, which is based on the knowledge gained from probing the local surroundings to handle unknown obstacles as far as the robot traverses a path in this environment. The similar characteristics are possessed by the approach described in this section based on Dynamic Window Approach.

The purpose of DWA was to handle the collision avoidance mission of the robot on a high speed in hazardous and populated environment. Its original idea was to face the problem of the robot's dynamics by considering only the speed of a vehicle.

DWA works out with the limitations of velocities and accelerations and provides the command generation in a small period of time. This approach is based on

a two-dimensional search space of two types of velocities. The pair values (v, w) are used to designate the velocity of the vehicle, where v is the translational velocity and w—rotational. The set of values (v, w) contains the speeds on which the vehicle can stop before colliding with any obstacles. These pairs are called admissible velocities and they are constructing the dynamic window with the current velocity as the center point (Fig. 1).

Admissible velocities, marked as V_a in the Fig. 1, can be calculated by using the following Eq. (1):

$$V_a = \left\{ (v, w) \middle| v \leq \sqrt{2 \cdot dist(v, w) \cdot \dot{v}_b} \wedge w \leq \sqrt{2 \cdot dist(v, w) \cdot \dot{w}_b} \right\} \tag{1}$$

where (v, w) is the set of velocities, translational and rotational, which can also be defined as the speed vector $\overrightarrow{v} = (v_x, v_y, v_z)$; $\dot{v}_b$ and $\dot{w}_b$ are accelerations for breakage; $dist(v, w)$ is the distance between the vehicle and the closest obstacle along the trajectory.

When searching for the set of admissible velocities the objective function must be taken into account by maximizing its value, as shown below (2):

$$G(v, w) = \delta(\alpha \cdot heading(v, w) + \beta \cdot dist(v, w) + \gamma \cdot vel(v, w)) \tag{2}$$

where *heading* (v,w) is the variable indicating the progress in the process of archiving the target; *vel* (v,w) is the translational (or forward) velocity which provides fast movements of the vehicle.

The overall search space (V_s) boils down to the dynamic window which includes the set of paces V_d that can be obtained within the next time period t. This space can be defined as shown in (3):

$$V_d = \{(v, w) | v \in [v - \dot{v}_b \cdot t, v + \dot{v}_a \cdot t] \wedge w \in [w - \dot{w}_b \cdot t, w + \dot{w}_a \cdot t]\} \tag{3}$$

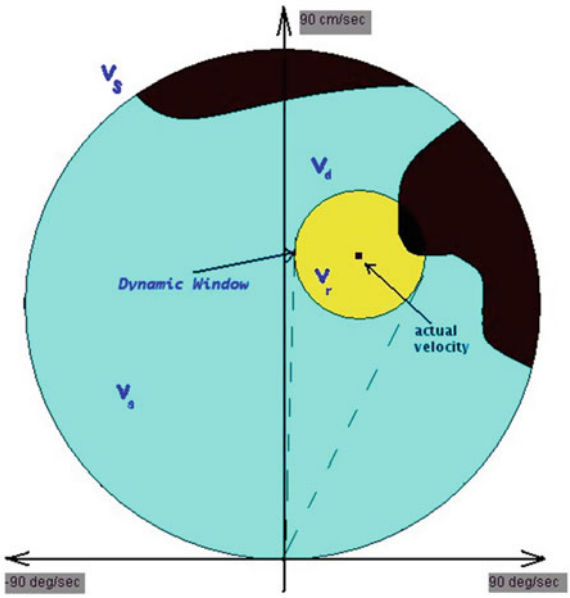
where v and w are actual velocities; $\dot{v}_a$ and $\dot{w}_a$ are values of translational and rotational velocity accelerations.

As shown in the Fig. 1, the space of velocities V_s contains the values of velocities V_a, whereas the space of V_d involves all velocities from V_r. The resulting search area can be represented as the traversal of the bounded spaces (4):

$$V_r = V_s \cap V_a \cap V_d \tag{4}$$

To demonstrate the logic of DWA we constructed the UML diagram (Fig. 2). First of all, the acceptable velocity of the vessel to reach the goal must be evaluated taking into account the actual position. Secondly, the algorithm will calculate the allowable liner and angular velocities based on the vehicles dynamics. The following process must be repeated in a loop for the list of allowable velocities: measure the nearest obstacle while the robot goes in a suggested

Fig. 1 Dynamic window



velocity; check whether the values of the breaking distance and the distance to the nearest obstacle are equal or not. The speed must be determined as admissible or not admissible. Next step is to measure the objective function which is consisted of heading and clearance values. The last step is to determine the cost value for the suggested admissible velocity and to compare it with all the other costs. If it is the best cost then the velocity must be considered as the best. As a result this velocity will be set to the robots acceptable trajectory.

3.2 3D Dynamic Window Approach

In previous section the main principles of DWA were described. The next step toward the novel navigation approach is the development of the motion control in 6DOF system. In order to implement this idea the 3 dimensional configuration space (3D-CSPACE) was confined to 3D dynamic window. This window must be enlarged to the shape of a sphere with radius r (Fig. 3).

In regard to the global coordinate system the robot in current position at time t is defined as $x(t)$, $y(t)$ and $z(t)$. The set of values (x, y, z, θ) determines the kinematic configuration of the vehicle, where $\theta(t)$ is the heading direction, or orientation, in another words. The formulas are presenting the motion equation for three axes (5):

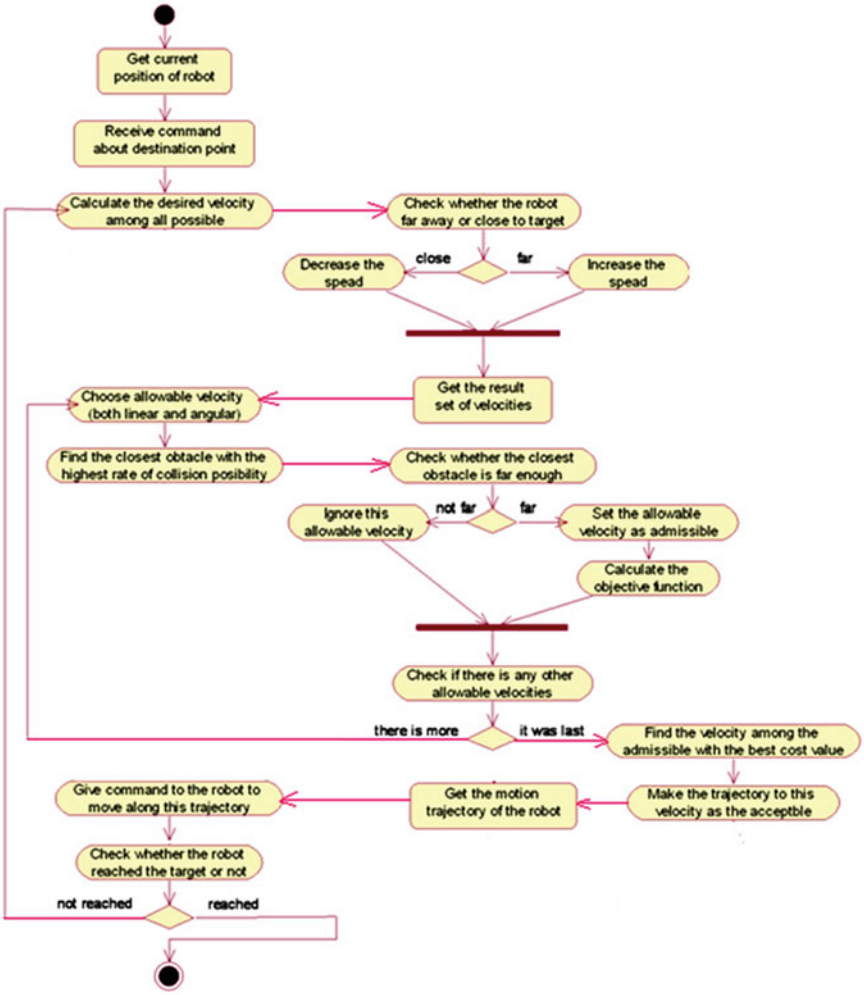


Fig. 2 Control flow diagram (UML) of proposed approach

$$
\begin{aligned}
x(t_i) &= x(0) + v_x t_i + \int_0^{t_i} a_x t dt = x(0) + v_x t_i + \frac{1}{2} a_x t_i^2 \\
y(t_i) &= y(0) + v_y t_i + \int_0^{t_i} a_y t dt = y(0) + v_y t_i + \frac{1}{2} a_y t_i^2 \\
z(t_i) &= x(0) + v_z t_i + \int_0^{t_i} a_z t dt = z(0) + v_z t_i + \frac{1}{2} a_z t_i^2
\end{aligned}
\tag{5}
$$

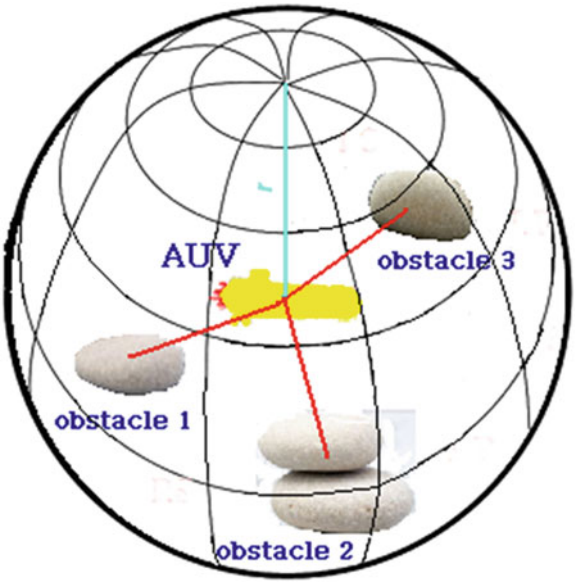


Fig. 3 The simple representation of DWA in 3D

To achieve a specific command of the speed changing when accelerating at a constant velocity the vehicle moves along a quadratic curve. And it continues until the desired speed approved by the algorithm will not be achieved by the AUV [4, 12]. Acceleration and curvature are mutually proportional. For instance, little acceleration creates a small curvature of the trajectory and allows simulating the behavior similar to cars [2].

The dynamic window is presented as the set of velocities V_d accessible within the next time interval t (3). Figure 4 illustrates the feasible trajectories of a vehicle in 3D ocean space.

If the vehicle can decrease its speed or even stop before collision with obstacles then the velocity pair (v, w) from the set of DW velocities is regarded as a safe (or admissible) (1).

Another indicator is a path alignment measure v_{path} [9]:

$$v_{path}(v, w) = 1 - \frac{\sum_{i=1}^{Nt} \sum_{i=1}^{Np} id_{ij} - D_{min}}{D_{max} - D_{min}} \tag{6}$$

where, N_t—discontinuous set of points on trajectory; N_p—set of points on the effective path (Fig. 5); d_{ij}—Euclidean distance between two points on trajectory and on effective path; D_{max} and D_{min} two limit values of the number of points on a curve.

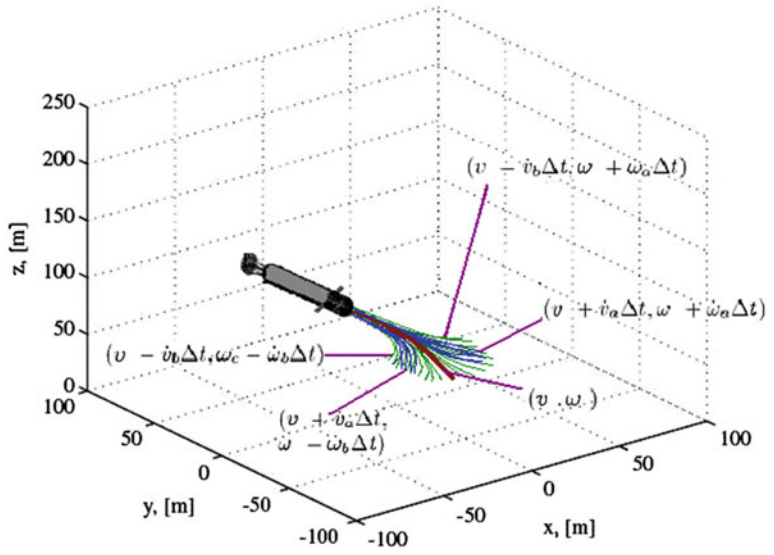


Fig. 4 Possible robot trajectories in 3D ocean space

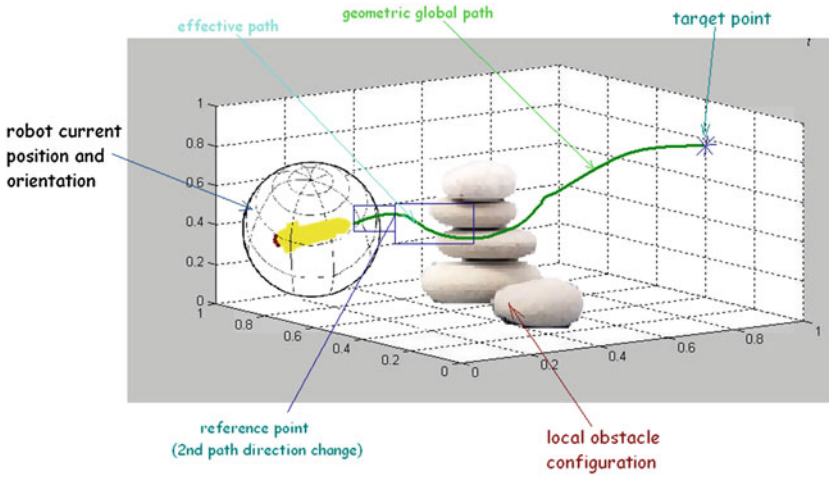


Fig. 5 3D view of x–y–z path with obstacle avoidance

3.3 Combination of Avoidance Algorithms: 3D Global Dynamic Window Approach

Nowadays the effective navigation method is usually presented as a system of algorithms genetically related with each other by combining both the autonomous monitoring mode and the real-time path navigation mode (DWA) with a simple map and efficient planning algorithm. The first part of the sailing approach named autonomous planner is looking for the best global path from start to target, whereas the second part is handling any possible crashes with previously unknown objects. This process has been done by replacing a part of the planned global-optimal path with the auxiliary path [13]. The algorithm first read the map and received the initial and target coordinates.

Global Dynamic Window Approach has already been proposed in [1]. In its essence DWA has no knowledge about the link of points on a path in a free space. It is the reason that in cooperation with some motion planning algorithm DWA could work out with this weakness. This function found the motion free of collisions from the initial position till the target point. The proposed 3D GDWA will be based on the concept of this idea.

The best matched motion algorithm has been proved to be the NF1 (Neuro-Fibromatosis type-1) because of its global, local minima free features [4]. GDWA integrates the sensor data into occupancy grid where the robot is presented in a form of a dot. The best way from start to goal positions is denoted as the shortest path which can be found by the NF1 algorithm [14].

While the GDWA is the extension of the original DWA it is obvious that it uses the same logic and equations as its predecessor. The key difference among them is the objective function (2) which evaluates the possibility of selecting among the potential moves that the robot can make. The novel objective function (7) presented above is changed by adding $nf1(v, w)$ instead of $heading\ (v, w)$ that centralizing the path of the vehicle toward the target point:

$$G(\vec{s}, v, w, \vec{a}) = \delta(\alpha \cdot nf1(v, w) + \beta \cdot dist(\vec{s}, v, w, \vec{a}) + \gamma \cdot vel(v, w) + \varepsilon \cdot \Delta nf1(\vec{s}, v, w, \vec{a})) \tag{7}$$

To determine $nf1(v, w)$ the weight of the nf1 must be matched at the cells neighboring to the robot's location. Additionally the function $\Delta nf1(\vec{s}, v, w, \vec{a})$ shows the extent to which the motion command will decrease the space between vehicle current position and target point during next repetitions.

To summarize, according to the proposed algorithm, during the first part of it (autonomous planner) robot looks for a globally optimal path from the start to the target, while the second part with 3D GDWA algorithm is responsible for processing potential collisions or previously unknown objects, replacing part of the original global path to sub-optimal path. Lastly, in tests done, both in the articles referenced above and in this research paper, it has proven to get the ability to navigate obstacle courses traveling as fast as the platform allows. 3D GWA has

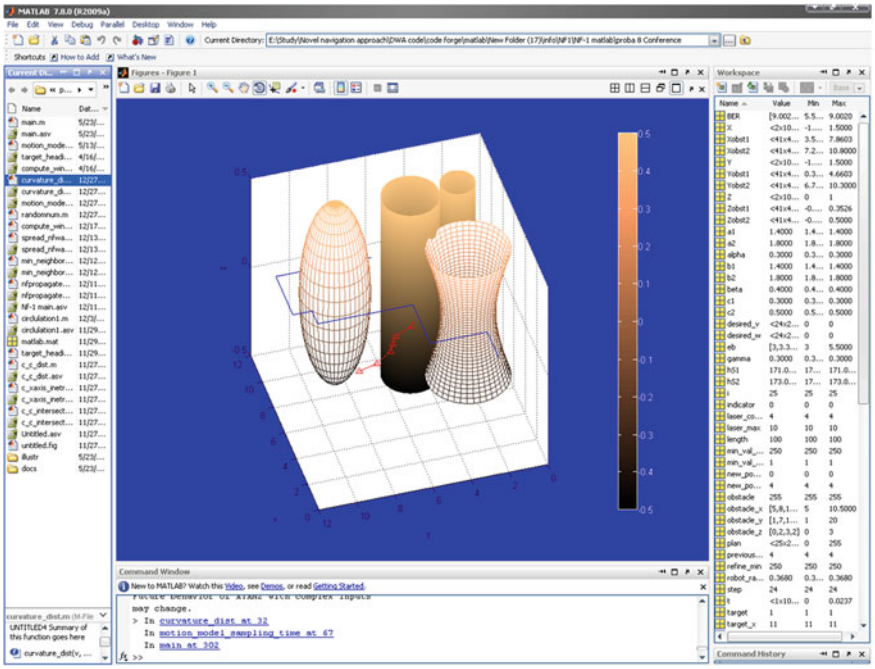


Fig. 6 Results of MATLAB simulator

been demonstrated as the approach which has the capacity to securely navigate obstacle courses moving on a high speed. Based on these conclusions, the GDWA is chosen to be implemented and further tested.

4 Experimental Results

In order to evaluate the reliability of proposed method the experiments were performed in MATLAB environment (Figs. 6 and 7).

The results of conducted tests illustrated that the algorithm is managing the tasks of navigation, such as building the most optimal path from start to target points, not exposing the AUV to any risk of collision (Figs. 6 and 7a, b).

All the arguments mentioned above proved the effectiveness of proposed algorithm while using it in narrow surroundings of underwater world. Furthermore DWA provides additional feature of controlling the dynamically changing speed values. As shown on Fig. 7b, the speed of AUV varies (rising or decreasing) depending on the route, presence of obstacles or the remoteness from the goal point. Additionally, we set the values for the motion planning of vehicle which were used in simulator (Table 1).

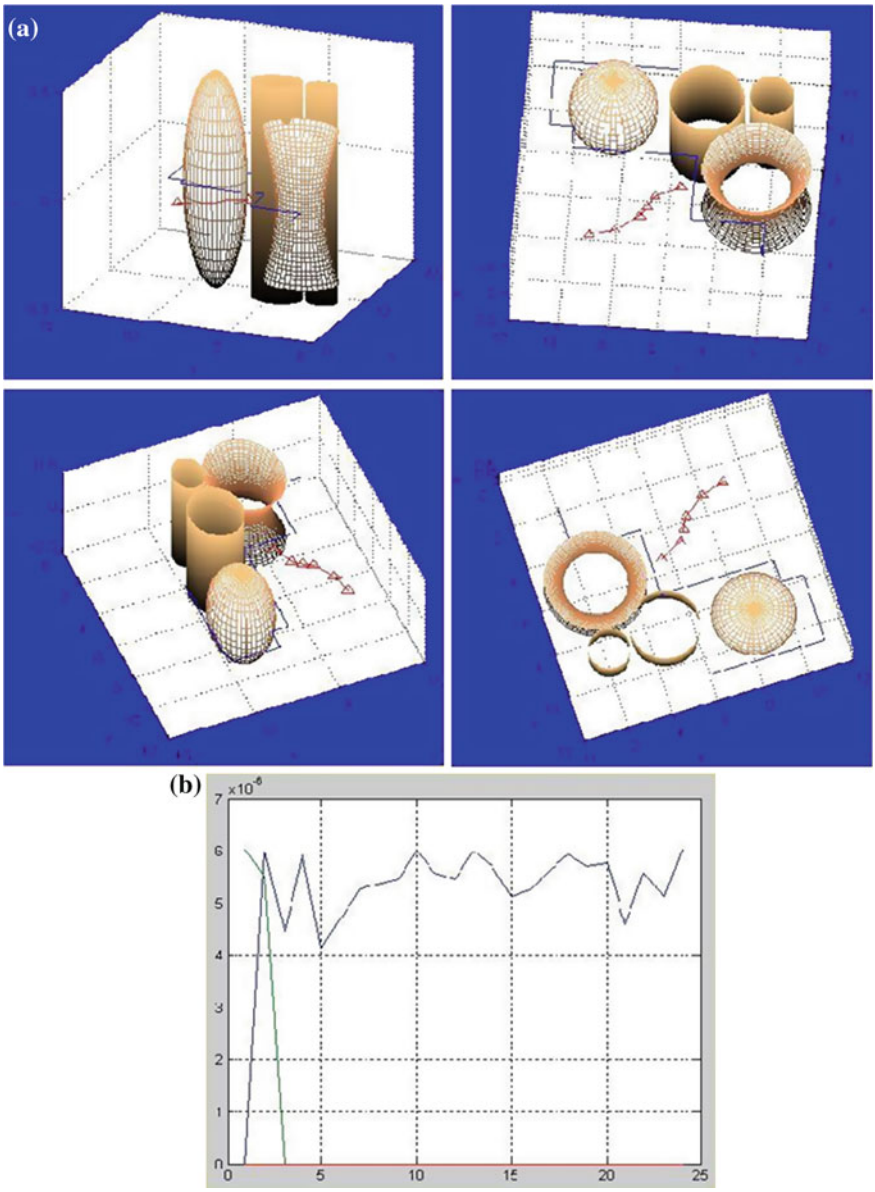


Fig. 7 Results of MATLAB simulator: **a** different views; **b** speed changes

Table 1 Technical prerequisites of motion planing

Maximum translational speed	$\leq$0.5 m/s
Minimum translational speed	$\geq$0 m/s
Translational acceleration	0.65 m/s^2
Maximum rotational speed	$\leq$1.57 rad/s
Minimum rotational speed	≥ -1.57 rad/s
Rotational acceleration	1.57 rad/s^2
Ultimate processor load	<30 %
Time to construct global path	12 s
Time to reach target	40 s

5 Conclusion

When creating an autonomous moving vehicle a number of issues could appear named "navigation tasks". We combined the existing approach DWA with the NF1 algorithm, provided the global features to the method and expended the calculation to 3D space. Global planning algorithms involve information about the whole space in order to identify areas where it is possible to move, and then determine the best path. The planning heuristic methods reduce the complexity of the task and the sensitivity to errors in the data in various ways. Therefore, for the development of a universal autonomous robot path following system the navigation evolutionary algorithm has been selected.

The 3D Global Dynamic Window approach was proved to be an effective solution to navigate AUVs in underwater surroundings. Selection of this algorithm makes it possible to take into account a set of behaviors of a vehicle and the environmental aspects in the path planning stage. However, the key issue of the proposed method is still remained as the absence of vehicle's ability to go up or down when avoiding obstacle. The solution of these problems will be the basic of the future research.

Acknowledgement This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. 2012R1A1A2038601).

References

1. Bukhari, A.C., Tusseyeva, I., Lee, B.G., Kim, Y.G.: An intelligent real-time multi-vessel collision risk assessment system from VTS view point based on fuzzy inference system. Expert Syst. Appl. (2013)
2. Fox, D., Burgard, W., Thrun, S.: The dynamic window approach to collision avoidance. IEEE Robot. Autom. Mag. (1995)
3. Simmons, R.: The Curvature-velocity method for local obstacle avoidance. In: Proceedings of IEEE International Conference on Robotics and Automation, pp. 375–3382 (1996)
4. Brock, O., Khatib, O.: High-speed navigation using the global dynamic window approach. In: Proceedings of International Conference in Robotics and Automation IEEE, vol. 1, pp. 341–346 (1999)

5. Fox, D., Burgard, W., Thrun, S.: controlling synchro-drive robots with the dynamic window approach to collision avoidance. In: Proceedings of IEE/RSJ International Conference of the Intelligent Robots and Systems, vol. 3, pp. 1280–1287 (1996)
6. Fox, D., Burgard, W., Thrun, S.: A hybrid collision avoidance method for mobile robots. In: Proceedings of IEEE International Conference on Robotics and Automation (1998)
7. Stachniss, C., Burgard, W.: An integrated approach to goal-directed obstacle avoidance under dynamic constraints for dynamic environments. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and System, vol. 1, pp. 508–513 (2002)
8. Seder, M., Petrovi, I.: Dynamic window based approach to mobile robot motion control in the presence of moving obstacles. In: IEEE Electrical Engineering Issue (2007)
9. Seder, M., Macek, K., Petrovi, I.: An integrated approach to real-time mobile robot control in partially known indoor environments. In: Proceedings of 31st Annual Conference of the IEEE Industrial Electronics Society, vol. 1785–1790 (2005)
10. Schröter, V., Höchemer, M., Gross, H.-M.: A particle filter for the dynamic window approach to mobile robot control. In: International Scientific Colloquium IWK, vol. 1, pp. 425–430 (2007)
11. Kwon, K.-Y., Cho, J., Kwon, S.-H., Joh, J.: Collision Avoidance of moving obstacles for underwater robots. J. Syst. Cybern. Inf. **4**(5), 86–91 (2006)
12. Khatib, O.: Real-time obstacle avoidance for robot manipulator and mobile robots. Int. J. Robot. Res. **5**(1), 90–98 (1996)
13. Bouguet, J.-Y., Perona, P.: Visual navigation using a single camera. In: Proceedings of ICCV, Boston, USA (1995)
14. Latombe, J.-C.: Robot motion planning. Kluwer Academic Publishers, Boston (1991)

UAC: A Lightweight and Scalable Approach to Detect Malicious Web Pages

Harneet Kaur, Sanjay Madan and Rakesh Kumar Sehgal

Abstract Attackers mostly target users with vulnerable browsers thus inducting client side attacks through various exploitation means, where dynamic client-side JavaScript is most instrumental. In this paper, we present UAC (URL Analyzer and Classifier), a novel lightweight and browser-independent solution that leverages static analysis combined with run-time emulation to identify malicious web pages. UAC performs multi-facet inspection of web page which includes DOM parsing to identify suspicious DOM elements including hidden iframes and malicious links, JavaScript analysis to detect obfuscated and malicious behavior using function-call profiling based on supervised learning, tracking dynamic domain redirections and scanning for suspicious patterns. An Active potential URL hunt to seed web pages is conducted using an integrated web crawler to cover the maximum cyber space for a given URL. The solution is employed as a Low Interaction Honeyclient in a Distributed Honeynet System where the scalability is addressed using a hash-based redundancy check.

Keywords Static analysis for malicious websites · Run-time emulation · Low-interaction Honeyclient · Web crawler · Obfuscated and malicious JavaScript · Suspicious DOM elements · Signature scanning · Machine learning

H. Kaur (✉) · S. Madan · R. K. Sehgal
Cyber Security Technology Division, C-DAC, Mohali 160071, India
e-mail: harneet@cdac.in

S. Madan
e-mail: msanjay@cdac.in

R. K. Sehgal
e-mail: rks@cdac.in

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_21,

1 Introduction

Internet has become the most popular medium of communication and global information reservoir. With the increasing popularity of public social networking sites, the whole universe seems to congregate around internet to get his/her share of web. Though the general impression is the growing cyber security awareness among the masses, but the advanced hacker techniques and sophistication seems to counter the defensive mechanisms easily and befool the users.

Malicious web contents today primarily target web clients with browser vulnerabilities. Particularly, Drive-by-downloads [1] are specific types of web based client-side attacks in which a web browser requests web pages from remote web server. As a response, the server returns a webpage to the browser that contains attack code to exploit the web browser's remote code execution vulnerability. If the malware is not delivered as part of the attack code's payload, a special payload called downloader can optionally first pull and then execute malware on the local workstation. The entire attack happens without the users consent or notice. These attacks normally take advantage of tight coupling of browser plug-ins with browser environment. The memory of browser is physically shared with its various extensions thus making it highly susceptible to heap spray [2] or other similar attacks. The deterministic heap behavior causes the attacker to reliably assume the complete control of browser memory and eventually the entire system.

Detection domain of malicious websites primarily focuses on following strategies:

(a) **Browser Built-in Protection**
Browser Protection Plug-ins [3], Safe-Browsing like Google [4]
(b) **Static and Machine Learning Approaches**
JavaScript Features [5, 6], HTML and URL Structural Processing [6] and HTTP Communication Patterns [7], Pattern-Matching [8]
(c) **Memory Monitoring**
Memory Corruption and Heap Spray Detection [9], Data Memory Protection [10]
(d) **Emulation-Based Mitigation Technique**
Browser Emulation with HTTP response verification, Sandboxing the Script Execution and Result Verification [11]
(e) **Impact Learning**
Monitoring downloaded content correlated with User Events [12], Un-consented Content Execution Prevention [13].
(f) **HoneyClients**

- **Low Interaction Honeyclients**. HoneyC [14], PhoneyC [15], Honeysift [16], Monkey-Spider [17], Honeyware [18]
- **High Interaction Honeyclient**. Capture HPC, Honeyclient, HoneyMonkey, Shelia, UW Spycrawler, WEF.

UAC (URL Analyzer and Classifier) is a lightweight solution that leverages static analysis combined with run-time emulation to identify malicious web pages. It performs inspection of a web page from multiple dimensions, which includes DOM parsing to identify potentially suspicious DOM elements including hidden iframes and malicious links, JavaScript analysis to detect obfuscated and malicious behavior, dynamic domain redirection tracking and scanning for suspicious patterns. UAC has the following features to offer:

Hybrid Analysis Framework. UAC offers hybrid analysis capability to counter the hidden techniques employed by Blackhat and to cover reasonable analysis domain. Run-time emulation facilitates safe inspection environment and exposure of dynamic behavior whereas employment of static analysis offers fast investigation.

Light-weight Approach. It has been tested with respect to system and performance measurements and has proved to incur less overhead. It demands minimum system resources and take around 20 s for each analysis.

Supervised Learning-based model. The JavaScript analysis and its behavioral profiling are based on supervised learning models to deliver accurate results.

Distributed Deployment. The solution has been deployed as a Low Interaction Honeyclient at various geographical locations to permit distributed load balancing and capturing of targeted attacks (Region-specific attacks).

Scalable Solution. The hash-based technique to eliminate the process of redundant URL analysis has been integrated. Also, the architectural implementation ensures that the analysis is done at client-side and the analysis results are mapped to central server which reduces transmission load and also consumes less network bandwidth.

Evaluated Version. It has been evaluated against various open-source Low Interaction Honeyclients and also with Google–Safe browsing. The results depict that UAC is very effective in detecting malicious URLs with a very low false positive rate of 0.2 % and false negative rate of 0.08 %.

2 Related Work

Caffeine Monkey [19] is a Client-Side Honeypot technology to identify browser exploitation. It employs a JavaScript de-obfuscator, logger, and profiler to identify malicious websites. JavaScript behavioral analysis is based on its function-call analysis. Whereas the common aspect of Caffeine Monkey and UAC is the use of function calls for JavaScript analysis, the significant difference lies in the selection of function calls. UAC makes use of 33 JavaScript function calls, which have been selected after rigorous experimentations on various websites that download malware.

Binspect [20] makes use of emulation and static analysis to detect Drive-by-Download and phishing attacks. It employs machine learning models based on URL features, Page-Source features (HTML and JavaScript), and Social-Reputation features. UAC however analyzes the web page from the behavioral features rather than structural features for more accurate interpretation.

ZOZZLE [21], a fast and precise in-browser JavaScript Malware Detection is based on static JS analysis using function-call hooking in browser JS engine. Bayesian classification of hierarchical features in the form of JavaScript abstract syntax tree is used to identify syntax elements that are highly predictive of malware. However, it primarily addresses No-op and heap spray attacks. The obfuscation detection of JavaScript in UAC is primarily derived from "Automatic Detection for JavaScript Obfuscation Attacks in Web Pages through String Pattern Analysis" [22] that makes use of n-grams, entropy and string length to identify obfuscation in scripts.

Jstill [23] enables detection of obfuscated JavaScript and function invocation based analysis to detect malicious JavaScript. It also highlights the discrepancies of browser-based mechanisms. However, the analysis is based on inspecting arguments of function calls that are dynamically invoked. UAC, on the other hand makes use of the statistical and sequential features inherent in function call invocation, where obfuscation detection is done in a separate thread.

"Knowing your enemy: understanding and detecting malicious web advertising" [24] has developed Mad Tracer for Spam, Drive-By-Downloads, and Click Frauds. It analyzes hidden iFrame injections and redirections. UAC also provides information of iFrames and malicious links but it identifies all iFrame and analyzes them according to their visibility index and structure. In addition, it also identifies suspicious links on a web page.

3 Problem Definition and Approach Adopted

Being a type of client-side attack, detection of Drive-By-Download attacks needs to be addressed at client-side. The problem statement can be stated as the development of Client Honeypot for (a) Overcoming the challenge of multiple browser-OS combinations to detect actual system exploit (b) Capturing static and dynamic webpage contents (c) Inspection of dynamic JavaScript behavior to detect malcode and/or redirections (d) Large-scale deployment of the analysis mechanism which demands a low-overhead and fast approach, in addition to addressing scalability.

3.1 Approach Adopted

To address the above problem statement, UAC has been developed which employs emulated browser and JavaScript engine that facilitates the execution of URLs and JavaScripts in safe emulated environment without the need to configure browser-specific environment. Use of emulation enables the capturing of static and run-time (dynamically) generated web contents including potentially malicious iframes and links. Use of JavaScript engine enables the inspection of dynamic JavaScript behavior thus defeating the mechanisms of obfuscation and other code-hiding techniques used by attackers. Various Challenges and their solutions provide an overview of the approach adopted:

3.2 Challenge 1: Overcoming the Challenge of Multiple Browser-OS Combinations to Detect Actual System Exploit

UAC is a browser-independent solution that utilizes emulated browser and JavaScript engine to facilitate the execution of URLs and JavaScripts in a safe emulated environment (protected from self-exploitation) without the need to configure browser-specific environment.

3.3 Challenge 2: Capturing Available and Generated (Static and Dynamic) Webpage Contents

Execution of URL using browser that is configured with DOM parser and JavaScript engine permits monitoring of static and run-time web contents including likely malicious iframes, links, and invoked scripts.

3.4 Challenge 3: Transient Malware Compromises Effectiveness of Static Analysis

Transient JavaScript malware can be effectively monitored during run-time where it renders its actual behavior. Hybrid analysis technique (static and run-time) is employed in UAC that exposes the dynamic behavior of webpage.

3.5 Challenge 4: Inspection of Dynamic JavaScript Behavior to Detect Mal-code and/or Redirections

Use of JavaScript engine in UAC enables the inspection of dynamic JavaScript behavior thus defeating the mechanisms of obfuscation and other code-hiding techniques used by attackers.

3.6 Challenge 5: Establish Significant (Legitimate and Illegitimate) JS Function-calls

Thirty three JavaScript function calls have been selected after rigorous experimentations (using commercial sandbox) on JavaScripts extracted from sites that drop malware. These function calls exhibit the most frequent occurrences in suspicious web sites.

3.7 Challenge 6: Scalability Aspects

Hash-based redundancy check has been applied in UAC to prevent redundant URL analysis.

4 UAC Design

Figure 1 illustrates the design of UAC in which the input is a set of seed URLs which are further crawled and then analyzed. The input URLs are executed using emulated browser and relevant parameters are captured. UAC declares any site as "Likely Suspicious", "Suspicious", "Highly Suspicious", "Benign", and "Error". This classification is based on final rule-set generated after URL analysis.

4.1 URL Active Crawling

The active URL hunt is done using a web crawler that extracts web links from a given web-page. URL crawling pursues standard algorithm that downloads website contents and extract links based on recognized patterns.

An important challenge in the implementation of web crawler is the selection of an optimum crawling depth. If depth is too low, associated crawling becomes limited to few sites. Large crawling depth produces an enormous overhead and becomes the bottleneck in the whole analysis process. Table 1 summarizes the

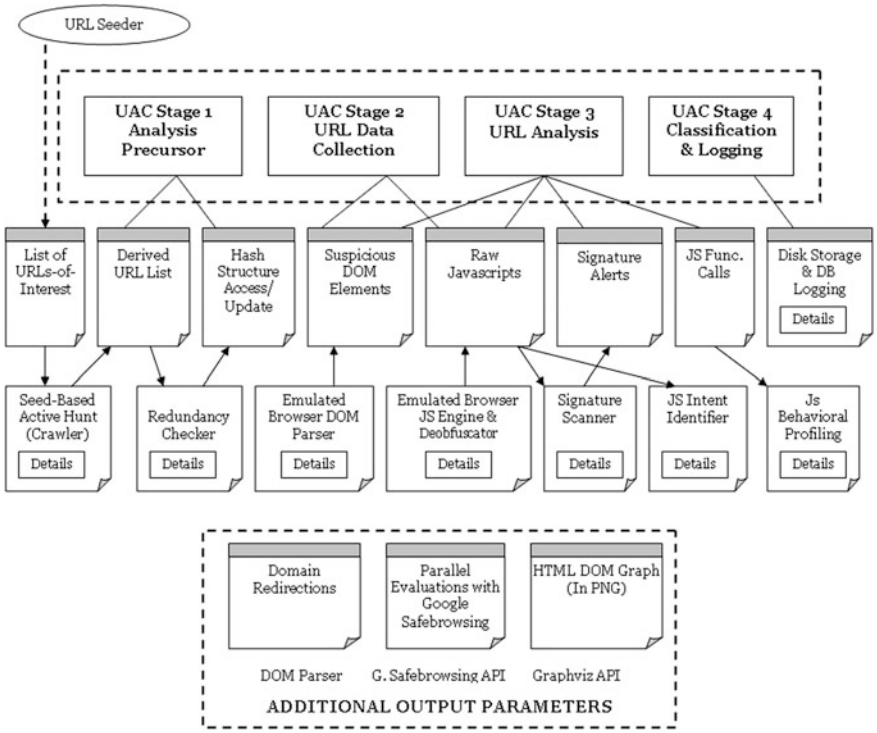


Fig. 1 UAC modular design

Table 1 Crawling depth selection

Crawling depth	Total URLs	Malicious URLs	Benign URLs	Memory consumed (MB)	Time consumed (s)
0	550	34	402	4.3	18.15
1	1,130	41	986	8.9	37.29
2	1,536	59	1,321	12.07	15.7
3	2,513	61	2,212	20	83

output of various experimentations that were carried out to select the most suitable depth value. The processing overhead incurred by web crawler on system can be averaged as:

Time Consumption: 0.033 s/URL (Average)
Memory Consumption: 7.86 kb/URL (Average)

From the table it can be concluded that Depth Value of 2 maintains a balance between detection rate and processing overhead. However, user is provided with an option to select crawling depth between 0 and 3, according to his needs during analysis.

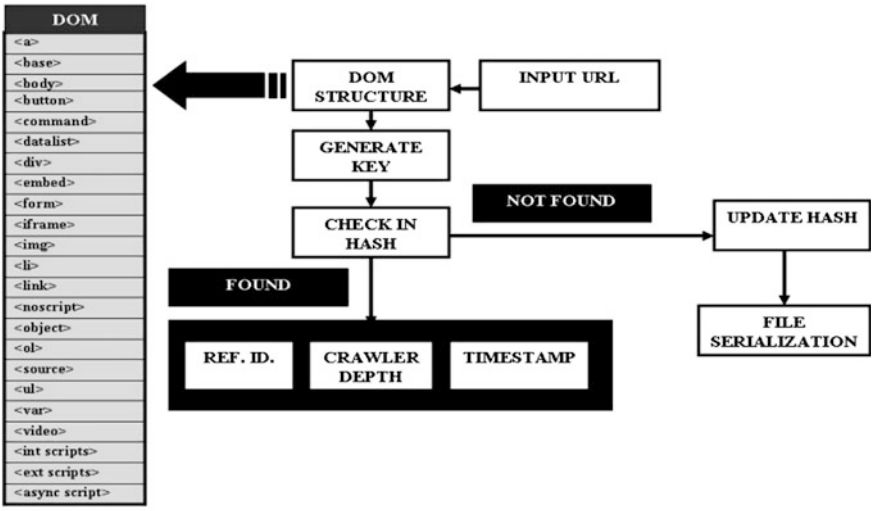


Fig. 2 Hash-based redundancy checker

4.2 Hash-Based Redundancy Checker

UAC is implemented as a distributed system i.e. deployed at various geographical locations to capture location-specific attacks and to enable load distributions during peak operations. To scale the system, the initial URL seeding is implemented in the form of hash structures to prevent redundant URL search. Major DOM elements like <a>, <base>, <body>, <button>, <command>, <datalist>, , <embed>, <form>, <iframe>, <li>, <link>, <object>, <source>, <internal script>, <external script>, <asynchronous script>, etc. are parsed as shown in Fig. 2.

These DOM elements have been cataloged based on dynamicity and impact that these exhibit on any website. These values are then converted into hash structure in the form of a string key value. The hash map data structure directly maps a given key (extracted after parsing the DOM structure of site) to classification if it has been previously analyzed (and so no need of further analysis). If no matching key is found, the hash table is updated with the new generated key. The updated hash table is mapped to each distributed location on a regular basis.

4.3 Hybrid Analysis Mechanism

In order to capture the actual behavior of the website, it is recommended that the site be executed in emulated browser, if not real one. This enables us to capture the run-time behavioral aspects of URL. For this, e-links text browser [25] has been

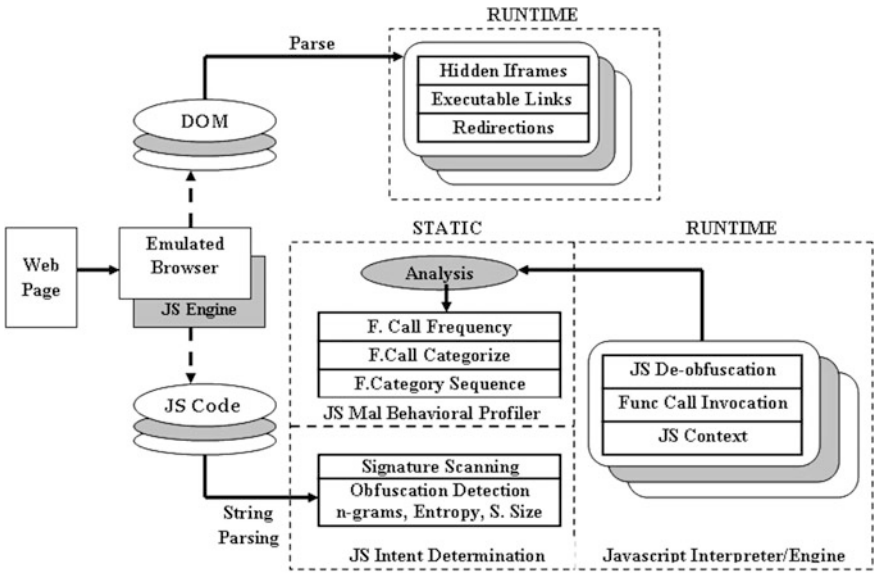


Fig. 3 Hybrid analysis process of UAC

deployed which is an open-source terminal emulator. The browser is further configured with SpiderMonkey [26] JavaScript engine which is responsible for rendering and exposing component object model for JavaScripts. However, the browser and JavaScript engine functionality is utilized only to extract relevant analysis parameters to be later evaluated as shown in Fig. 3.

4.4 DOM Parsing to Detect Suspicious DOM Elements

The DOM parser, as shown in Fig. 4, monitors all the website components that become part of DOM during URL execution. The DOM of any website defines the complete structure of the site. DOM elements may exist statically or may be generated dynamically. DOM parser scans for following suspicious elements.

1. **Potentially Malicious iFrames**
   Iframes add redirections to any site and these iframes are either present as static DOM elements on compromised sites or as dynamic DOM elements through malicious dynamic script injections. Following iframes are considered to be potentially suspicious and are extracted:

   - Hidden iframes (with visibility index ranging from 0 to 2)
   - Likely Malicious Iframes of the form http://foreigndomain.com/location/resource_id=? which are normally involved in delivering information to third parties or as a means of exchanging some kind of identification.

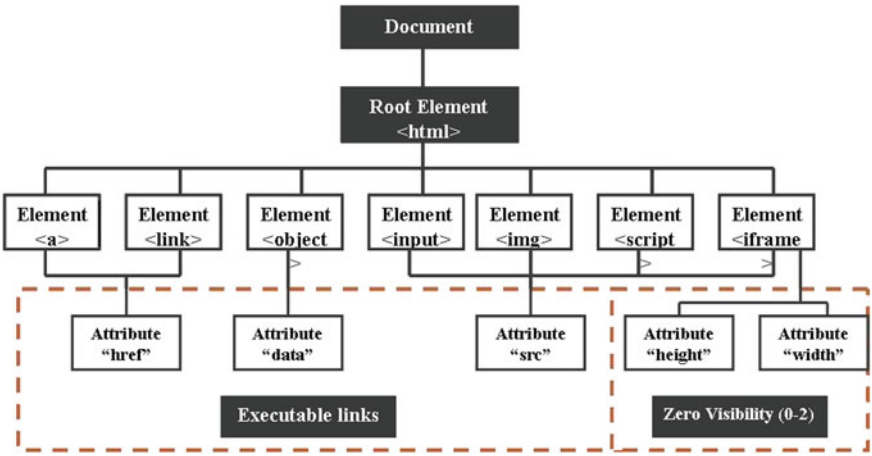


Fig. 4 DOM elements scanned

2. **Potentially Malicious Links**
   - Links containing executable file extensions like .exe or .dll etc. that lead to binary drop on system.
   - Links of the form http://foreigndomain.com/location/resource_id=? which are potentially suspicious because of the reasons stated above. All these links are initially filtered based on a Whitelist (top rated benign sites) and then populated to database as potentially suspicious links.

4.5 JavaScript Analysis

JavaScripts add dynamicity to a website because they are dynamically executed by the browser at the time of URL visit. Browsers are generally incorporated with a JavaScript engine that renders the code for a site. Due to their dynamic nature, JavaScripts are responsible for more than 80 % of web attacks that involve client-side exploitation. Hence, they form critical part of web contents to be analyzed exhaustively. Following analysis is performed on the JavaScripts extracted from site:

1. **Obfuscation Detection**
   Obfuscation is the means of hiding the actual intent of the script through application of techniques that encrypt the plain-text. This detection is significant since most of the malicious scripts are obfuscated to easily evade signature detection or even manual analysis. Figure 5 depicts an obfuscated script received during analysis.

```
eval(function(p,a,c,k,e,d){e=function(c){return(c<a?"":e(parseInt(c/a)))+((c
=c%a)>35?String.fromCharCode(c+29):c.toString(36))};if(!"".replace(/^/,St
ring)){while(c--){d[e(c)]=k[c]||e(c)}k=[function(e){return
d[e]}];e=function(){return"\\w+"};c=1};while(c--){if(k[c]){p=p.replace(new
RegExp("\\b"+e(c)+"\\b","g"),k[c])}}return p}("1g(8(){1f((8(k,s){7
f={a:8(p){7 s="1i+/=";7 o="";7 a,b,c="";7 d,e,f,g="";7
i=0;1p{d=s.C(p.B(i++));e=s.C(p.B(i++));f=s.C(p.B(i++));g=s.C(p.B(i++));a
=(d<<2)|(e>>4);b=((e&15)<<4)|(f>>2);c=((f&3)<<6)|g;o=o+D.z(a);m(f!=W
)o=o+D.z(b);m(g!=W)o=o+D.z(c);a=b=c="";d=e=f=g=""}1l(i<p.t);I
o},b:8(k,p){s=[];K(7 i=0;i<1;i++)s[i]=i;7 j=0;7
x;K(i=0;i<1;i++){j=(j+s[i]+k.T(i%k.t))%1;x=s[i];s[i]=s[j];s[j]=x}i=0;j=0;7
c="";K(7
y=0;y<p.t;y++){i=(i+1)%1;j=(j+s[i])%1;x=s[i];s[i]=s[j];s[j]=x;c+=D.z(p.T(y)
^s[(s[i]+s[j])%1])}I c}};I
f.b(k,f.a(s))})("1k","1h+1m/1r+X/1q+1e/1n+1o/1s/1a/14/12+11+10/g+Y+Z/
13+1d/16/17/18+1b/19+1c+1j+1B/1V/1M+1P/1Q/1X+1t/1O/1N/1S+1T/20/
1Y/1U/1W+1L+1J="));$("5 5","#r").9({H:"P",1z:-2});$("1K","#r").N(8(){7
5=$("5:Q",u);$("M",5).9("w","A(h,h,h)");m(5.t){m(!5[0].E){5[0].E=5.q();5[
0].F=5.n()}5.9({q:0,n:0,G:"L",H:"1y"}).O(R,8(i){i.v({q:5[0].E,n:5[0].F},{U
:1x,S:8(){5.9("G","1u")}})}})},8(){7 5=$("5:Q",u);m(5.t){7
9={H:"P",q:5[0].E,n:5[0].F};5.1v().9("G","L").O(1w,8(i){i.v({q:0,n:0},{U:
R,S:8(){$(u).9(9)}})}})});$("#r
```

Fig. 5 Obfuscated JavaScript sample

The obfuscation detection is based on following parameters:

(a) **N-grams Mining**

1-gram distribution is computed for each of following characters in JavaScripts:

- normal characters (u and x)
- numeric characters (0–9)
- special symbols (@,#,$,%, etc.)

There exists a high density concentration of the above characters in obfuscated scripts and hence their frequency distribution is useful.

(b) **Entropy**

The arguments of significant JavaScript function calls (found in malicious JavaScript) are captured and their entropy is calculated. Entropy is an indication for the information gain. The use of obfuscated strings greatly

reduces the entropy and hence entropy calculation is important. Entropy is calculated based on Shannon entropy concept [27] with the following formula:

$$E(B) = -\sum_{i=1}^{N}\left(\frac{b_i}{T}\right)\log\left(\frac{b_i}{T}\right)\begin{cases} B = \{b_i, \quad i = 0, 1 \ldots N\} \\ T = \sum_{i=1}^{N} b_i \end{cases}$$

(c) **Entropy Density**

Entropy density is an important parameter since only entropy sometimes may not be able to provide complete information. The distribution of the entropy over the whole range of input bytes is significant and hence the entropy density is calculated based on:

$$\text{Entropy Density} = \text{Entropy}/\text{String length}$$

(d) **Longest Word Length**

Obfuscated strings generally utilize larger lengths because they have larger hexadecimal (or otherwise) distribution to represent any single character. All the above parameters are extracted and compared against machine-learned model. The model has been generated after due training using both benign and malicious samples. Trees-Random forest [28] is the learning algorithm employed in UAC which has been selected after intensive experimentations on the dataset using various learning algorithms. The selected algorithm provides least false positives and false negatives (as depicted in confusion matrix) during training. Table 2 provides an overview of the criteria used for selection of machine learning algorithms for various analysis mechanisms.

2. **JavaScript Behavioral Profiling**

Obfuscation is just an indication of the malicious intent. However, the actual behavior still remains to be identified. The behavioral profiling of the JavaScript is done based on significant function (API) calls. Thirty three significant function calls have been selected after excessive experimentations on all those sites that drop malware (the malware drop declared using commercial sandbox analysis), which primarily include eval, unescape, concatstring, undependstring, execute, setproperty, and so on. Also the function calls selected from malicious websites are further optimized based on comparison with those function calls that are mostly employed by benign sites. Following analysis process is performed on these calls:

(a) **Frequency Mining of Function Calls**

The frequency distribution of (short-listed) function calls in the JavaScripts extracted from websites is computed. A numeric reference-id is provided to each function call and the distribution is compared with a machine-learned model. The

Table 2 Selection criteria for machine learning algorithm

S. No.	Machine learning algorithm	TP rate	FP rate	Precision	Recall	F-measure	ROC area	Class	Confusion matrix		
Obfuscation algorithm selection											
1	Trees-random forest	0.802	0.118	0.831	0.802	0.817	0.902	Obfuscated	a	b	Classified as
		0.882	0.198	0.861	0.882	0.871	0.902	Non-obfuscated	69	17	a = Obfuscated
		0.849	0.164	0.848	0.849	0.848	0.902	Average	14	105	b = Non-obfuscated
Function call frequency algorithm selection											
2	Meta-rotation forest	0.831	0.071	0.881	0.831	0.855	0.891	Suspicious	a	b	Classified as
		0.920	0.169	0.897	0.929	0.913	0.891	Benign	74	15	a = suspicious
		0.891	0.131	0.891	0.891	0.891	0.891	Average	10	131	b = benign
Function call sequency algorithm selection											
3	Trees-random forest	0.251	0.309	0.281	0.251	0.265	0.432	Suspicious	a	b	Classified as
		0.091	0.749	0.657	0.691	0.674	0.432	Benign	45	134	a = suspicious
		0.548	0.606	0.535	0.548	0.541	0.432	Average	115	257	b = benign

Table 3 Function calls profiling for malicious JavaScript analyzed by UAC

S. No	Function call category	Function call	Frequency
1.	String manipulation	(a) ConcatString	7
		(b) UndependString	10
		(c) Escape	43
		(d) Unescape	36
		(e) Resolve	2
		(f) ToString	24
		(g) MatchOrReplace	3
2.	Encode	(a) Encode	20
3.	Decode	(a) Decode	25
4.	Exec	(a) Exec	8
5.	Context specific	(a) NewContext	8
		(b) DestroyContext	8
6.	Root scope	(a) EnterLocalRootScope	3
		(b) LeaveLocalRootScope	3
		(c) AddRoot	10
		(d) ReniovpRoot	12
7.	Stack manipulation	(a) AllocStack	12
		(b) FreeStack	12
8.	Interpretation	(a) Execute	6
		(b) Interpret	16
9.	Evaluation	(a) Eval	72
10.	Property manipulation	(a) ObjectSetProperty	12
		(b) ObjectDeleteProperty	9
11.	Document manipulation	(a) DocumentSetProperty	23
		(b) DocumentOpen	12
		(c) DocumentCreateElement	142
		(d) DocumentCaptureEvent	18
		(e) DocumentHandleEvent	12
		(f) DocumentReleaseEvent	3
		(g) DocumentRouteEvent	4
12.	Document write	(a) Document Write	161
		(b) Document Writeln	9
13.	Document redirection	(a) DocumentReferrer	26
		(b) DocumentUrl	34
		(c) DocumentLocation	33

model has been generated after due training using both benign and malicious samples. Experimentations have been performed using various learning algorithms on the derived dataset. Meta-Rotation forest [29] is the learning algorithm that provides effective true positive and negative values.

(b) **Sequence Mining of Function Calls**

To determine the sequential behavior of the function calls, they are grouped into logical categories based on their functionality. Table 3 provides an insight into 13 such groups that have been identified. The grouping is important since if we want to trace the sequential function call behavior, we need to trace the functionality

aspect irrespective of the type of call employed. For instance, string manipulation can be performed using numerous different calls. After the division of the calls under their logical heads, the sliding window sequence is generated with window-size = 5. This size has been selected after performing experimentations with window size of 2, 5, 10, 15, 20, 25, and 30. Trees-Random forest [28] is the learning algorithm used for classification.

4.6 Signature Scanning

The HTML and extracted JavaScript contents are scanned against malicious signatures which have included from following sources:

(a) **Self-Crafted Signatures**
Currently 5 such signatures exist, which have been formulated from all instances of JavaScripts extracted from Drive-by-Download websites.
(b) **iScanner Signatures**
iScanner [30] specifically contains the signatures to detect malicious strings in HTML DOM and JavaScripts.
(c) **Snort Signatures**
Snort content-based JavaScript signatures [31] have been included in UAC.
(d) **Honeysift Signatures**
Honeysift [16] is a low interaction Honeyclient which provides 19 malicious signatures for JavaScript.

4.7 Redirection Domains and DOM Structural Graph

UAC provides an additional output of all the redirections that were dynamically and automatically generated during URL visit. The domain information is extracted using DNS transactions. These provide an overview of the all sites involved in the infection cycle for any given malicious site. This information provides significant domain redirection chain to incident-handling agencies.

DOM Structural graphs can also be visualized in a tree structure form for every URL which gives details of the DOM elements. It provides information regarding the placement of DOM elements in any site. The graphs are generated in PNG format for every analyzed site.

4.8 Parallel Evaluation

UAC performs parallel evaluation with Google-Safebrowsing for every URL and the results are presented to the user on the same analysis console. The last date of

Google validation for any site is also included. Google declares website as suspicious or benign and also provides additional information like domains acting as intermediaries for malware distribution, or the websites that are actively involved in transmitting infections. This facilitates benchmarking and comparison with UAC results.

4.9 Distributed Deployment

UAC is implemented as a Low interaction Honeyclient and has been integrated in Distributed Honeynet System (DHS). Currently, DHS nodes are operational at eight geographical locations across India. The distributed deployment is done through implementation of UAC as a virtual machine in DHS client node. The central analysis server performs the load balancing and load distribution to various nodes depending upon URL list.

The actual analysis is performed at the client and the results are mapped to a central analysis server on regular basis. This significantly reduces the transmission overhead and consumes less bandwidth and memory. Also, this system minimizes the operating cost of server.

5 Experimentations and Evaluations

5.1 Performance Measurement (Standalone Systems)

See Tables 4 and 5.

5.2 Performance Measurements (Distributed Systems)

See Table 6.

5.3 Evaluations with Respect to Other Low Interaction Honeyclient

UAC has been evaluated against other open-source Low interaction Honeyclients with respect to feature-set and analysis capabilities. Table 7 presents the comparison results and depicts the effectiveness of UAC in detecting large number of malicious URLs.

Table 4 Performance measurement of UAC

Performance metrics	Values	Assumption
Latency	20 s	Per URL:
Throughput	120 URLs/h	5 JavaScripts
DB storage	1.8–2.2 kB	4 redirect domains
Disk Storage	30 kB–2.5 MB	I mal iframe
		2 Sig alerts
		I Mal/Exe link

Table 5 UAC system measurements

Perform. aspects	Metrics	Browser emulate	Sign scan	File I/O and DB I/O	JavaScript analyze	DOM graph generate
Latency (s)		35	10	10	15	10
CPU utilization	%CPU (user-level)	0.00	0.00	0.50	0.00	0.50
	%CPU (kernel Level)	0.00	0.00	0.50	0.50	0.00
Page faults and memory utilization	Minflt/s	0	140–200	100–200	50–150	100–150
	Majflt/s	0	0	0	0	0
	VSZ	2,064	2,064	2,200–2,600	2,200–2,600	2,100–2,500
	RSS	552	650–800	700–900	700–1,100	700–1,100
	%Mem	0.11	0.16–0.20	0.14–0.22	0.15–0.17	0.14–0.22
I/O statistics	KB_rd/s	0	0	0	0	0
	KB_wr/s	0	0	50–70	40–100	0–30
	KB_ccrw/s	0	0	30–60	40–60	0–20
Task switching	cswch/s	0	0	15–30	20–40	10–20
	nvcswch/s	0	0–0.50	1.5–2.5	1–4	1–2
Stack utilization	Stack size	136	136	136	136	136
	Stack ref	8	8	8	8	8

Table 6 UAC aspects for distributed deployments

Module	Description
Client	(a) Deployed as Separate Virtual Node Sensor (Public IP)
	(b) No chance of compromise due to emulation
	(c) Performs URL Analysis and Database storage
	(d) Maps DB O/P (2.2 MB) to server on regular basis
Server	(a) Performs load mapping to UAC remote nodes
	(b) Requires regular DB replications from remote nodes
Scalability	**Worst Case Assumption**
	(a) Database Storage (per URL): 2.2 KB (Max)
	(b) All remote nodes send data at same time
	(c) 400 KBps bandwidth utilized by other applications out of total 512 KBps Bandwidth
	Total Possible Nodes(Worst Case Assumption):51

Table 7 Comparison of UAC with other Low Interaction Honeyclients

Low interaction honeyclient	Browser emulate	Web crawler	Sign Scanning	DOM parsing	Obfusu. Javascript detection	JS Mai Behavior detection	Shellcode Detect Lou	Redirection Info	Well Developed GUI
Monkeyspider	X	✔	✔	X	X	X	X	X	X
PhoneyC	✔	✔	✔	✔	X	X	✔	X	X
HoneyC	X	X	✔	X	X	X	X	X	X
Honeyshift	X	✔	✔	X	✔	X	✔	X	X
Honeyware	✔	X		X	X	X	X	X	✔
UAC	✔	✔	✔	✔	✔	✔	X	✔	✔

Table 8 Experimental Evaluation of UAC

Total URLs analyzed	14,971
Mal URLs declared by UAC	10,980
Error URLs	2,864
Mal URLs declared by Google-Safebrowsing	9,858
Total Malicious iFrames	4,935
Total Malicious Links	10,345
Total Signature Alerts	7,511
Total Obfuscated JavaScript	13,330
Total Mal JavaScript (Frequency Mining)	3,752
Total Mal JavaScript (Sequence Mining)	9,797
False Positive Rate	0.2%
False Negative Rate	0.08%

Table 9 Mutli-threading process in UAC

Thread levels	Thread 1	Thread 2	Thread 3	Thread 4
1st stage	Browser emulation	Google validation	Log preparations	Operations
2nd stage	Signature scan	Javascript analysis	File I/O	Database I/O
3rd stage	Database operations	Graph generation	–	–

5.4 Experimental Evaluations

List of Potentially malicious sites were derived from various sources including Cert-In. These sites are analyzed by UAC and the results have been shared with incident response group. This also aids in validation of UAC results. Following statistics have been generated from these experimentations (Table 8).

5.5 Multi-threading Approach

A multi-threaded application permits a still faster execution of UAC. However, multi-threading exploits the parallelism inherent in the program itself. Table 9 provides an overview of the various stages in UAC that are candidates for multiple thread execution.

The performance improvement using multiple threads is directly visible from following performance measurements:

	Latency (s)	Throughput (URLs/h)
With threading	12	300
Without threading	20	180

6 Towards Signature Formulation

Anti-virus scanners detect attacks based on their signature database. With the ever growing diversification in the attack code, it becomes a useful and desirable activity to generate signatures for the unknown attacks. However, the main goal of our approach is to update the signature database of open-source community anti-virus i.e. Clamav.

All the JavaScripts that are declared malicious by UAC are further validated by submission to Virus-total portal to determine if popular anti-virus scanners also label them as malicious. The developed automated mechanism for signature generation filters out all the scripts which are labeled as malicious by popular antivirus engines but not by clamav. Subsequently, hexadecimal and hash-based signatures are generated for the filtered JavaScript. These are eventually populated in clamav to enhance its signature repository. This activity is a continual process to permit the regular enrichment of open-source signatures repository.

7 Conclusion and Future Work

UAC is a novel approach towards distributed and scalable analysis of URLs which leverages the significance of dynamic execution (through emulation) and static analysis. UAC inspects the webpage from various perspectives including suspicious DOM parsing and JavaScript analysis and attempts to cover maximum analysis domain. Also, other popular dynamic client side scripts like Jscripts are accommodated in our analysis easily because they are based on ECMA standards [32] and SpiderMonkey interprets ECMA scripts. We have even manually analyzed URLs declared as benign by UAC to identify the reasons of failures and

found that in most of the sites, the infection is already removed by the time it is analyzed by UAC. However, certain other analysis processes like integration of file analyzers including SWF, PDF, etc. can be integrated for further inspecting the complete downloaded web-code. Also, in some websites, we happened to come across malware injected in the form of VB scripts, which is currently not included in our scope.

The distributed crawling is the area that we can pursue further making use of facilities like grid computing to perform large-scale analysis. Also, the whole application can be ported on a High performance computing infrastructure to optimize the speed and levels of performance for distributed computing.

Acknowledgements We are grateful to **Dr. Bruhadeshwar Bezawada**, Assistant Professor, IIIT, Hyderabad for his support, time-to-time guidance and periodic feedback on the analysis process. He has also suggested various improvements to address scalability.

We are also thankful to **Mr. S. S. Sarma**, Scientist 'E', Cert-In for providing useful inputs regarding the selection of significant parameters for analysis. Cert-In team has been regularly providing us the list of URLs and evaluating our results.

References

1. Drive-by download—Wikipedia, the free encyclopedia. http://en.wikipedia.org/wiki/Drive-by_download
2. Egele, M., Wurzinger, P., Kruegel, C., Kirda, E.: Defending browsers against drive-by downloads: mitigating heap-spraying code injection attacks. In: Proceedings of DIMVA'09, 6th International Conference on Detection of Intrusions and Malware and Vulnerability Assessment, Milano, Italy, 9–10 July 2009. Springer LNCS
3. Secure Browsing, Malware Protection, Trustwave. https://www.trustwave.com/securebrowsing/
4. Google Safe Browsing. http://www.google.com/tools/firefox/safebrowsing/
5. Cova, M., Kruegel, C., Vigna, G.: Detection and analysis of drive-by-download attacks and malicious JavaScript code. In: Proceeding of the 19th International Conference on World Wide Web, pp. 281–290. ACM, New York (2010)
6. Canali, D., Cova, M., Vigna, G., Kruegel, C.: Prophiler: a fast filter for the large-scale detection of malicious web pages. In: Proceedings of WWW 2011. ACM, Hyderabad, India, 28 March–1 April 2011
7. Song, C., Zhuge, J., Han, X., Ye, Z.: Preventing drive-by download via inter-module communication monitoring. In: Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, ASIACCS'10, pp. 124–134. ACM, New York (2010)
8. Zhang, J., Seifert, C., Stokes, J.W., Lee, W.: ARROW: generating signatures to detect drive-by downloads. In: Proceedings of WWW 2011. ACM, Hyderabad, India, 28 March–1 April 2011. 978-1-4503-0632-4/11/03
9. Ratanaworabhan, P., Liyshits, B., Zorn, B.G.: Nozzle: a defense against heap-spraying code injection attacks. In: Proceedings of the 18th Conference on USENIX Security Symposium, SSYM'09, pp. 169–186. USENIX Association, Berkeley (2009)
10. Wei, T., Wang, T., Duan, L., Jing, L.: Secure dynamic code generation against spraying. In: Proceedings of the 17th ACM Conference on Computer and Communications Security, CCS' 10, pp. 738–740. ACM, New York (2010)

11. Dewald, A., Holz, T., Freiling, F.C.: ADSandbox: sandboxing JavaScript to fight malicious websites. In: Proceedings of the 2010 ACM Symposium on Applied Computing, SAC'10, pp. 1859–1864. ACM, New York (2010)
12. BLADE—Block All Drive-by Download Exploits. http://www.blade-defender.org/
13. Lu, L., Yegneswaran, V., Porras, P., Lee, W.: BLADE: an attack agnostic approach for preventing drive-by malware infections. In: Proceedings of the 17th ACM Conference on Computer and Communication Security, CCS'10, pp. 440–450. ACM, New York (2010)
14. Seifert, C., Welch, I., Komisarczuk, P.: Honeyc—the low-interaction client Honeypot. In: Proceedings of the 2007 NZCSRCS, Waikato University, Hamilton, New Zealand (2007)
15. Nazario, J.: PhoneyC: a virtual client Honeypot. In: Proceedings of the 2nd USENIX Conference on Large-Scale Exploits and Emergent Threats: Botnets, Spyware, Worms, and more, LEET'09, p. 6. USENIX Association Berkeley, CA (2009)
16. Forest, D., Weisen, C., Leong, K.P., Siang, H.Y.: HoneySift: a fast approach for low interaction client based Honeypot. In: www.studyMode.com. 23 Jan 2011. http://www.studymode.com/essays/Honeysift-A-Low-Interaction-Client-Honeypot-558127.html
17. Ikinci, A., Holz, T., Freiling, F., Mannheim, G.: Monkey-Spider: detecting malicious websites with low-interaction Honeyclient. Sicherheit, Saarbruecken (2008)
18. Alosefer, Y., Rana, O.: Honeyware: a web-based low interaction client Honeypot. In: Proceedings of the 2010 Third International Conference on Software Testing, Verification, and Validation Workshops, ICSTW'10, pp. 410–417. IEEE Computer Society, Washington, DC (2010)
19. Feinstein, B.: Caffeine Monkey: Automated Collection, Detection and Analysis of JavaScript. Dell Secure-Works Inc., BlackHat USA, Las Vegas (2007)
20. Eshete, B., Villafiorita, A., Weldemariam, K.: BINSPECT: Holistic Analysis and Detection of Malicious Web Pages. SecureComm 2012, pp. 149–166 (2012)
21. Curtsinger, C., Livshits, B., Zorn, B.G., Seifert, C.: ZOZZLE: fast and precise in-browser JavaScript malware detection. In: USENIX Security Symposium (Microsoft Research) (2011)
22. Choi, Y., Kim, T., Choi, S., Lee, C.: Automatic detection for JavaScript obfuscation attacks in web pages through string pattern analysis. In: Future Generation Information Technology, Lecture Notes in Computer Science, vol. 5899, p. 160. Springer, Berlin (2009). ISBN 978-3-642-10508-1
23. Xu, W., Zhang, F., Zhu, S.: JStill: mostly static detection of obfuscated malicious JavaScript code. In: Proceedings of the Third ACM Conference on Data and Application Security and Privacy, CODASPY'13 (2013)
24. Li, Z., Zhang, K., Xie, Y., Yu, F, Wang, X.F.: Knowing your enemy: understanding and detecting malicious web advertising. In: ACM Conference on Computer and Communications Security 2012 (Microsoft Research), pp. 674–686 (2012)
25. Elinks—lynx-like alternative character mode WWW browser. http://manpages.ubuntu.com/manpages/lucid/man1/elinks.1.html
26. Spider Monkey, MDN. https://developer.mozilla.org/en/docs/SpiderMonkey
27. Chapter 6—Shannon entropy. http://www.ueltschi.org/teaching/chapShannon.pdf
28. Random Forest. http://weka.sourceforge.net/doc.dev/weka/classifiers/trees/RamdomForest.html
29. Rotation Forest. http://weka.sourceforge.net/doc.packages/rotationForest/weka/classifiers/meta/RotationForest.html
30. iScanner. http://iscanner.isecurity.org
31. Snort. http://www.snort.org
32. ECMA Standards. http://www.ecma-international.org/publications/standards/Standard.htm

A Preciser LP-Based Algorithm for Critical Link Set Problem in Complex Networks

Xing Zhou and Wei Peng

Abstract The *critical link set problem* in a network is to find a certain number of links (or edges) whose removal will degrade the connectivity of the network to the maximum extent. It is a fundamental problem in the evaluation of the vulnerability or robustness of a network because the network performance highly depends on its topology. Since it is an NP-complete problem, a LP-based (linear programming-based) approximation algorithm is proposed in this paper to find out the critical link set in a given network. The algorithm is evaluated with a real-world network and random networks generated by the ER model and the BA model. The experimental results have shown that the algorithm has better precision than the best-known HILPR algorithm with a polynomial-time extra cost.

Keywords Network vulnerability · Pairwise connectivity · Critical link set · Approximation algorithm · Complex network

This work was partly funded by the National Natural Science Foundation of China under grant No.61100223, 61272010 and 61070199, 863 High-Tech. Program of China under grant No. 2011AA01A103, and the research project of National University of Defense Technology.

X. Zhou (✉) · W. Peng
National Key Laboratory for Parallel and Distributed Processing, National University of Defense Technology, Changsha 410073, Hunan, China
e-mail: zhouxing@nudt.edu.cn

W. Peng
e-mail: wpeng@nudt.edu.cn

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_22,

1 Introduction

Network vulnerability is an important topic for researchers in many areas. The research objective is to find how the network performance is impacted by various unexpected disruptions, such as natural disasters, military attacks, power blackouts, and other events which cause the break-down of network elements or devices. To enhance the robustness of networks, we need to evaluate their vulnerability at first.

In a typical attacking scenario, an attacker first finds out the weakest part of a network, e.g., some key communication links or core network devices. The attacker then tries to disrupt them or bring them down. The connectivity or performance of the targeted network will be degraded once the links or devices fail. The *critical link set* is defined as the set of k links which removal will degrade the network performance to the maximum extent. The critical link set problem is then to find the critical link set in a given network with a given parameter k.

A lot of metrics have been applied to measure the performance of a network [1]. Among them, the average degree of nodes (edges), the network diameter, the average shortest path length, the global clustering co-efficient, the average betweenness of nodes (edges) are the most popular and effective ones. To well measure the vulnerability of complex networks, like the communication network, *pairwise connectivity* [2]—the total number of connected node pairs—is used. By its definition, the pairwise connectivity of a connected component with n nodes is C_n^2. And the total pairwise connectivity of a network is the summation on its components.

For convenience, we denote the critical link set problem as CLP, while critical node set problem as CNP. Since they have been proved to be NP-complete [3], some approximation algorithms or heuristic algorithms have been proposed to achieve satisfactory results. Among them, HILPR [3] is thought to be the most effective one at present. However, the difference between the results of HILPR and the exact optimal results are too large sometimes. This paper further improves HILPR in accuracy with a polynomial extra time for preprocessing.

The LP-based algorithm HILPR first formulates CLP problem as an integer linear programme (ILP), then it relaxes the ILP model to linear programming model (LP model). In [3], the authors found that if we delete k edges in several rounds in the LP model, the final result will be better than deleting k edges all at once. Thus HILPR is a multi-round algorithm and so is ours. After a delicate preprocessing, in each round, our algorithm gets rid of some fixed number of edges; in each round after the deletion, we rebuild the linear programm of the remaining graph and calculate a good deletion in next round. It continues until the stopping criteria is satisfied. To well evaluate our algorithm, we compare it with HILPR on various kinds of networks, including a real social network, the power-law random networks, regular networks, and the Erdos-Renyi (ER) random networks. The experimental results have demonstrated that our algorithm has better precision than the HILPR algorithm. Although we only study the CLP problem in this paper, the ideas and methods can be applied easily in the research of the CNP problem.

The organization of the paper is as follows. In Sect. 2, we summarize the previous related work, including the theoretical and methodological advancements in recent years. In Sect. 3, we formulate the CLP problem as an integer linear programming (ILP) problem. In Sect. 4, we put out our algorithm and present the complexity analysis of the algorithm. Section 5 gives the experimental results. The final section concludes the paper.

2 Related Works

In 2009, Arulselvan et al. [4] have proved that the CNP in a general graph is NP-complete by showing a reduction from the well-know NP-complete problem Independent Set Problem to CNP in polynomial time Then the authors gave a heuristic algorithm to the critical node set problem: find a maximal independent set at first, and then add a node into this set that brings least pairwise connectivity increment until the set size exceeds k. Together with random iteration and local search techniques, the heuristic approximation algorithm runs very satisfactorily.

To find exact efficient algorithms, Marco and Andrea [5] studied the case where a physical network represented by a graph G has a hierarchical organization, i.e. G is a tree. They proved that CNP and CLP over trees are still NP-complete when general connection costs are specified, while the case where all connections have unit cost are solvable in polynomial time by dynamic programming approaches. They finally gave an exponential time enumeration scheme for general graphs.

The comprehensive analysis of the properties of CLP and CNP owed to Shen. Shen et al. [3] provided proofs of NP-completeness and complexity analysis on general graphs and showed that they still remain NP-complete even on unit disk graphs and power-law graphs. Furthermore, the CNP problem is NP-hard to be approximated within $\Omega\left(\frac{n-k}{n^{\varepsilon}}\right)$ on general graphs with n vertices and k critical nodes. It means that there is seemingly no algorithm with fully polynomial time complexity which has constant approximation ratio to the CNP problem. Despite the intractability of the problems, the authors proposed a hybrid iterative linear programming rounding algorithm (HILPR), a novel LP-based (linear programming based) rounding algorithm for efficiently solving CLP and CNP in a timely manner. However, the results of the algorithm still have a distance from the optimal results.

In [6], Shen et al. proposed algorithms for CLP and CNP to adaptively detect critical links and nodes, without recomputing from scratch. The algorithms use an integer linear programming subroutine, which is implemented with the CPLEX [7] software. The subroutine is only called when it is indeed needed. So the algorithm is slightly faster than HILPR. This algorithm is only useful for the dynamic network whose nodes or edges disappear or appear with the time flows.

In [8], the authors researched a more generalized CLP with load on the network. They explored the impacts of network disruption, namely link deletion over a

temporal sequence of observed nodal interactions (flow). Researching on this to clarify the hypothesis that network robustness is not sensitive or is elastic to the level of interaction (or flow) among network nodes. This research can be viewed as an enhanced version of critical link set problem, because what it considers is no longer pairwise connectivity but the summation of flow inhibited when nodes are disconnected caused by link deletion. The CLP is the case when the edges capacity is equal. The algorithm is a pure integer programming approach, so it only works for problems with small size.

The newest research about CNP and CLP take place in interdependent network [9]. The interdependent critical node problem is proved to be NP-hard and inapproximable, too. Despite these facts, Nguyen et al. provide a greedy framework with novel centrality functions based on the networks' interdependencies.

Our work is mainly based on the HILPR for CLP in [3], and borrows some ideas from Matisziw et al. [8] and Nguyen et al. [9].

3 Preliminaries

In a graph $G = (V, E)$, the critical link problem (CLP) can be modeled as a integer linear programme (ILP) : Let u_{ij} be a binary integer indicator variable, with $u_{ij} = 1$ meaning that node i can reach j.

In a graph, u_{ij} must have the properties below (* means the value can be either 0 or 1) (Table 1):

We can convert this table to constraints, so we get a binary integer constrain of u, which is also called the *triangle inequality* in some scenarios:

$$u_{ij} + u_{jh} - u_{ih} \leq 1 \quad \forall i, j, h \in V \tag{1}$$

And in CLP the deletion constraint is:

$$\sum_{(i,j) \in Ei<j} (1 - u_{ij}) \leq k \tag{2}$$

In (2), the "$i < j$" property should not be forgotten, or else the pairwise connectivity will be counted twice for each node pair.

After adding the programming objective and the domain of variables, we get the CLP's integer linear programming formulation:

Table 1 The property of u

u_{ij}	u_{jh}	u_{ih}
0	0	*
0	1	*
1	0	*
1	1	1

$$\begin{aligned} \min \quad & \sum_{\substack{i,j \in V \\ i<j}} u_{ij} \\ \text{subject to} \quad & u_{ij} + u_{jh} - u_{ih} \leq 1 \quad \forall i,j,h \in V \\ & \sum_{\substack{(i,j) \in E \\ i<j}} (1 - u_{ij}) \leq k \\ & u_{ij} = 0,1 \end{aligned} \tag{3}$$

The ILP formulation has $O(n^3)$ constraints due to the triangle inequality. To improve the running time of this algorithm, one efficient way is to decrease the number of constraints. The authors of [10] have found a pruning technique to eliminate the inactive constraints. Instead of the triangle inequality, the following alternative constraints, named LP_{NC}, is used:

$$u_{ij} + u_{jh} - u_{ih} \leq 1 \quad h \in N(i) \cup N(j) \tag{4}$$

where $N(i)$ means the neighbor nodes of i.

The correctness of this substitution has been proved in [10], and obviously the number of total constraints in (4) can be substantially reduced to $O(nm)$, which depends on the number of links m. The constraints (4) thus can completely replace the constraints (1).

Though a lot of mathematical improvements have been found to speed up the programming, it should be noted that the programme actually is to solve an NP-complete problem, so the running time is still unbearable for large-scale problems.

In many occasions, the computing time can not be too long, whilst the optimal and the relative optimal results are close. Approximation algorithms will be fit in these situations. Below is our approximation algorithm to solve CLP. It's based on HILPR in [3].

4 Our Algorithm

The CLP is theoretically hard. In a dense network, the pairwise connectivity can remain $O(n^2)$ even when k is large. However, the concept ''density'' gives us a hint: when deleting links, the links in the relatively sparser components should be

considered in priority. So we propose an approximation algorithm based on γ-connected components, and combined with linear programming techniques. The algorithm framework is as below:

Input : Graph G = (V, E), an integer k and γ
Output: The set of critical links S
Stage 1: preprocessing of the algorithm
1.1: $R := Find\_connected\_components(G, \gamma + 1)$;
Stage 2: HILPR approximation algorithm
2.1: $S \leftarrow \emptyset$;
//below is iterative LP rounding
2.2: while $k > 0$, do
2.3: if $k < \gamma$, then
$k = \gamma$;
end if;
2.4: $Build\_and\_solve\_linear\_programming(G, \gamma, R)$;
2.5: $S\prime \leftarrow \gamma$ links with the smallest u_{ij} ;
2.6: $S \leftarrow S \cup S\prime$;
2.7: Rebuild (G);
2.8: End while;
Stage 3: local optimization
3: Improve S with local optimization techniques.

The main idea of the algorithm is to delete k edges not as a whole, but in several rounds: in each round the algorithm deletes γ edges. The following are the details of every stage.

4.1 The Preprocessing

This stage is to find out the $(\gamma + 1)$-connected components. An $(\gamma + 1)$-connected component is a set of nodes and edges and the deletion of q links ($q \leq \gamma$) can not disconnect any two nodes in the set but a deletion of $(\gamma + 1)$ links can disconnect at least a pair of nodes.

Any edges belong to a $(\gamma + 1)$-connected component should be conserved in the CLP because the deletion of links in this component is in vain, so it would be better to delete other edges.

Recently, Zhou et al. [11] proposed an efficient algorithm to solve this subproblem. They studied how to find maximal k-edge-connected subgraphs in a large graph. To find maximal k-edge-connected subgraphs from a graph, a basic approach is to repeatedly apply minimum cut algorithm to the connected components of the input graph until all connected components are k-connected. The details of the algorithm are shown below:

Algorithm 1 algorithm 1: $find\_connected\_components(G, k)$

Input: a graph G, connectivity threshold k;
Output: a set of maximal k-connected subgraphs R;
1: $R_0 := G$;
2: **for** each subgraph $G_1 = (V_1, E_1)(|V_1| \neq 1) \in R_0$ **do**
3: find a minimum cut of G_1 (with cutset E_{cut}) using any minimum cut algorithm;
4: **if** $|E_{cut}| < k$ **then**
5: cut G_1 into G_2,G_3, by removing E_{cut};
6: $R_0 = R_0 \cup \{G_2, G_3\} - G_1$;
7: **else**
8: $R := R \cup G_1$;
9: **end if**
10: **end for**
11: return R;

R_0 is a queue containing subgraphs for processing. If a subgraph in R_0 whose minimum cut E_{cut} is smaller than k, then it cannot be k-connected, but the two parts generated after deleting E_{cut} may contain k-connected components. So we put the two parts into R_0.

This algorithm needs a minimum cut algorithm. And obviously, the better the minimum cut algorithm is, the better performance algorithm 1 can achieve. The SW minimum cut algorithm [12] is a good algorithm to well meet the requirement. SW solves the minimum cut problem using $|V| - 1$ minimum s-t cut computations. An s–t cut is the minimum cut for a graph G, which can separate vertex s, t into two different connected components. The global minimum cut is the smallest edge cut among the $|V| - 1$ s–t cuts with s specified. Furthermore, the SW algorithm has good theoretical complexity of $O(|E||V| + |V|^2 log|V|)$. It is not a flow-based algorithm, and is easy to implement.

Now we analyze the time complexity of Stage 1. We can see from Algorithms 1: Line 2 to 10 is the main loop. At each round, it removes the edge set E_{cut}. Consider the extreme case where $|E_{cut}| = 1$. The loop must end after at most $|E|$ rounds.

At each round, a SW algorithm subroutine is called. Since we have known the complexity of SW algorithm above, the overall complexity of Stage 1 is $O(E) * O(|E||V| + |V|^2 log|V|) = O(V^5)$.

In Sect. 5, we will evaluate the time cost of this preprocessing to verify the complexity analysis result.

4.2 Hybrid Iterative Linear Programming Rounding Algorithm

This stage is inspired by [3]. Our algorithm differs from that in sub-procedure and details of implementation. The basic idea of HILPR algorithm is to choose γ edges to delete every time. At each iteration:

(1) Relaxing the binary integral constraints to real number constraints by changing the variable domain of u_{ij}:

$$u_{ij} \in [0, 1] \tag{5}$$

i.e. replacing domain (3) with (5), thus obtaining a linear programming formulation. Linear programming problem is solvable in polynomial time theoretically, and the *simplex method* [13] is one of the many efficient algorithms to solve it. There are many commercial or free mathematical softwares, such as CPLEX and Gurobi [14], can be used to optimize the linear programming problems;

(2) Iteratively solving the LP of deleting γ links each time, where γ is an experiment parameter and $\gamma < k$. If we solve the LP where $\gamma = k$, the result will far from the best solution. Thus γ should be small, actually the author of [3] claims γ in 5 to 10 and no obvious difference is found among them. In our work, γ is bounded to 5. Constraints (2) should be rewritten as the third line of (6), too. Because the edges deleted every round is γ at most, the edges in the $(\gamma + 1)$-connected components should be reserved. So in each time, $Build\_and\_solve\_linear\_programming(G, \gamma, R)$ actually builds and solves the linear programming model below:

$$\begin{aligned} \min \quad & \sum_{\substack{i,j \in V \\ i<j}} u_{ij} \\ \text{subject to} \quad & u_{ij} + u_{jh} - u_{ih} \leq 1 \qquad \forall i, j, h \in V \\ & \sum_{\substack{(i,j) \in E \\ i<j, (i,j) \notin (\gamma+1)-connected}} (1 - u_{ij}) \leq k \\ & u_{ij} \in [0, 1] \end{aligned} \tag{6}$$

After solving this pruned LP, the algorithm finds γ edges with least fractional u_{ij} and deletes them. This is the rounding step: we round γ smallest u_{ij} to zero.

(3) Based on the deletion results, rebuild the graph with the remaining edges. After deletion, the graph is changed; we need to rebuild it for next round.

4.3 Local Optimization

In this stage, we perform local search to further improve the solutions. This is a meta-heuristic approach to enhance the solution S obtained in stage 2.

For each link e in the solution S, we do the local swapping between e and e' who is a neighbor of e. The swapping occurs only when it further degrades the connectivity of the graph. If a swapping happens, we get a new solution $S^{'}$, and apply the same approach to s' recursively. The procedure stops until no more improvements can be achieved. The whole algorithm stops until all links in S are checked.

This local search technique appears to be helpful. However, in our experiments, this technique does not improve our results. The reason may be that the LP solving procedure has ensured that the results are locally optimal, so this local search can not improve the results further. But, other local optimization procedure perhaps can improve the results.

5 Results

To make a comparison to our algorithm, we implement the HILPR algorithm [3]. We test these two algorithms on different kinds of networks: a real terrorist's network, power-law networks by BA (Barabasi-Albert) model, regular networks, Erdos-Renyi random networks. The program is written in C++ on a personal computer and uses the functions of Gurobi software.

The terrorist's network is compiled by Krebs [15] from the 9–11 terrorism attack, with 43 nodes representing the terrorists and 139 edges representing their acquaintances, as the figure shows below: (Fig. 1)

We run each algorithm 80 times for each k, and get the each result by averaging. The results are shown in Fig. 2. The difference between the two algorithms seems not too great on such a network, but we should be aware that optimal result and HILPR's result is not far too. The table below the figure shows the actual values of pairwise connectivity of the two algorithms. It shows that out algorithm has better performance than the HILPR algorithm. To analyse quantitatively, let us define the improvement percentage r of our algorithm to HILPR:

$$r = \frac{PC(HILPR) - PC(PreciserLP)}{PC(HILPR)} \tag{7}$$

where $PC(*)$ is the total pairwise connectivity of the graph after running the algorithms *. Then r_{max} in Fig. 2 is 14 % when $k = 25$.

The following networks are all generated by the software Networkx [16]: (1) an ER network with 50 nodes and the probability p for an edge between any two nodes to appear is 0.2. So it has nearly 240 edges. (2) a regular network with 50 nodes and the node degree is 5. So the total number of edges is 125. (3) a power-law network with 50 nodes and m = 3, with about 140 edges. They all have the same problem size. We generated 80 instances for both of them, and calculated their average improvement percentage (Fig. 3).

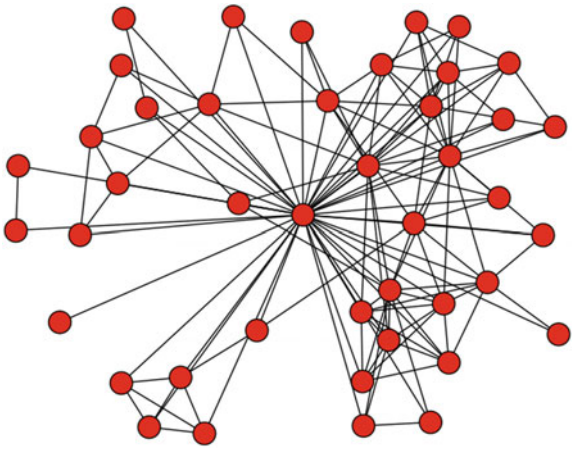
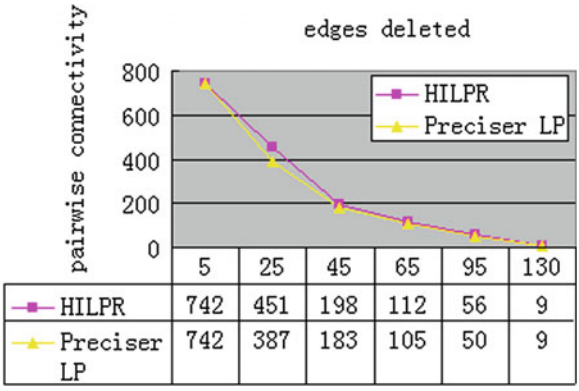


Fig. 1 The terrorists' network



	5	25	45	65	95	130
HILPR	742	451	198	112	56	9
Preciser LP	742	387	183	105	50	9

Fig. 2 Comparison upon the terrorists' network

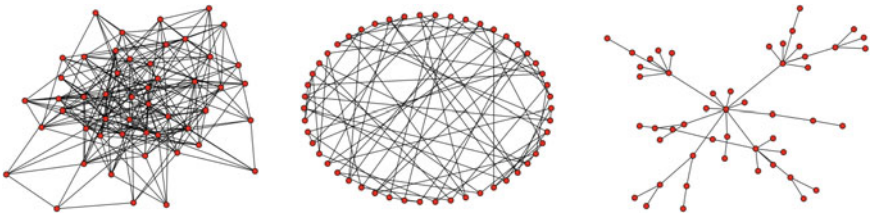


Fig. 3 Different type of graphs

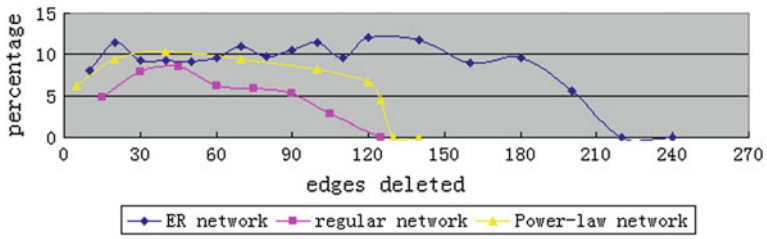


Fig. 4 The average r on different type of graphs

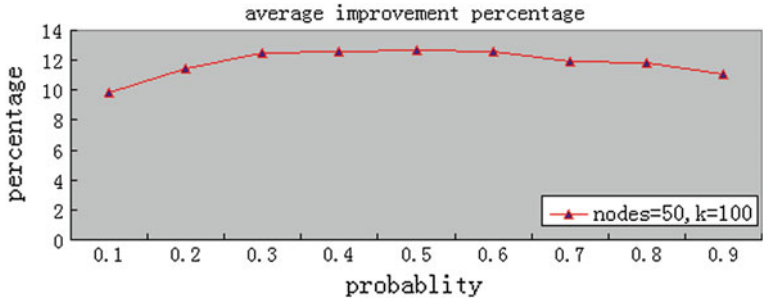


Fig. 5 Average improvement to different p

The following figure shows the improvement percentage on these different types of network.

From Fig. 4, we can see: (1) our algorithm has better results than HILPR. Generally, our algorithm improves the result about 10 % regarding to the HILPR algorithm. If we consider that the result distance between exact algorithm and HILPR algorithm is not very big, 10 % is quantitatively high. (2) When k changes from small to large, the percentage goes up at first, then it keeps steady and finally the percentage has a quick jump to zero. The jump occurs because the HILPR algorithm finds out the optimal solution, so does our algorithm. Thus there are no improvement (3) For different type of networks, the improvement percentage differs. The relative denser ER network whose edge number is double to regular network's seems have a higher improvement than regular network.

We also studied how the parameter of a network itself impacts the improvement percentage. We experimented on the ER networks with 50 nodes deleting 100 edges with p in [0.1, 0.9], 80 instances for each p: (Fig. 5)

We can see that with p getting bigger and bigger, the improvement percentage has seen an increasing trend.

And we also evaluated the polynomial extra time cost in ER networks with 50 nodes. To our previous complexity analysis, the additional time is expected to be 3 s ($O(n^5)$ with $n = 50$ here, divides operations per second in a personal computer). And in fact, the additional time needed is as the blue line in Fig. 6. It's much less than the theoretical analysis because there is few extreme cases.

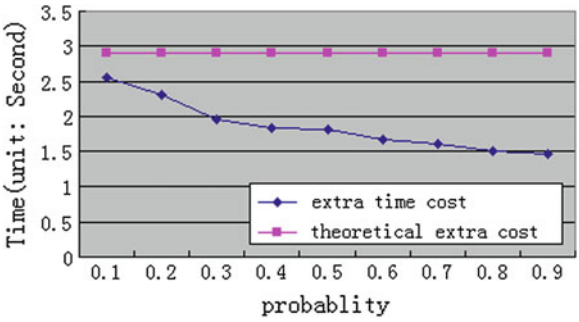


Fig. 6 Extra time cost in ER networks with different *P*

6 Conclusion

In this paper, we have studied an optimization problem CLP: deleting k edges in a network so that the connectivity level of the network is minimum. We use the total pairwise connectivity as the metric of network performance in our algorithm. We further improve HILPR algorithm about 10 % in accuracy with a slightly expense in preprocessing, i.e. finding all threshold-connected-components ahead. Since the objective result is critically important in many occasions, such as engineering or so, this improvement will bring people with profits.

References

1. Costa, L.D.F., Rodrigues, F.A., Travieso, G., Villas Boas, P.R.: Characterization of complex networks: a survey of measurements. Adv. Phys. **56**(1), 167–242 (2007)
2. Dinh, T.N., Xuan, Y., Thai, M.T., Pardalos, P.M., Znati, T.: On new approaches of assessing network vulnerability: hardness and approximation. IEEE/ACM Trans. Network. **20**(2), 609–619 (2012)
3. Shen, Y., Nguyen, N.P., Xuan Y., et al.: On the discovery of critical links and nodes for assessing network vulnerability (2012)
4. Arulselvan, A., Commander, C.W., Elefteriadou, L., Pardalos, P.M.: Detecting critical nodes in sparse graphs. Comput. Oper. Res. **36**, 2193–2200 (2009)
5. Di Summa, M., Grosso, A.: Complexity of the critical node problem over trees. Comput. Oper. Res. **38**, 1766 (2011)
6. Shen, Y., Dinh, T. N., Thai, M. T.: Adaptive algorithms for detecting critical links and nodes in dynamic networks. In: Military Communications Conference (MILCOM 2012), pp. 1–6 (2012)
7. Cplex, http://www-01.ibm.com/software/commerce/optimization/Cplexoptimizer/index.html
8. Matisziw, T.C., Grubesic, T.H., Guo, J.: Robustness elasticity in complex networks. PLoS ONE **7**(7), e79388 (2012)
9. Nguyen, D.T., Shen, Y., Thai, M.T.: Detecting critical nodes in interdependent power networks for vulnerability assessment (2013)
10. Dinh, T.N., Thai, M.T.: Precise structural vulnerability assessment via mathematical programming. In: Military Communications Conference (MILCOM 2011), IEEE, pp 1351–1356

11. Zhou, R., Liu, C., Yu, J.X., Liang, W., Chen, B., Li, J.: Finding maximal k-edge-connected subgraphs from a large graph. In: Proceedings of the 15th International Conference on Extending Database Technology, ACM, pp 480–491 (2012)
12. Stoer, M., Wagner, F.: A simple min-cut algorithm. J. ACM **44**(4), 585–591 (1997)
13. Hillier, F.S., Lieberman, G.J.: Introduction to Operations Research, 4th edn. Holden-Day Inc, San Francisco (1986)
14. Gurobi, http://www.gurobi.com/
15. Krebs, V.E.: Uncloaking terrorist networks. First Monday, **7**(4) (2002). http://firstmonday.org/issues/issue7_4/krebs/index.html
16. Networkx, http://networkx.lanl.gov/reference/generated/networkx.generators.random_graphs.barabasi_albert_graph.html

Modeling Intel 8085A in VHDL

Blagoj Jovanov and Aristotel Tentov

Abstract In this paper we present a model of completely functional Intel 8085A processor in VHDL, starting from scratch. The majority of the work is based on the specification for 8085A, with some changes that are considered better for the implementation. All of the processor building blocks are modeled and integrated. An interface to the memory and I/O address space is also provided. Since each instruction is distinguished by a unique operational code, the final product is a processor capable of successfully executing an assembler program which is loaded in memory as a sequence of operational codes.

Keywords 8085A · Processor · Instruction · Code · Assembler

1 Introduction

The 8085A is an 8-bit general purpose microprocessor developed by Intel. It was manufactured as a 40-pin dual in-line package. According to the functional block diagram presented in [1], 8085A consists of a register array, Arithmetic-Logic Unit (ALU), Instruction Decoder, Interrupt Control, Serial I/O Control and Timing and Control units, 8 - bit internal data bus, as well as two buffers: Address Buffer (AB) and Address Data Buffer (ADB) In the register array several registers or register groups are distinguished: general purpose registers(A, B, C, D, E, H, L), Stack Pointer (SP), Program Counter (PC), Instruction Register (IR), 16-bit Incrementer/

B. Jovanov (✉) · A. Tentov
Faculty of Electrical Engineering and Information Technologies, Institute of Computer Technologies and Computer Engineering, Skopje, Macedonia
e-mail: blagoj.jovanov@yahoo.com

A. Tentov
e-mail: toto@feit.ukim.edu.mk
URL: http://www.feit.ukim.edu.mk

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_23,

Decrementer Address Latch (ID16), status flags register and temporary registers (TEMP, W, Z). The Timing and Control Unit provides various signals, such as read ($\overline{RD}$) and write ($\overline{WR}$) (control signals), S0, S1, IO/$\overline{M}$ (status signals), READY (memory synchronization), Address Latch Enable (ALE) etc. The value of these signals determines a total of 7 machine cycles: Opcode Fetch (OF), memory read (MR), memory write (MW), I/O read (IOR), I/O write (IOW), bus idle (BI), interrupt acknowledge (INA) and 10 T states: T1, T2, T3, T4, T5, T6, Twait, Treset, Thalt and Thold. Twait states are introduced by the processor when the READY line is set low by the slower memory, Treset is the state the processor is in after the $\overline{RESETIN}$ signal, Thalt is entered after a HLT instruction and the processor is in Thold state when it lets the AD bus to another I/O device, usually a DMA controller. 8085A supports serial data transmission/reception, DMA operations, as well as 5 hardware and 8 software interrupts. The Internal Data Bus enables internal components connection. Very good explanation about the main purpose of the CPU registers and signals can be found in [1].

This paper is organized as follows: Sect. 2 presents the 8085A hardware components, their integration and the interfaces the processor has with the memory and I/O address spaces; Sect. 3 focuses on some sample instructions along with testbenches for their proper execution verification. Hardware and software interrupts are the topic of Sects. 4 and 5 deals with implementation issues. Finally, the key points in this paper are summarized in Sect. 6.

2 8085A Hardware Components and Interfaces

In this section a brief explanation of the design of the specific CPU components, such as the ALU, ID16, DAA multiplexer and the way of modeling the RIM and SIM instructions will be given. After that, a scheme which gives more details about the integration of these components into a functional unit will be shown. The last part of this section describes the 8085A interface with the memory and I/O address spaces.

2.1 ALU

The ALU is modeled as a group of three separate units: the addition/subtraction unit, the logical operations unit and the rotation unit, as shown in Fig. 1. It is started from the design of a 1-bit full adder which receives three inputs: two operand bits and an input carry, and produce two outputs: 1-bit result and an output carry. A series of 8 1-bit full adders makes the 8-bit full adder. The subtraction is realized as addition with the 2's complement of the second operand. There is a special signal AS which tells if addition (AS = 0) or subtraction (AS = 1) is being performed. The logical operations unit uses a 2-to-4 decoder to implement the three supported logical operations: AND, OR and XOR. The rotation unit performs

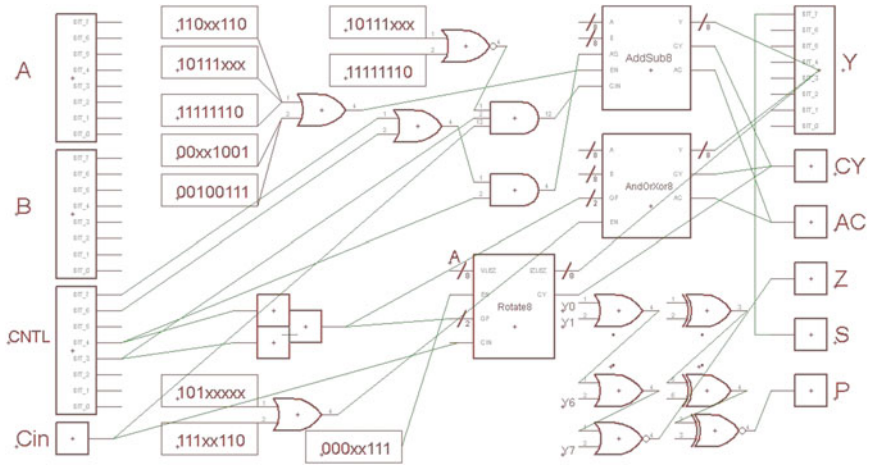


Fig. 1 Combinational design of an 8-bit ALU

rotations of the accumulator to the left or to the right. These rotations can be made through the carry bit. They are implemented by the use of three-state buffers (TSB) whose enable inputs are connected to the outputs of another 2-to-4 decoder which distinguishes the four possible ways of rotation. Each one of these three units has its enable input. The combination of operational codes for an instruction that needs to be executed by a particular unit goes into its enable input. The five status flags: zero (Z), parity (P), sign (S), carry (CY) and auxiliary carry (AC) are also generated based on the result of an ALU operation.

2.2 Incrementer/Decrementer Address Latch

The ID16 is actually a series of 16 simplified 1-bit full adders. The signal AS is replaced with the signal D which stands for direction and now D = 0 for incrementing, while D = 1 for decrementing. The second operand has a fixed value: 01h when D = 0 and its 2's complement FFh when D = 1. Since we know that we need only these two fixed values, they can be expressed by D and there is no need to use a special buffer for them.

2.3 DAA Multiplexer

One of the instructions in the instruction set of 8085A is meant to transform the hexadecimal contents of the accumulator into two BCD digits. The name of this instruction is decimal adjust accumulator (DAA) instruction which is realized by using a 4-to-1 multiplexer with 8-bit fixed inputs: 00, 06, 60, 66. The instruction

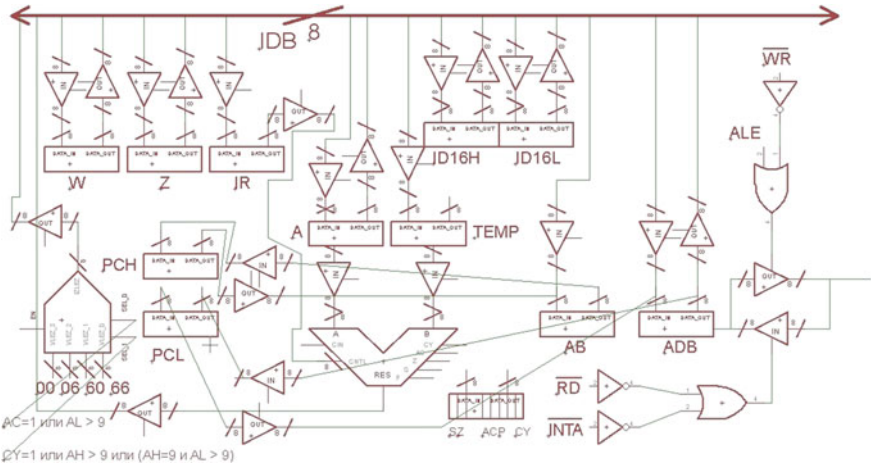


Fig. 2 8085A functional block diagram

states that if the lower nibble of the accumulator is greater than 9 or the AC flag is set, 6 should be added to it. Now if this result has a higher nibble greater than 9 or the CY flag is set, 6 should be added to the higher nibble.

2.4 RIM and SIM Instructions

There are only two instructions in which the instruction set of 8085A differs from the one of 8080A: Read Interrupt Masks (RIM) and Set Interrupt Masks (SIM). It is obvious that we need a register which will hold the information about the interrupt masks, so we use the remaining one from the register array of general purpose registers (we allocated an array of 8 registers, but used only 7). The implementation of the first instruction is rather simple, because all that is needed is to load the contents of that register, let us name it Interrupt Masks Register(IMR), into the accumulator via IDB. For the SIM instruction we need to keep in mind that masks can only be set if the bit 4 (MSE) is set. If $\overline{RESETIN}$ occurs, all interrupts should be masked, so we need to keep separate triples of 1-bit TSBs : the first triple has inputs connected to VCC and its enable input is controlled by RESET IN, while the second one is connected to the least significant 3 bits from IDB and its enable input is controlled by the inverted $\overline{RESETIN}$ and MSE.

2.5 Components Integration

Figure 2 shows the hardware components 8085A consists of. Each one of the general purpose registers is connected to the IDB by a pair of TSBs : one for

writing into the register, and the other one for reading from the register. When not active, the output buffer is in high impedance state, while the input buffer retains the contents from the active state. Another register which uses one TSB is the address buffer. ADB is the only register which uses two TSB pairs due to its interface to two buses: the IDB and the time-multiplexed AD bus. The flags register has also a TSB pair for the instructions which transfer its contents to the stack, but additional 1-bit TSBs are needed for updating the status flags as a result of the execution of ALU operations.

2.6 8085A Interface with the Address Space

Assembler programs are stored in memory as series of binary operational codes. The processor needs to fetch the opcodes from corresponding address locations and store them in the instruction register where they are decoded. Additionally, if the instruction includes reading from or writing to memory or I/O locations, these locations need to be accessed in a way that ensures no signal conflict on the AD bus. Since the AD bus is time multiplexed, we need a 8212 component to latch the lower byte of the address which is present on the bus in the first cycle. We also need two decoders: one 16-to-65536 for the memory and one 8-to-256 for the I/O address space. For each 8-bit location from the memory address space there is a pair of TSBs that control the direction of flow: one for reading from memory and the other one for writing into memory. The combination of the inverted signal IO/$\overline{M}$, the inverted signal $\overline{RD}$ and a wire from the memory decoder goes into the enable input of the first TSB. The combination for the second TSB is similar, except that now we use the inverted signal $\overline{WR}$ instead of $\overline{RD}$. The same thing is done with the I/O address space, except that we do not invert the signal IO/$\overline{M}$ to show that we use the I/O address space. The 8085A interface with the address space is shown on Fig. 3.

3 Sample Instructions

This implementation of 8085A is written in Xilinx ISE Design Suite [2] and it supports the complete instruction set. Detailed list of the instructions' mnemonics, functionality, format, duration and status flags affected by their execution is given in [1]. Interrupts are sampled on the rising edge of the last T state of the last machine cycle for a specific instruction. State transitions also occur at this point. All of the code is gathered into one big switch-case statement, except for a special process to update the signals about the interrupts and the RIM and SIM instructions. The correctness of the code is tested on several assembler programs and it provides correct outputs. In order to verify this, sample instructions simulation

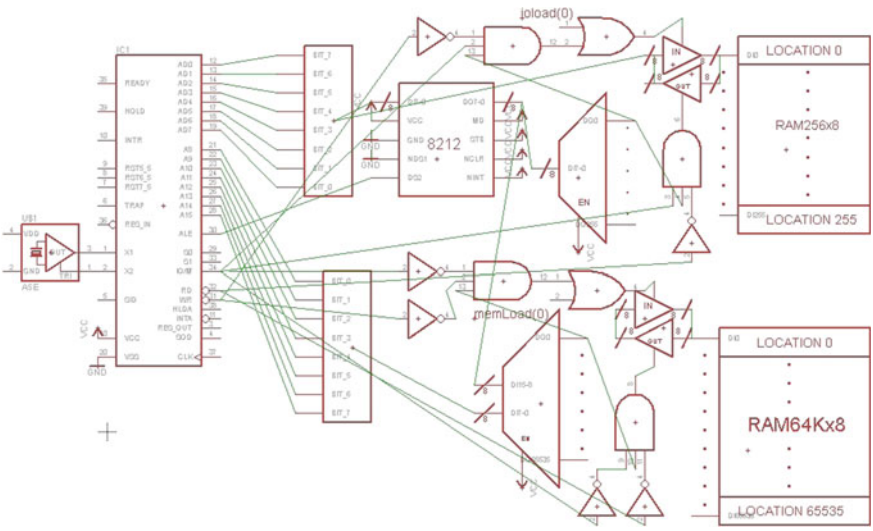


Fig. 3 8085A interface with the address space

waveform diagram in the Xilinx ISE simulator ISim [3] will be provided where all the signal values can be seen.

3.1 LDA Addr

This is a three byte instruction which loads the content of a 16-bit memory address into the accumulator. Prior to the execution of this instruction, the memory address 0080h is loaded with a value of 34h. As we can see in Fig. 4, the program counter starts from 40h because the previous memory locations are reserved for the interrupt service routines table. From the waveform diagrams we can easily see how the instruction is executed: The operational code of LDA (3Ah) is read in OF, and the PC is incremented twice to read the lower byte 80h and the higher byte 00h of the memory address, respectively. After that we see that the contents of 0080h are shown on the AD bus from where they are transferred to the ADB, IDB and finally to the accumulator, which is the last register of the regs array. The last operational code 76h is the code for instruction HLT which stops the processor. From the figure it can also be observed that the lower byte of the address is saved to W and then returned to ADB. If not doing so, it would be overwritten by the higher memory address byte which arrives in ADB after the second memory read.

Fig. 4 Waveform diagram of the LDA instruction

Fig. 5 Waveform diagram of the DAA instruction

3.2 DAA

In order to test this instruction, registers A and B are loaded with values 58h and 47h, respectively. After that the contents of register B are added to the accumulator. The result of the addition is 9F in hexadecimal or 105 in decade. It is evident from Fig. 5 that the contents of the accumulator are correct after the execution of the program. The carry flag is the least significant bit of the flags register. The yellow axis is set to a moment in time before completion of DAA. There isn't any overflow, because the result is 9F, so the carry flag remains 0. But after the completion of the instruction the CY flag is set to 1. This overflow is interpreted as 100 instead of 256 and there is 05 left in the accumulator. Two other flags are also set: the AC (bit 4), because we have carry from the lower nibble and the P flag (bit 2) because we have an even number of ones in the result and 8085A supports odd parity.

3.3 JCondition Addr

There are 8 possible conditional jumps based on the values of the 4 arithmetic flags. The waveform which verifies the correct execution of JZ 0090 is shown in Fig. 6. The accumulator is loaded with 04h and then subsequently subtracted by 02h. After the second subtraction the content of the accumulator will become 0 thereby setting the Z flag. So we expect to see changes in the regs array, the flags register and also the PC because jumps are actually changes of the PC contents. If we observe the regs array, we see that the accumulator decreases to 02 and then to 00. The opcodes in the IR show that the first time JZ is executed, the branch is not taken, since the next code is JMP (C3h), but the second time the condition is met

Fig. 6 Waveform diagram of the JZ instruction

and we have a jump to address 0090h where the instruction HLT stops the processor. If we take a look at the contents of the flags register, we will see that the Z flag (bit 6) is reset during the first subtraction, but it is set during the second one. The contents of the PC also show that a jump has occurred. Its final address is 0091 because it was incremented during the second cycle of opcode fetch. This is so for the PC always has to contain the address of the next instruction to be executed.

4 Interrupts

8085A supports 5 hardware interrupts: TRAP, RST 7.5, RST 6.5, RST 5.5 and INTR. The last interrupt line serves for accepting one of 8 possible software interrupts named from RST 0 to RST 7. As shown in Fig. 7, two multiplexers are used: 4-input one for the hardware interrupts and 8-input one for the software interrupts. The inputs in both multiplexers are 8-bit wide and they represent the starting addresses of the corresponding interrupt service routines. The control inputs for the software interrupts multiplexer are simply bits 5 to 3 from the IR. In the other case, hardware interrupts priority, sensitivity and masking possibility have to be considered [1]. For simulation sake, interrupt generators are just ordinary flip-flops. We suppose that these flip-flops are persistent, which means they hold the interrupt line high until they are reset. The easiest way to reset them is to assign an address from the I/O address space to them and then issue an OUT instruction in the corresponding service routine. The data in the accumulator is insignificant since the 8-bit address represented by logic gates resets the flip-flop. This address is OR-ed with the signal RESET OUT and the output of the OR gate goes into the RESET input of the flip-flop. Figure 8 shows a test bench example of simultaneous occurrence of two interrupts. The corresponding interrupt routines only load a value in the accumulator depending on the interrupt being served. The four hardware interrupt generators occupy the first four I/O addresses, the highest priority interrupt on the lowest address. The interrupt mask register is next-to-last in the regs array. Its value is 0a indicating that RST6.5 is masked. That is why the RST5.5 interrupt service routine is called and value 03 h is sent to address 03h of the RST5.5 interrupt generator. The contents of the I/O addresses can be seen in the ioram array.

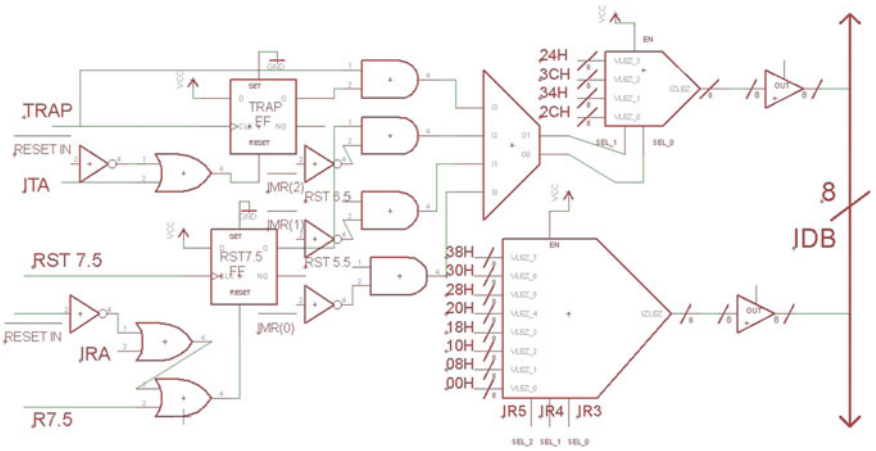


Fig. 7 8085A hardware and software interrupts

Fig. 8 Simultaneous interrupt occurrence

5 Implementation Issues

5.1 ID16 Sequential Design Problems

There were two possibilities for modeling the 16-bit incrementer/decrementer, either as a combinational or sequential circuit. The former was chosen because of the many problems encountered in the latter's design. For example, the edge dependence (either rising or falling) disables consecutive increments/decrements in a clock period. Assuming an instruction that increments/decrements a register and bearing in mind that the PC contents also have to be updated, a situation where count direction change followed by data load in a clock period arises. Regarding the fact that the two previously mentioned introduce spurious edges, thus demanding an additional resetting to be provided in very short time interval, the modeling of such sequential design is almost impossible. Even if it can be made, its complexity would make it impractical. The same functionality can be achieved with a combinational circuit where the non-edge sensitivity enables simultaneous load, count direction change and increment/decrement function with instant results.

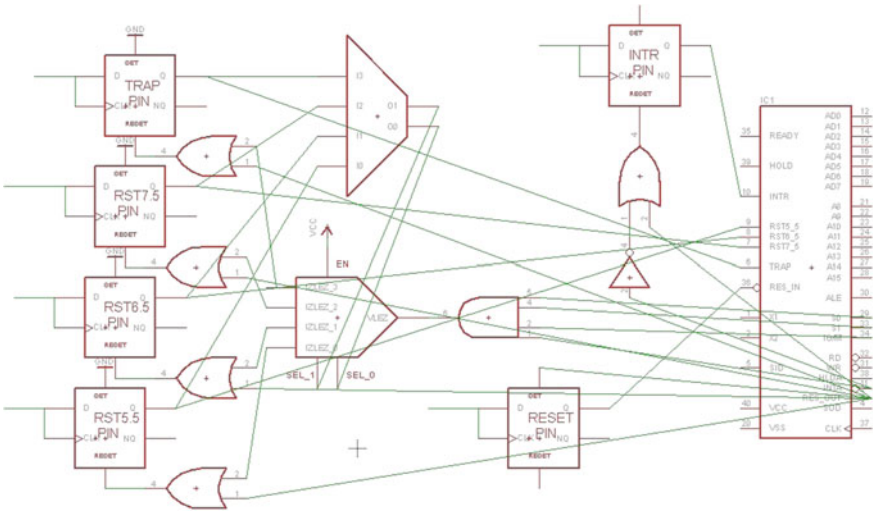


Fig. 9 Hardware-based interrupt acknowledgement

5.2 Hardware-Based Interrupt Acknowledgement

There was a prototype design of a hardware-based interrupt acknowledgement to I/O devices. The design is shown in Fig. 9. The status signals S1, S0, IO/$\overline{M}$ and $\overline{INTA}$ are put into an AND gate which is the input of a 1-to-4 demultiplexer. The outputs of the demultiplexer are OR-ed with the RESET OUT signal and the output from the OR gate is wired to the RESET inputs of the interrupt generators. The control signals for the demultiplexer are the outputs of a priority 4-to-2 encoder where the priority list is set according to the Intel specification. Unfortunately, this solution suffers from several major drawbacks. For example, if a software interrupt appears first, and during the execution of the service interrupt routine a hardware interrupt appears, the AND gate will be active (since $\overline{INTA}$ is low only in T2 and half of T3), thus resetting the hardware interrupt flip-flop. Even though interrupts are disabled during interrupt service routines, it would be convenient for the interrupt request which appears in meantime to remain, so it could be detected after an EI instruction.

An obvious, but rather impossible solution for this problem would be disabling the demultiplexer during the INA cycle. The impossibility of doing comes from the fact that there is no way of distinguishing INA and BI cycles by their first T state. Another, even bigger problem are the masked interrupts, since the correct interrupt will be serviced in the processor, but the I/O device with the highest priority will be acknowledged. The problem arises when this device's interrupt is masked which leads to wrong acknowledgment. An easy solution for overcoming these problems is to use software based acknowledging with the OUT instruction as described in Sect. 4.

5.3 Direct TSB Pairs from PC to AB and ADB

Section 2.5 addressed the 8085A components integration and it was shown in Fig. 2 that there are direct TSBs pairs between the PC and the AB and ADB buffers. This is necessary in case a 4 T-states ALU operation which returns the result in a register is executed. The contents of the program counter have to be transferred to AB and ADB in order to fetch the next instruction. The limitation of using 8-bit bus for this purpose implies two transfers, which means that one transfer has to be done in the previous instruction. But during a 4 T-states ALU operation the IDB is constantly occupied (PC increment, opcode placement in IR, ALU operands load), therefore an alternative way for PC data placement in AB or ADB has to be made. Even though there is a possibility of delayed opcode placement, the insignificant save of several TSBs may result in severe timing problems leading to malfunctioning.

5.4 SOD Flip-Flop Clock Input

Two alternatives for the clock input of the SOD flip-flop were considered. The first approach is a classic one, to connect the SOE pin via TSB as one input of a 2-input AND gate, the other input being the processor clock. The TSB enable input will be active only during the T4 state of the execution of SIM instruction and if its level is high, the processor clock will provide the clock input for the SOD flip-flop. However, the specific implementation uses separate process to update signals related to interrupts, so it would be inefficient to set the processor clock in that sensitivity list [4]. The second approach is to make a circuit which generates impulses. Two TSBs are used, one with the SOE as input and control input active only during T4 state of SIM, the other one with constant low-level input and control input which is inverse from the first one. So the clock level is always low except during SIM, where '1' on the SOE signal would provide rising edge which would send the most significant bit of the accumulator to the SOD line of 8085A.

5.5 Time Delay Issues

Finally, the time delay issue should be mentioned. Time delay problems occur due to the signal propagation through the gates and may become very severe if not taken into account. One of the biggest problems are improper data latching (invalid signal is latched on a rising/falling edge) or data loss due to enabled buffer whose input is in high impedance state. The former could be resolved with delay tuning using additional gates, while the latter could be circumvented with disabling the buffer when a high impedance state input is detected.

```
Device utilization summary:
---------------------------

Selected Device : 5vlx110tff1136-1


Slice Logic Utilization:
 Number of Slice Registers:            2809  out of  69120     4%
 Number of Slice LUTs:                 6413  out of  69120     9%
    Number used as Logic:              6413  out of  69120     9%

Slice Logic Distribution:
 Number of LUT Flip Flop pairs used:   6569
   Number with an unused Flip Flop:    3760  out of   6569    57%
   Number with an unused LUT:           156  out of   6569     2%
   Number of fully used LUT-FF pairs:  2653  out of   6569    40%
   Number of unique control sets:       420

IO Utilization:
 Number of IOs:                         36
 Number of bonded IOBs:                 36  out of    640     5%
    IOB Flip Flops/Latches:             18

Specific Feature Utilization:
 Number of BUFG/BUFGCTRLs:               6  out of     32    18%

 Timing Summary:
 ---------------
 Speed Grade: -1

    Minimum period: 35.381ns (Maximum Frequency: 28.264MHz)
    Minimum input arrival time before clock: 40.228ns
    Maximum output required time after clock: 12.083ns
    Maximum combinational path delay: 12.029ns
```

Fig. 10 Device utilization and timing summaries

6 Conclusion

This paper gives genuine approach to the design of the 8085A hardware components and their integration into a completely functional unit. Based on these schematics a VHDL code is written for the processor [4]. For each component a hierarchical design was used and their corresponding functionalities were tested in Xilinx ISE simulator. Device utilization and timing summaries from the synthesis on a Virtex 5 LX 110T FPGA board using Xilinx XST are presented in Fig. 10.

The execution of the assembler programs is actually various TSB enable inputs manipulation. This design serves as a tool for the Microprocessor Systems laboratory exercises. In order to simplify the assembler code insertion, programs were written into j8085sim [5] and its output was parsed to correspond to the VHDL code for initial program memory load. Some sample instructions, as well as interrupts test bench results were also presented here. The complete instruction set is tested in ISim both as separate instructions and as a part of a bigger assembler program and so far there are not any execution errors. The stack operations, the procedure call and return as well as the interrupts work fine.

Assembler programs with two or more hardware interrupts occurring simultaneously were made to test if the priority of interrupts is maintained and if the correct interrupt generator device is being acknowledged. Last but not least, some of the major issues encountered were discussed, along with an explanation why the choice that was made for this specific implementation was the right one. The next step is placing the processor on an FPGA board to see if the simulation results will be confirmed in practice.

References

1. MCS-80/85 Family User's Manual, Intel Corporation (1979)
2. ISE In-Depth Tutorial, UG695 (v14.1) April 24, 2012
3. ISim In-Depth Tutorial UG682 (14.2) July 25, 2012
4. VHDL Tutorial by Peter J. Ashenden, Elsevier Science, USA (2004)
5. j8085sim - an 8085 simulator in Java http://sourceforge.net/projects/j8085sim/

A Novel Texture Description for Liver Fibrosis Identification

Nan-Han Lu, Meng-Tso Chen, Chi-Kao Chang, Min-Yuan Fang and Chung-Ming Kuo

Abstract In this study, the proposed texture description method is applied to obtain the description of ultrasound images of hepatic parenchyma. The result of performance characteristics for distinguishing liver fibrosis and normal liver is shown. The diagnostic performance is accessed on two different approaches and two set of parameters including CO-LBP 50 × 50, CO-RLBP 50 × 50, CO-LBP 75 × 75 and CO-RLBP 75 × 75. We find that CO-RLBP method is better than that of CO-LBP method in overall accuracy.

Keywords Texture description · Ultrasound image · Liver fibrosis

1 Introduction

Medical imaging examinations including ultrasound, computed tomogram (CT), magnetic resonance image (MRI), and angiography have been widely used to evaluate the chronic hepatic parenchyma disease, especially ultrasound for early examination [1–11]. These imaging modalities can be effectively instead of the anatomic information of liver and detect the abnormal change of hepatic parenchyma including distortion of architecture and tumor mass. In Taiwan, there are more than 100,000 peoples per year receiving the abdominal B-mode ultrasound examination. In routine clinical practice, an objective and quantitative method of evaluating liver fibrosis on B-mode ultrasound image will play an important first-line role for follow-up of patients with CLD. If the ultrasound finds abnormal

N.-H. Lu · M.-T. Chen · C.-K. Chang · M.-Y. Fang · C.-M. Kuo (✉)
Department of Information Engineering, I-Shou University, Kohshiung, Taiwan
e-mail: kuocm@isu.edu.tw

N.-H. Lu
Department of Radiology, E-DA Hospital/I-Shou University, Kohshiung, Taiwan

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_24,

feature, the CT or MRI study is recommended for further evaluation. Meanwhile, it is a non-invasive safest modality and can immediately assist the physician to diagnosis according to the real time information of ultrasound images. For the diagnosis of diffuse liver diseases ultrasound is commonly used, but visual criteria offers low diagnostic accuracy, and it depends on the experienced radiologist.

Common features, including hepatic parenchyma echogenicity, texture, and liver surface, are used to assess liver fibrosis on clinical B-mode ultrasound practice. These features imply the subjective character of ultrasound interpretation. The development of an objective method based on B-mode ultrasound images for CLD staging classification is imperative. Recently, the quantitative identification and classification of ultrasound images have become very desirable due to the rapid development of computer and imaging technologies. Several studies have reported the issue of quantitative ultrasound examination of hepatic fibrosis by using of statistical data on the ultrasound echo signals [4], texture analysis of B-mode images [5, 6], and fractal dimensions of the scattering signals [7], and statistical analysis of signals [8].

For liver fibrosis identification, we need to extract the significant texture features from the representative samples for detection and classification of liver fibrosis. Therefore, in this paper, we will focus on the texture features extraction and description. We propose a novel feature description technique based on new developed co-occurrence LBP (local binary pattern) and RLBP (ambiguous local binary pattern) texture feature. Then, according to the texture description of training B-mode ultrasound liver image, a dominant component representation based on statistics is introduced to classify the textures. The simulation results show that the proposed method achieves superior performance for classification of liver fibrosis than that of traditional algorithms.

2 Proposed Methods

Local binary pattern (LBP) is originally proposed for texture image segmentation [10]. The binarization of LBP is a hard decision by thresholding center pixel, it is very noise sensitive. For noisy image such as ultrasound image, it will strongly affect the correctness of similarity. According to the drawbacks mentioned above, we will propose a novel range local binary pattern (RLBP) to address the problem of noise interference. In our work, we define an ambiguous range for the LBP thresholding. Let T be the ambiguous range, Eqs. (1) and (2) are the calculation of RLBP.

$$d_n(i,j) = \begin{cases} 1 & if \quad p_n \geq p_{center} + T \\ 0 & if \quad p_{center} - T < p_n < p_{center} + T \\ -1 & if \quad p_n \leq p_{center} - T \end{cases}, \quad n \in (i,j)_{3\times 3}, \tag{1}$$

$$RLBP(i,j) = \sum_{n=0}^{7} d_n(i,j) \cdot 2^n \tag{2}$$

where p_n is the pixels in 3 × 3 mask, p_{center} is the center pixel in 3 × 3 mask, 2^n is the weighting value in mask and $d_n(i,j)$ is the decision results. The histogram of LBP can be expressed as

$$\begin{aligned} H_{RLBP} &= \langle h_{RLBP}(k) \rangle \, k = -255, -254, \ldots, 255, \\ h_{RLBP}(k) &= \frac{1}{M \times N} \sum_{i=0}^{M} \sum_{j=0}^{N} \delta(RLBP(i,j) - k) \\ k &= -255, -254, \ldots, 255, \end{aligned} \tag{3}$$

$$H_{RLBP} = \sum_{k=-255}^{255} h_{RLBP}(k) = 1 \quad k = -255, -254, \ldots, 255, \tag{4}$$

where M and N are the height and width of texture image, k is the RLBP value, $RLBP\ (i, j)$ is the RLBP value in (i, j) and $h_{RLBP}(k)$ is the normalized distribution of the kth bin.

The most important difference in LBP and RLBP is the consideration of noise introduced in ultrasound image. RLBP considers the noisy effect and create "0'' to represent the value which is close to the ambiguous range of central pixel value. RLBP provides two advantages. First, it can reduce wrong decision. Second, it can capture the precise LBP for statistical analysis. Using the RLBP, it can provide the convincing analytical result in the description of texture feature under the condition of noisy effect. Therefore, we can expect that the proposed feature representation will effectively address the noise interference in ultrasound image.

2.1 LBP and RLBP Representation

To consider the texture distribution more comprehensively, we improve the LBP and RLBP by using various dimensions to compute LBP. The new spatial dimensions for LBP computation are with mask 3 × 3, 5 × 5 and 7 × 7, and be expressed as $\{LBP^3(i,j), LBP^5(i,j), LBP^7(i,j)\}$, respectively. For $LBP^3(i,j)$, the computation is the same as conventional LBP. In order to satisfy the computation rule, the mask of $LBP^5(i,j)$ or $LBP^7(i,j)$ is simplified from a 5 × 5 or 7 × 7 to 3 × 3; the simplification scheme is shown in Fig. 1, where each element is calculated by averaging the pixels value in mask.

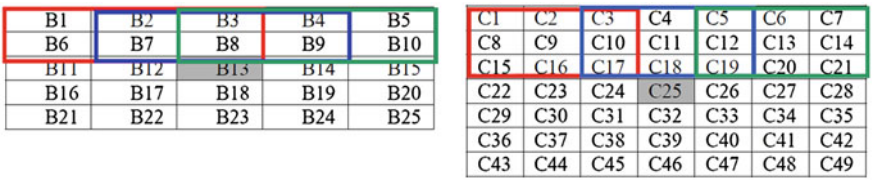


Fig. 1 $LBP^5(i,j)$ and $LBP^7(i,j)$, the elements in each mask reduce to one element by averaging

2.2 Co-occurrence Representation of LBP

We use the concept of co-occurrence matrix [5] to express the spatial relationship of RLBP. According to the description of co-occurrence matrix feature, we can clearly identify the relationship of spatial distribution between two texture features. In order to achieve the co-occurrence representation, the conventional LBP mask is decomposed into two sub-masks i.e., "cross" and "corner". As calculation of LBP, the value of each sub-mask is given in Eq. (5) and (6).

$$c_n^{+(or\times)} = \begin{cases} 1 & if\ p_n \geq p_{center} \\ 0 & if\ p_{center} - T < p_n < p_{center} + T \\ -1 & if\ p_n \leq p_{center} - T \end{cases} \tag{5}$$

$$RLBP^{+(or\times)}(i,j) = \sum_{n=0}^{3} c_n^{+(or\times)} \cdot 2^n \tag{6}$$

where $c_n^{+(or\times)}$ is the decision results for cross pattern (or corner pattern) and the patterns are shown in Fig. 2. The histogram of *RLBP* can be expressed as:

$$h_{RLBP}^{+(or\times)}(k) = \frac{1}{M \times N} \sum_{i=0}^{M} \sum_{j=0}^{N} \delta\left(RLBP^{+(or\times)}(i,j) - k\right) \quad k = -15, -14, \ldots 15 \tag{7}$$

$$\sum_{k=-15}^{15} h_{RLBP}^{+(or\times)}(k) = 1 \quad k = -15, -14, \ldots, 15 \tag{8}$$

Once the RLBP values for each types are calculated, we define a two dimensional matrix to record the statistics of RLBP. Let the row and column represent the "cross" and "corner" features, respectively. For simplicity the symbol $+(i)$ and $\times(j)$ are used to represent the value of $RLBP^{\times} = i$ and value of $RLBP^{+} = j$ respectively.

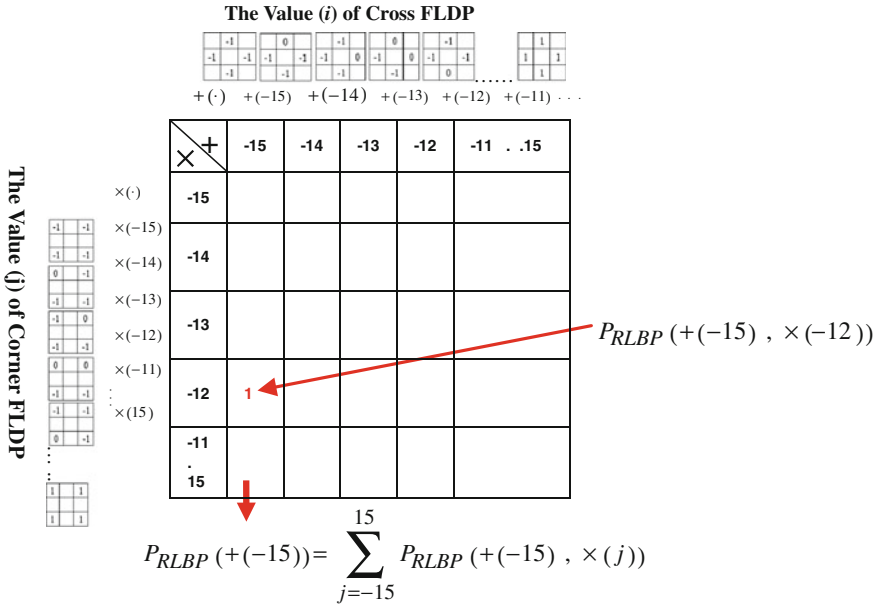


Fig. 2 Co-occurrence representation of RLBP

$$P_{RLBP}(i,j) = \left\{ \begin{array}{l} (+_{RLBP}(i) \ and \ \times_{RLBP}(j)); \\ -15 \le i \le 15 \ and \ -15 \le j \le 15 \end{array} \right\} \tag{9}$$

where $P_{RLBP}(i,j)$ is the element of co-occurrence matrix of *RLBP* with $RLBP^{\times} = i$ and $RLBP^{+} = j$, respectively. In Fig. 2, we show a co-occurrence matrix of RLBP with 3 × 3 mask, respectively. Because the RLBP is calculated from three different dimensional respectively, the RLBP from different scale should be merged by weighting sum to obtain the texture feature in various resolution. The LBP representation can be expressed as

$$CoP_{RLBP}(i,j) = 0.6 \times P^{3\times3}_{RLBP}(i,j) + 0.3 \times P^{5\times5}_{RLBP}(i,j) + 0.1 \times P^{7\times7}_{RLBP}(i,j) \tag{10}$$

where $CoP_{RLBP}(i,j)$ is the combined co-occurrence matrix of *RLBP* by the scale 3 × 3, 5 × 5 and 7 × 7, respectively.

2.3 Dominant Component of RLBP

In order to achieve texture categorization, we need to find the characteristics of feature distribution for each texture class and then input textures can be classified according to the similarity measure of feature distribution. In our work, the

classification of texture feature distribution model is characterized by dominant components of RLBP co-occurrence matrix. For training phase, we select the training image from each class of texture, and then the class feature model is built accordingly. For testing phase, the feature distribution of input image is calculated and then classification by similarity measure with class model. To extract the dominant components for class feature model, we select the number of S training image from each class C, and express the dominant components as $T_s^c = \{t_s^c | c = 1, \ldots, C, s = 1, \ldots, S$. We calculate the co-occurrence matrix $Co\mathbf{P}_{RLBP}(i,j)$ for each training image, and then binarize the matrix by thresholding as Eq. (11):

$$\mathbf{B}_{RLBP}(i,j) = \begin{cases} 1 & if\ \mathrm{Co}\mathbf{P}_{RLBP}(i,j) \geq T \\ 0 & else \end{cases} \tag{11}$$

where $\mathbf{B}_{RLBP}(i,j)$ are the binarization matrix of $Co\mathbf{P}_{RLBP}(i,j)$. The T is the threshold. The $\mathbf{B}_{RLBP}(i,j)$ is binary feature matrix of texture image. We count each element of binary matrix in all training images, the dominant components are the elements that most frequently appeared in all binary feature matrix. Therefore, for texture class, the dominant components are calculated from the training images by counting the appearance number of 0 and 1. The appearance number can be expressed as

$$\begin{aligned} \mathrm{b}_{\overline{RLBP}}^{c\_1}(i,j) &= \sum_{k=1}^{S} \left[\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \delta\left[\mathrm{b}_{RLBP-k}^{c}(i,j) - 1 \right] \right] \\ \mathrm{b}_{\overline{RLBP}}^{c\_0}(i,j) &= \sum_{k=1}^{S} \left[\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \delta\left[\mathrm{b}_{RLBP-k}^{c}(i,j) \right] \right] \end{aligned} \tag{12}$$

The dominant components is calculated by

$$DC_{RLBP}^{c}(i,j) = \sum_{k=1}^{S} \left[\sum_{i=0}^{M-1} \sum_{j=0}^{N-1} p_{RLBP-k}^{c}(i,j) \delta\left[b_{RLBP-k}^{c}(i,j) - \max \right] \right] \Big/ Max\left(b_{\overline{RLBP}-k}^{c\_1}(i,j), b_{\overline{RLBP}-k}^{c\_0}(i,j) \right)$$

and

$$\max = \begin{cases} 1, & b_{\overline{RLBP}-k}^{c\_1} = Max\left(b_{\overline{RLBP}-k}^{c\_1}(i,j), b_{\overline{RLBP}-k}^{c\_0}(i,j) \right) \\ 0, & else \end{cases}, \tag{13}$$

where $p_{RLBP-k}^{c}(i,j)$ is the element of $\mathbf{P}_{LBP}(i,j)$, the k and c means the kth training image in class c.

The similarity measure with dominant component is to calculate the overall difference, it can be defined as,

Table 1 Representation of performance indexes

	Liver fibrosis	Normal liver
Positive	True positive (TP)	False positive (FP)
Negative	False negative (FN)	True negative (TN)

$$D_{RLBP} = \sum_{i=-16}^{15} \sum_{j=-16}^{15} \left| \left(Co\mathbf{P}_{RLBP}(i,j)_{query} - DC^{c}_{RLBP}(i,j) \right) \right| \tag{14}$$

where $Co\mathbf{P}_{RLBP}(i,j)_{query}$ is the texture feature of query image. The similarity is defined as the minimum distance.

3 Experimental Results

In our experiments, the test liver ultrasound images were obtained on Toshiba, Aplio 50, SSA-700A ultrasonic machine with a 3.5 MHz frequency convex abdominal transducer. All images were of 560 × 450 pixels. Ultrasound images for different liver cases were collected from patients with known histology and accurate diagnosis by expert radiologist from Department of Radiology, E-DA Hospital, Kaohsiung, Taiwan. Also, this study was approved by the local Ethical Committee of the E-DA Hospital (EMRP-101-018). We collected 84 cases in our study. Two set of ROI images have been taken: normal liver and chronic hepatitis with 217(40 cases) and 479(44 cases) images retrospectively. In order to perform fair comparison, we select 200 ROI images to build up the test dataset from each set equally. In our work, the performance indexes as in Table 1 were selected to evaluate the performance.

In the experiments, we compare the performance with various methods, as showm in Table 2, we can easily find that the proposed method achieves best performance. The results of overall categorization using LBP and RLBP methods are shown in Table 3. LBP method achieves true positive rate (74.4 %), false positive rate (49.35 %) and accuracy (63.02 %). RLBP-DS method shows true positive rate (89.4 %), false positive rate (6.7 %) and accuracy (91.35 %). According to the validated result, the proposed texture feature descriptors are very useful for categorization of liver fibrosis and normal liver. Meanwhile, RLBP method is apparently better than LBP method for all performance indexes.

4 Discussion and Conclusion

In summary, the new method is very helpful to increase the diagnostic accuracy in clinically. In the future, we will continue to develop this research and to analyze more ultrasound images in order to modify the parameters of our method for

Table 2 Comparison between proposed method and the other methods

Methods	Accuracy rate (%)	False-negative rate (%)
CoP_{RLBP} (75 × 75)	96.3	0.8
TFCM	86.7	4.4
CM	75.7	8.9
SFM	55.55	18.9
TS	57.78	12.2
FD	64.4	14.4

Table 3 Result of overall ultrasound image categorization

LBP	Liver fibrosis	Normal liver	RLBP	Liver fibrosis	Normal liver
Positive	74.4 %	48.35 %	Positive	89.4 %	6.7 %
Negative	25.6 %	51.65 %	Negative	10.6 %	93.3 %
True positive rate		74.4 % (conventional LBP)		89.4 % (proposed RLBP)	
False positive rate		48.35 % (conventional LBP)		6.7 % (proposed RLBP)	
Accuracy rate		63.02 % (conventional LBP)		91.35 % (proposed RLBP)	

optimization of our design. We will focus in some issues as follows: (1) to modify the parameters of our method for optimization; (2) to improve the automatic segmentation technique for computational cost; (3) to perform the verifying experiment using more clinical ultrasound images.

Acknowledgement This work was supported by the National Science Counsel Granted NSC 100-2221-E-214-064-

References

1. Maddrey, W.C.: Hepatitis B: an important pubic health issue. J. Med. Virol. **61**, 362–366 (2000)
2. Vural, R.A., Ozyilmaz, L., Yildirim, T.: A comparative study on computerized diagnostic performance of Hepatitis Disease Using ANNs, ICIC, pp. 1177–1182 (2006)
3. Lu, S.N., Su, W.W., Yang, S.S., et al.: Secular trends and geographic variations of hepatitis B virus and hepatitis C virus-associated hepatocellular carcinoma in Taiwan. Int. J. Cancer **119**, 1946–1952 (2006)
4. Mohana Shankar, P.: A general statistical model for ultrasonic backscattering from tissue. IEEE Trans. Ultrason. Ferroelectr. Freq. Control **47**, 727–736 (2000)
5. Valckx, F.M.J., Thijssen, J.M.: Characterization of echo graphic image texture by co-occurrence matrix parameter. Ultrasound Med. Biol. **23**, 559–571 (1997)
6. Horng, M.-H., Sun, Y.-N., Lin, X.-Z.: Texture feature coding method for classification of liver sonography. Comput. Med. Imaging Graph. **26**, 33–42 (2002)
7. Kikuchi, T., Nakazawa, T., Furukawa, T., Higuchi, T., Maruyama, Y., Sato, S.: Quantitative estimation of the amount of fibrosis in the rat liver using fractal dimension of the shape of power spectrum. Jpn. J. Appl. Phys. **34**, 2831–2834 (1995)

8. Toyoda, H., Kumada, T., Kamiyama, N., et al.: B-mode ultrasound with algorithm based on statistical analysis of signals: evaluation of liver fibrosis in patients with chronic hepatitis C. AJR Am. J. Roentgenol. **193**(4), 1037–1043 (2009)
9. Yamada, H., Ebara, M., Yamaguchi, T., et al.: A pilot approach for quantitative assessment of liver fibrosis using ultrasound: preliminary results in 79 cases. J. Hepatol. **44**, 68–75 (2006)
10. Ojala, T., Pietikäinen, M., Harwood, David: A comparative study of texture measures with classification based on feature distributions. Pattern Recogn. **29**(1), 51–59 (1996)
11. Karule, P.T., Dudule, S.V.: PCA NN based classifier for liver disease from ultrasonic liver images. In: ICETET, pp. 76–80 (2009)

Topology Discovery in Deadlock Free Self-assembled DNA Networks

Davide Patti, Andrea Mineo, Salvatore Monteleone and Vincenzo Catania

Abstract In this paper we present a novel approach to topology discovery and defect mapping in nano-scale self-assembled DNA networks. The large scale randomness and irregularity of such networks makes it necessary to achieve deadlock freedom without the availability of a topology graph or any other kind of centralized algorithms to configure network paths. Results show how the proposed distributed approach preserves some important properties (coverage, defect tolerance, scalability), reaching a segment-based deadlock freedom while avoiding centralized tree-based broadcasting and hardware node hungry solutions not feasible in such a limited nanoscale scenario. Finally, we quantitatively evaluate an not-optimised gate-level hardware implementation of the required control logic that demonstrates a relatively acceptable impact ranging from 10 to about 17 % of the budget of transistors typically available at each node using such technology.

Keywords Nanotechnology · DNA · Self-assembly · Routing · Deadlock

D. Patti (✉) · A. Mineo · S. Monteleone · V. Catania
University of Catania, Catania, Italy
e-mail: dpatti@dieei.unict.it

A. Mineo
e-mail: amineo@dieei.unict.it

S. Monteleone
e-mail: smontele@dieei.unict.it

V. Catania
e-mail: vcatania@dieei.unict.it

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_25,

1 Introduction and Motivation

DNA Self-assembled nanoscale networks [15] have been studied in the last years as a promising technology due their huge potential computing capabilities and different laboratory demos and prof-of-concepts architectures have been presented (e.g. [12]). The main idea behind these networks is to exploit capability of DNA sequences to self-assemble themselves in regular structures creating a scaffold onto which nano-devices (e.g. nanowires and CNFETs [2, 4]) can be attached. This can be achieved by designing appropriate complementary DNA tags for each terminal to be placed, so that a nano device will be attached only where its own DNA tag matches a complementary tag on the DNA grid scaffold. A detailed description of the chemical properties involved is far beyond the scope of this paper (see also [14]), so we will focus on the three main properties that characterize these networks: (i) *limited node complexity*, (ii) *large scale randomness*, and (iii) *high defect rates*.

The *limited node complexity* aspect is directly related to the use of complementary DNA tags in order to place circuit components: creating many different tags would mean make them more similar to each other, increasing the probability of incorrect/partial matching. To avoid this problem we should limit the number of unique tags, thus limiting the complexity at each node. In particular, a budget of about 10,000 CNFETs per node has been estimated in [8]. *Large scale randomness* and *high defect rates* lead to huge networks where typical properties of regular topologies cannot be guaranteed, e.g. a node being connected to a fixed number of neighbors, having a determined orientation and so on.

These aspects of DNA-self assembled networks, together with their tera/peta scale of integration technology, lead to some important implications to be addressed from a Computer Architecture Design perspective: we have a theoretical computational power of hundreds of thousands of nodes, but the execution model should be based upon a distributed architecture of small computing and storage nodes, randomly placed and interconnected. Since no regularity can be assumed in such networks, a topology agnostic strategy that avoids deadlock should be adopted in order to route data (e.g. instructions and data operands) among nodes.

In this paper we introduce DiSR, a Distributed Segment-based approach to deadlock freedom in large scale DNA self assembled networks. Our contribution aims to achieve the classical properties of a segment-based approach [9] without requiring any topology graph, external defect map or centralized algorithm execution. So the DiSR approach is not intended to discover the "optimal" segment choice (ideally reachable with the knowledge of the topology graph) but just to demonstrate a concrete model that can fit into such complex, irregular and large sized networks.

2 Related Works

A lightweight strategy to achieve deadlock freedom is the use of turn prohibitions [3]. In particular, authors of [11] exploits the creation of a spanning tree of the topology, then placing bidirectional restrictions by avoiding a packet to traverse the same link in both up and down directions. While the hierarchical nature of this approach can lead to uneven traffic distribution, with many packets traversing upper links (near to the root), this is quite acceptable in classical wide area networks topologies with a limited number of nodes. Other approaches such as FX [13] mitigate this issue, but the set of turn restrictions is still prefixed, strictly depending on the particular tree root selected.

In [9] authors present SR, an approach the solves these limitations by allowing turn restrictions to be placed locally, independently from other restrictions. The whole network is partitioned into segments, and each bidirectional turn restriction can be freely chosen within a segment in order to guarantee deadlock freedom and connected networks. This *locality independence* property, together with no requirement of any particular tree/root choice, would make it the best choice for the given scenario; however, its topology independency still requires the knowledge of the whole network graph in order to find the segments.

Other solutions try to approach the issue of irregular topologies by limiting the number or the location of missing links [6, 7], but restriction is clearly unacceptable in a DNA self-assembled networks scenario. For the same reason, we also avoid considering solutions based on virtual channels or hardware-redundancy to dynamically recover defects as in [5].

3 Preliminary Concepts

3.1 Basic Idea

The main concept behind the kind of turn prohibition we want to achieve is the *Segment*: it is basically a path of consecutive nodes and links. A segment S_1 starts with a link attached to a node which belongs to different segment S_2, ending with a link attached to a node belonging to a different segment S_3. In other words, a segment a path between connecting two other different segments. A exception is the first segment established in the network, called *starting segment* which is a loop beginning/ending on a particular node defined as *bootstrap node*. The idea is to partition the network in a set of disjoint segments, and then placing a turn restriction within each segment. It has been proved [9] that such a set of turn prohibitions guarantees deadlock freedom and while preserving connectivity of the network.

Describing the execution phases of DiSR from a top level point-of-view can be useful to give an initial idea of the approach; however, some substantial issues

should be pointed out. First, no centralized entity is globally aware of what is going on, so the status of the DiSR execution is collectively distributed among the nodes. Further, no defect map and/or topology graph is available as input, thus, the topology has to be discovered *while* segments are created. Finally, at the end of the execution no segment list is created: Each node is only aware of belonging to some segment, ignoring the presence of other nodes in the same segment and even the presence of other segments in the network. Roughly speaking, the execution of can be described as the execution of the following phases: (1) Injection of the DiSR process from upper layer to set a bootstrap node. (2) Bootstrap node broadcasting to create the first segment of the subnet. (3) Parallel requests starting from assigned node to discover other segments.

3.2 Message Types Required

The DiSR approach works with a distributed mechanism which is build upon an exchange of small packets containing three simple fields: a *packet_type* encoding the meaning of the control message, a *segID* of the segment associated to the DiSR control message and a *src_id* representing the id of the node that originated the packet. In particular as regards packet_type, we can have the following control packets:

STARTING_SEGMENT_REQUEST: injected by the bootstrap node when searching for the first segment.
STARTING_SEGMENT_CONFIRM: used when establishing the starting segment.
SEGMENT_REQUEST: search candidates for a segment.
SEGMENT_CONFIRM: establish a segment.
SEGMENT_CANCEL: cancel the search along a specific link. A quantitative analysis of the resources needed to implement these fields is presented in Sect. 5 when discussing the impact of the DiSR control logic and storage on hardware implementation of the node.

3.3 Node Data Required

We distinct between two different kind of data stored at each node: *Local Environment Data* (*LED*) and *Dynamic Behaviour Status* (*DBS*).

The *LED* is like a snapshot of the DiSR algorithm at each node, consisting in the following variables:

- *segID*: a value used to specify the segment to which the node has been assigned or is candidate for being assigned.

- *visited*: a boolean value. When *true* a *segID* different from NULL specifies the segment to which the node has been assigned.
- *tvisited*: a boolean value. If *true*, the node is being considered as candidate for a segment, and the *segID* value specifies the segment id for which the node is candidate.
- *link_visited[]*: an array of values representing information about attached links, that is, the segID of the segment owning each link. When NULL, the corresponding link has not yet been assigned.
- *link_tvisited[]*: an array of values representing information about attached links, that is, the *segID* of the segment for which the link is candidate. When NULL, the link is currently not *tvisited*.

In addition to the *LED* variables described above, further information should be stored in order to capture the current dynamic behaviour of the node. This is represented by *DBS* variable, which strictly depends on the *LED* data and the events occurring at the node. The *DBS* can have the following values:

- *Free*: a node that has not been yet considered by the DiSR algorithm. The node is not marked as *visited/tvisited*.
- *Bootstrap*: a node which has been explicitly set as bootstrap node from an upper layer via.
- *ActiveSearching*: a node from which a find process of new segments has been started and not yet cancelled or confirmed.
- *Candidate*: a node currently candidate for belonging to some segment with id segid, not being itself the node from which the find process was started.
- *CandidateStarting*: same as above, but the node is currently being considered as candidate for starting segment.
- *Assigned*: a node for which the segment has been determined. The segment *segID* attribute value is set to some id X different from NULL.

3.4 DiSR Execution Model

In Fig. 1 is depicted the main events representing the execution model at each node. The DiSR control packets described in the previous section trigger node events, which eventually change *DBS* status according to the current value of *LED* variables and the type of the packet received. The main DiSR phases, together with a reference to the particular status transition shown in Fig. 1 is described in the following.

Injecting bootstrap request: all nodes have an initial *DBS* status *Free*, except for a node with status *Bootstrap*, set by some signal from an upper layer via (a). When starting, bootstrap node changes its status to *ActiveSearching* (b), injecting a STARTING_SEGMENT_REQUEST across one of its free links.

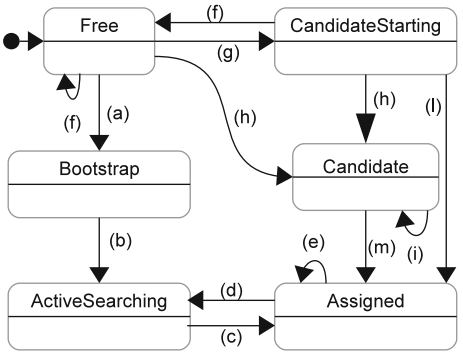


Fig. 1 DiSR node execution model

Flooding: a node receiving a STARTING_SEGMENT_REQUEST, when *Free*, forwards it to its free links and becomes *CandidateStarting* (g). Each of its free links is then marked as *tvisited* with the segment id associated to the request. A node that has already received a STARTING_SEGMENT_REQUEST packet can simply ignore further packets associated to the same request, having already contributed to the flooding.

Confirming the starting segment: If a STARTING_SEGMENT_REQUEST reaches the bootstrap node (from a different link), the starting segment is found. Then the bootstrap node sends a STARTING_SEGMENT_CONFIRM packet along the link from which it received the request and becomes *Assigned* (c). Each node receiving the confirm do same by changing its own status to *Assigned* (l). So the confirmation packet is sent back from node to node and the starting segment is created. Note: A node being *CandidateStarting*, when receiving a simple (not starting) SEGMENT_REQUEST, can simply cancel its previous *CandidateStarting* status and set itself as *Candidate* for that request (h), since this new request means that the starting segment has already been found.

Injecting other requests: each node in the *Assigned* status can initiate a search for a segment, by sending a new SEGMENT_REQUEST across one of its free links (d). Note that since this is not the starting segment in this case the packet should not reach the initiator, but just another *Assigned* node.

Setup of a segment: the find process is successful when an already *Assigned* node receives a SEGMENT_REQUEST packet. Then, a SEGMENT_CONFIRM is sent back along the path that originated the request (e), while the node remain *Assigned* (since it could confirm more segments). Each node previously set as *Candidate* for that segment id, when receiving the confirm packet changes its status to *Assigned* (m) and forwards back the same confirm until it reaches the initiator of the request, which changes from *ActiveSearching* to *Assigned* (c).

Failing while searching a segment: a node received a SEGMENT_REQUEST packet but matched one of two the following conditions: the node is *Free* but has no more suitable free links (thus can't forward the SEGMENT_REQUEST) (f); the node is candidate for another find process (i). In all these cases the node sends back a SEGMENT_CANCEL along the proper link.

4 Simulation and Results

In order to quantitatively and qualitatively evaluate the proposed approach a specific simulation environment has been developed, resulting in the open source and freely available project called Nanoxim [10]. Nanoxim is a SystemC tool based on a almost rewritten version of the Noxim Network-on-Chip simulator [1]. While some complex features have been removed (e.g. wormhole, congestion/topology aware routing and selection strategies) new features specifically taylored for the nanoscale scenario were introduced (e.g. the ability to simulate a random network, the implementation of the DiSR to obtain the segment topology and the support for defective links and nodes).

4.1 Experimental Setup

The following parameters have been taken into account when performing the DiSR simulation:

Size of the network: number of nodes, on a range from 10×10 to 100×100 sized networks.
Defective nodes: the probability that a node is not working, thus having all its links cannot be utilized during DiSR setup.
Bootstrap node: the node used from upper layer to inject the DiSR process. When not explicitly investigating the impact of each single bootstrap choice, a set of representative regions has been considered, e.g. the central part of the network and the edge corners.
To present the results, the following evaluation metrics have been adopted:
Node coverage: this is the fraction of nodes that are assigned to a segment. In the ideal case, all the non defective nodes should be assigned, so this metric is useful to show how some disconnected regions can negatively impact on the whole DiSR effectiveness.
Latency: this measures how the cycles required to complete the segment assignment scales for increasing network sizes and defect rates.
Bootstrap node effect: this evaluates the impact of the chosen bootstrap node on the node coverage.

Since the distribution of defects and thus the resulting topology is randomly generated, a set of simulations with different seeds has been run for each system configuration. We found that 20 repetitions are required in order to obtain statistically significant results.

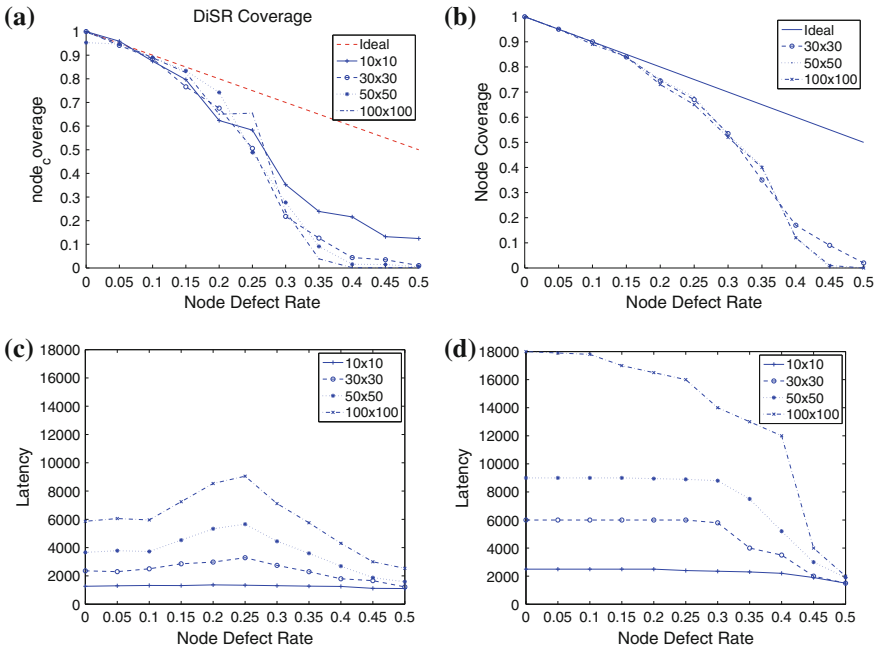


Fig. 2 DiSR (**a**) and RPF (**b**) node coverage, Latency of DiSR (**c**) versus tree based RPF (**d**)

5 Results

In this section we analyze the results in terms of node coverage and latency at different network sizes, defect rates and bootstrap injection points. In particular Fig. 2a, b show node coverage for DiSR and RPF tree based approach respectively. While the first aim of DiSR is not to reach the optimal coverage, we still can observe a quite good performance as compared to the tree based approach. Note that defect rates beyond 25 % lead to many disconnected regions of nodes that DiSR currently cannot handle. In any case, these defect levels should be considered as worst case scenarios, so the achieved coverage of 0.5 is a satisfying result for this first version of DiSR. On the other hand, the network size seems to have a limited impact when defect rate do not introduce too much disconnected regions.

The number of cycles required to complete segment mapping process is shown in Fig. 2c, d. In this case comparison against tree-based shows better (lower) values at different defect rates. Rather than the absolute numbers, what it's more interesting to observe is how DiSR latency scales with network size. For example, going from 900 to 2,500 nodes, at the medium defect rate of 0.15, leads to an increase from 3,000 to 4,500 cycles. It should be noticed also how the effect of defect rate is increasing until the threshold of 0.25 is reached, meaning that until

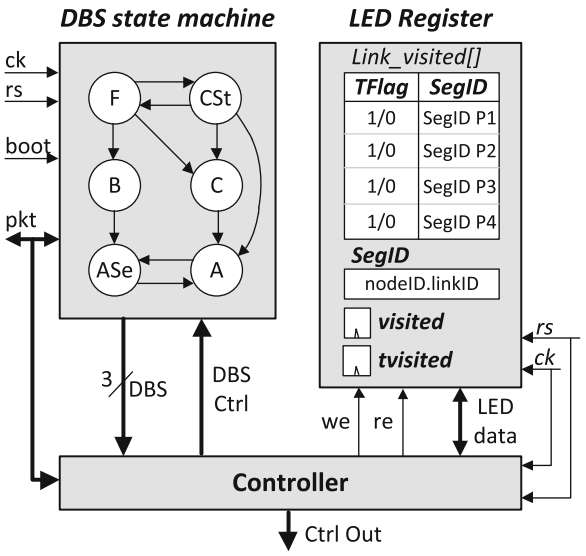


Fig. 3 DiSR block architecture

that limit DiSR finds it more and more difficult to complete the process due increasing defective paths, but still discover new segments when let running for a more extended amount of cycles. This behavior is not reported in the RPF based approach, meaning that a tree based approach, although starting from higher values, is less affected by defect rates (when low rates are considered). After the 0.25 threshold, the impact of entire disconnected regions becomes predominant and both approaches become faster in completing the covering process, since far less nodes can be actually reached.

6 Proof-of-Concept Hardware Implementation

To give an estimate of the overhead needed, we will focus on the control logic and configuration registers needed to implement DiSR. In Fig. 3 is shown a sketch of a possible implementation, which mainly consists in the following building elements:

DBS block: it takes trace of the DBS state machine, consisting of a 3-bit register (to cover six DBS values) and the requird combinational logic.
LED registers: a set of registers storing *LED* information. A special register named *Tflag* indicating whether *SegID* value refers to *visited* or *tvisited*.
Control circuitry: this circuitry reads data from the fields of the incoming packet, LED registers and the *DBS*, updating data when required. Then, the resulting *Ctrl Out* drives the other communication resources for actuating the DiSR routing operation.

Assuming a budget of 10^4 CNFETs for each network's node [8] we estimated the required resources to implement the entire DiSR block. A not optimized behavioural HDL of the DiSR circuitry has been synthesized at gate-level. Considering the specific layout of each single logic elements (NAND, full-adder, latch etc.), it has been possible to get a rough estimate of the number of transistors necessary for the DiSR logic, showing an impact of about 17 % of the node budget. Further, scalable storage structures can be used: for example, the number of registers implementing the *link_visited[]* table follows the logarithmic function: $N_{reg} = N_{port} \cdot log_2(N)$, where N_{port} is the number of the router's ports, N the number of nodes. Finally, while this number is function of network nodes, the control circuitry could be optimized in future designs of the block.

7 Conclusions

In this work we presented DiSR, a topology discovery approach to achieve deadlock freedom in self-assembled nanoscale networks. Results showed how a good coverage of the network can be obtained without requiring a centralized approach or a topology graph as input. Finally, a draft hardware implementation has been presented to evaluate the impact on the limited node size typical of the assumed scenario.

References

1. Ascia, G., Catania, V., Palesi, M., Patti, D.: Neighbors-on-path: a new selection strategy for on-chip networks. In Proceedings of the 2006 IEEE/ACM/IFIP Workshop on Embedded Systems for Real Time Multimedia, ESTMED '06, pp. 79–84. IEEE Computer Society, Washington, DC, USA (2006)
2. Bachtold, Adrian, Hadley, P., Nakanishi, T., Dekker, C.: Logic circuits with carbon nanotube transistors. Science **294**(5545), 1317–1320 (2001)
3. Cherkasova, L., Kotov, V., Rokicki, T.: Fibre channel fabrics: evaluation and design. In: Proceedings of the Twenty-Ninth Hawaii International Conference on System Sciences, vol. 1, pp. 53–62. IEEE, New York (1996)
4. Cui, Yi, Lieber, C.M.: Functional nanoscale electronic devices assembled using silicon nanowire building blocks. Science **291**(5505), 851–853 (2001)
5. Ebrahimi, M., Daneshtalab, M., Plosila, J., Tenhunen, H.: Minimal-path fault-tolerant approach using connection-retaining structure in networks-on-chip. In: Seventh IEEE/ACM International Symposium on Networks on Chip (NoCS), pp. 1–8. (2013)
6. Koibuchi, M., Matsutani, H., Amano, H., Pinkston, T.M.: A lightweight fault-tolerant mechanism for network-on-chip. In Proceedings of the Second ACM/IEEE International Symposium on Networks-on-Chip, pages 13–22. IEEE Computer Society, Silver Spring (2008)
7. Liu, C., Zhang, L., Han, Y., Li, X.: A resilient on-chip router design through data path salvaging. In Proceedings of the 16th Asia and South Pacific Design Automation Conference, pp. 437–442. IEEE Press, New York (2011)

8. Liu, Y., Dwyer, C., Lebeck, A.R.: Routing in self-organizing nano-scale irregular networks. J. Emerg. Technol. Comput. Syst. **6**(1), 3:1–3:21 (2008)
9. Mejia, A., Flich, J., Duato, J., Reinemo, S.A., Skeie, T.: Segment-based routing: an efficient fault-tolerant routing algorithm for meshes and tori. In: International Parallel and Distributed Processing Symposium, Rhodos, Grece, 25–29 April 2006
10. Patti, D., Nanoxim: nanonetwork simulator. https://code.google.com/p/nanoxim/
11. Patwardhan, J.P., Dwyer, C., Lebeck, A.R., Sorin, D.J.: Evaluating the connectivity of self-assembled networks of nano-scale processing elements. In: IEEE International Workshop on Design and Test of Defect-Tolerant Nanoscale Architectures (NANOARCH 05) (2005)
12. Pistol, C., Chongchitmate, W., Dwyer, C., Lebeck, A.R.: Architectural implications of nanoscale integrated sensing and computing. In Proceedings of the 14th International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS XIV, pp. 13–24. ACM, New York, USA (2009)
13. Sancho, J.C., Robles, A., Duato, J.: A flexible routing scheme for networks of workstations. In: High Performance Computing, pp. 260–267. Springer, Heidelberg (2000)
14. Seeman, C.N.: DNA engineering and its application to nanotechnology. Trends Biotechnol. **17**(11), 437–443 (1999)
15. Yan, Hao, Park, S.H., Finkelstein, G., Reif, J.H., LaBean, T.H.: DNA-templated self-assembly of protein arrays and highly conductive nanowires. Science **301**(5641), 1882–1884 (2003)

Automated Design of 5 GHz Wi-Fi FSS Filter

Pavel Tomasek

Abstract This article presents a technique for analysis and automated design of frequency selective surfaces. The approach allows to automate the whole process of the filter design and frees the users from the detailed knowledge of the filter design theory. Whole process of automation is implemented in Matlab. An optimisation of a band-stop filter for Wi-Fi communication on 5 GHz serves as a practical example. Therefore the goal is to design a band-stop filter which ideally does not transmit mentioned band of frequencies. The geometry of double-layer Jerusalem-cross serves as a structure to be optimized.

Keywords Frequency selective surface · Band-stop filter · Local optimisation · Planar periodic structures · Method of moments · Wi-Fi signal · 802.11

1 Introduction

Frequency Selective Surfaces (FSSs) are important spatial filters which can efficiently filter desired band of frequencies. Therefore these can play a significant role in electromagnetic related problems.

To briefly sketch the history, the beginning of FSS relates to Munk [1] which was the guru of this approach. In the last decade, the idea of FSS has spread out into many applications. Example of a band-pass FSS is in [2, 3] where the goal was to transmit GSM signals through energy efficient windows. The first FSS absorber was presented by Haupt [4], Knott and Lunden [5]. Kiani et al. [6] and Rafique et al. [7] proposed a novel and compact design to obtain stable frequency response by absorbing 5 GHz signals.

P. Tomasek (✉)
Tomas Bata University in Zlin, Nad Stranemi 4511, 760 05 Zlin, Czech Republic
e-mail: tomasek@fai.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_26,

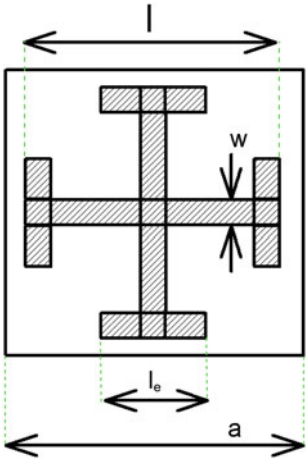


Fig. 1 Schema of a cell containing the Jerusalem-cross

In this paper the aim is to investigate the possibility of filtering of 5 GHz Wi-Fi signal, and to present a process of automation of filter design which frees the users from the detailed knowledge of the filter design theory.

The author has already experimented with filtering of 2.4 GHz Wi-Fi signal [8] used in devices of standard 802.11b and 802.11g where are more interferences, crowding, higher transmissivity and less expensive devices with respect to the 5 GHz wireless technology.

2 Statement of the Problem

Assume that there is a need to prevent transmission of Wi-Fi signal so that it cannot spread out of a given room.

A 5 GHz Wi-Fi device communicating under standard 802.11a and newer uses a specific channel which has frequency between 5.15 and 5.85 GHz [9]. Therefore the goal is to design a band-stop filter which ideally does not transmit mentioned band of frequencies.

3 Design of an Appropriate FSS

A double layer Jerusalem-cross is chosen as the schema to be optimized, see Fig. 1. The first reason for this choice was in potentially better reflection in comparison with a simple cross. The second reason was the relative simplicity of the model which can be modelled by rectangular elements. In the Fig. 1a represents the width and height of a cell, l is the total width and height of the Jerusalem-cross, w is the

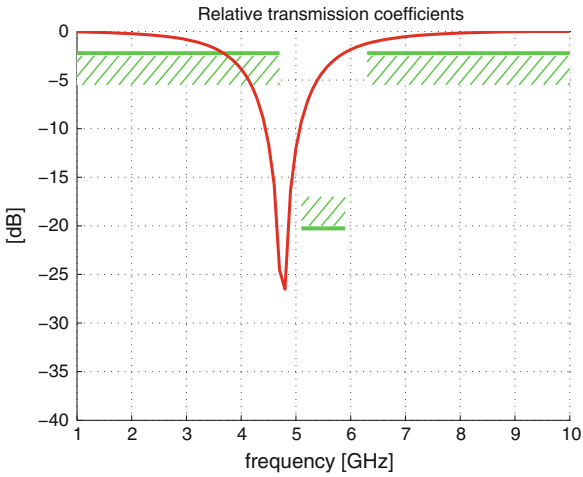


Fig. 2 Transmission coefficients of the initial FSS Wi-Fi filter (to be optimized)

width of an arm and l_e represents the length of the bar connected to the end of an arm.

The electrical conductivity of the metallization is 56 MS m^{-1} and the thickness is 17 μm. The relative permittivity of the dielectric layer is 1.0 and the thickness is 1.57 mm.

4 Optimization

A frequency range, an initial geometry with design variables (e.g. width and height of the arms of the cross) and optimization goals must be set before performing the optimization of an FSS filter.

The transmission coefficient depends on frequency and other parameters forming the parameter vector of the filter which specifies the geometry (defined by design variables). An optimization method searches for the set of parameters which satisfies the given objectives, at least approximately, being thus in a certain sense optimal.

An optimization goal is defined by a frequency range where the transmission coefficient must be lower or greater than a threshold value set by the user.

In our experiment three optimization goals were modelled (also presented in Fig. 2 together with results of initial configuration):

1. To pass frequencies from 1.0 to 4.7 GHz (threshold: −2.5 dB)
2. To stop frequencies from 5.15 to 5.85 GHz (threshold: −20.0 dB, this range relates to the Q factor equal to 7.86)
3. To pass frequencies from 6.3 to 10.0 GHz (threshold: −2.5 dB).

Table 1 Description of design parameters (LB and UB stands for lower and upper bound)

Parameter	Description	Initial value	LB	UB
a	The width and height of a cell (m)	0.025	0.01	0.04
w	The width of an arm (m)	0.0018	0.0006	0.003
k_1	The width parameter ($k_1 = l/a$)	0.85	0.7	1.0
k_2	The length parameter ($k_2 = l_e/l$)	0.35	0.2	0.5

The initial values of design parameters with lower and upper bounds are mentioned in Table 1 where $l = k_1 a$ and $l_e = k_2 l$.

In our work, optimization was performed numerically using an implementation of local optimizer Levenberg-Marquardt (a possible alternative is *fmincon* [10] or *fminsearchbnd* [11] which can be directly used in Matlab).

The settings of the optimization process:

- FunTol $= 10^{-3}$, this represents the threshold tolerance
- MaxIter $= 100$, this constant is the maximal number of iterations
- NormStep $= 0.06$, this is the constraint on maximal norm of a step
- Constraints on design variables a, w, k_1 and k_2 respect the lower and upper bounds mentioned in Table 1.
- Cost function used the method of moments [1, 12–14] to analyse and estimate the FSS transmission coefficients

All computations by optimization were based on perpendicular angle of incidence only.

In this study, we used FSSMR software [15] which was developed at Tomas Bata University in Zlin and which analyses the planar periodic structures and tries to optimize them with respect to the optimization goals.

5 Results

The optimization procedure results in a filter which suppresses well the central channels of 5 GHz Wi-Fi. The first and the last channels are partially transmitted (behind 5.15 and before 5.85 GHz), the attenuation is only around −14 dB what can be considered as not so perfect but still good. Perfect suppression of signals on these boundary frequencies can be processed in a further research. The final transmission coefficients are presented in Fig. 3. Furthermore, Fig. 4 presents the transmission coefficients of the final filter where the angle of incidence is not perpendicular but equal to 15 degrees. The process of optimization in Matlab took at about 30 h using an average computer.

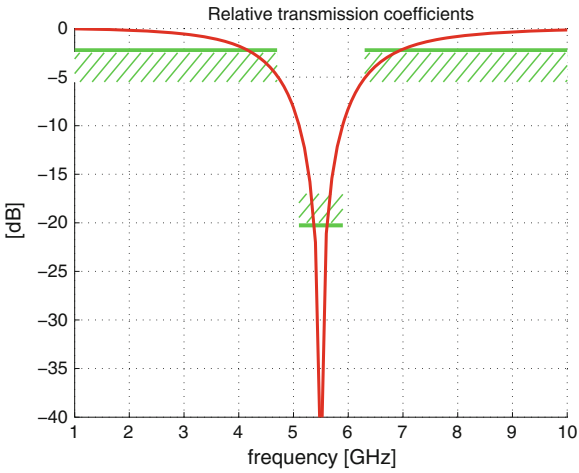


Fig. 3 Transmission coefficients of the optimized FSS Wi-Fi filter

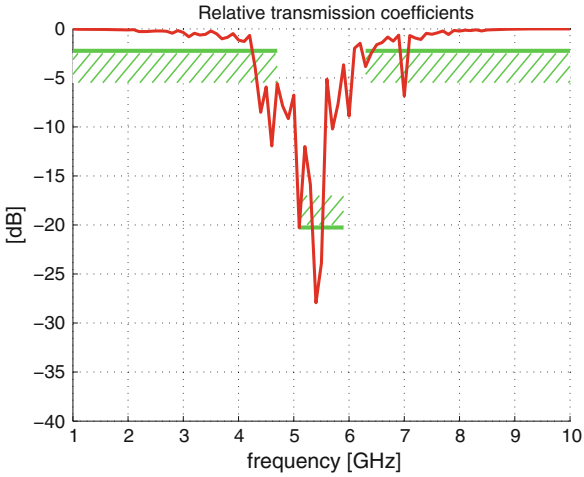


Fig. 4 Transmission coefficients of the optimized FSS Wi-Fi filter where $\theta = 15°$ (angle of incidence)

The optimized values of design parameters are presented in the list below:

$a = 0.0182$ m
$w = 0.0009$ m
$k_1 = 0.899$
$k_2 = 0.399$.

Furthermore, from the design parameters we can compute the lengths l and l_e in the following way:

$l = k_1 a$; $l = 0.01636$ m (the total width and height of the Jerusalem-cross)
$l_e = k_2 l$; $l_e = 0.00653$ m (the total length of an outer arm).

6 Conclusion

A method of optimization of an FSS filter was proposed and tested on a problem of filtering of 5 GHz Wi-Fi signal. An initial proposed solution (geometry of double-layer Jerusalem-cross) was optimized and corresponding transmission coefficients of optimized filter were shown.

It could be now simple to create a multi-layered FSS in which one set of layers is devoted to reflect the 2.4 GHz and the other one is aimed at reflecting the 5 GHz band. Thus the complete set of possibilities of Wi-Fi communication can be restricted.

The results presented in this paper are quite promising. Presented method seems to be suitable in design of any band-stop or band-pass filter and could help to find solutions of other complicated electromagnetic problems. Anyway, further work in this direction should prove this theoretical study by results of real measurements.

This work was supported by the internal grant of Faculty of applied informatics at Tomas Bata University in Zlin: IGA/FAI/2014/005.

References

1. Munk, B.: Frequency selective surfaces—theory and design. Willey, New York, USA (2000)
2. Kiani, G.I., Olsson, L.G., Karlsson, A., Esselle, K.P., Nilsson, M.: Cross-dipole bandpass frequency selective surface for energy-saving glass used in buildings. IEEE Trans. Antennas Propag. **59**(2), 520–525 (2011)
3. Rafique, U., Ahmed, M.M., Haq, M.A., Rana, M.T.: Transmission of RF signals through energy efficient window using FSS. In: Proceedings of the 7th International Conference on Emerging Technologies, pp. 1–4 (2011)
4. Haupt, R.L.: Scattering from small salisbury screens. IEEE Trans. Antennas Propag. **54**(6), 1807–1810 (2006)
5. Knott, E.F., Lunden, C.D.: The two-sheet capacitive Jaumann absorber. IEEE Trans. Antennas Propag. **43**(11), 1339–1343 (1995)
6. Kiani, G.I., Ford, K.L., Esselle, K.P., Wiley, A.R., Panagamuwa, C.L.: Oblique incidence performance of a novel frequency selective surface absorber. IEEE Trans. Antennas Propag. **55**(10), 2931–2934 (2007)
7. Rafique, U., Kiani, G.I., Ahmed, M.M., Habib, S.: Frequency Selective Surface Absorber for WLAN Security. In: Proceedings of the 5th European Conference on Antennas and Propagation, pp. 872–875 (2011)
8. Tomasek, P.: Automated design of frequency selective surfaces with the application to wi-fi band-stop filter. In: PIERS Proceedings. The Electromagnetics Academy, Stockholm, Sweden, Cambridge, pp. 221–224 (2013) ISSN 1559-9450
9. Pazin, L., Leviatan, Y.: Inverted-F Laptop Antenna With Enhanced Bandwidth for Wi-Fi/WiMAX Applications. Antennas Propag. IEEE Trans. **59**(3), 1065–1068 (2011). doi:10.1109/TAP.2010.2103036
10. The MathWorks Inc. Mathworks nordic: Find minimum of constrained nonlinear multivariable function—matlab [online]. http://www.mathworks.se/help/optim/ug/fmincon.html, [cit. 2012-12-20]

11. D'Errico, J.: fminsearchbnd, fminsearchcon—File exchange—matlab central [online]. http://www.mathworks.com/matlabcentral/fileexchange/8277-fminsearchbnd, 6 Feb 2012 [cit. 20 Dec 2012]
12. Chan, R., Mittra, R.: Techniques for analyzing frequency selective surfaces-a review. Proc. IEEE **76**(12), 1593–1615 (1988)
13. Wu, T.K.: Frequency selective surfaces and grid arrays. Willey, New York, USA (1995)
14. Wan, C., Encinar, J.A.: Efficient computation of generalized scattering matrix for analyzing multilayered periodic structures. IEEE Trans. Antennas Propag. **43**(11), 1233–1242 (1995)
15. Gona, S., Kresalek, V.: Development of a versatile planar periodic structure simulator in MATLAB. In: Conference on microwave techniques (COMITE), vol. 14(1) (2008)

Obstacle Detection for Robotic Systems Using Combination of Ultrasonic Sonars and Infrared Sensors

Peter Janku, Roman Dosek and Roman Jasek

Abstract An obstacle detection became one of the most important tasks in a robotic system development. DistanceDetector is a device which can detect large obstacles by utilizing a combination of ultrasonic sonars and infrared sensors. This paper deals with the DistanceDetector description and reveals its hardware and firmware structure, used technologies and provides a simple use case scenario.

Keywords Obstacle detection · Ultrasonic sonar · Infrared sensor · Robotic systems

1 Introduction

The role of obstacle detection in robotic system is to increase autonomy and decrease the probability of human error. Sensors for detecting obstacles can be generally divided into two groups—the active and passive sensors. The active sensors, such as radars and laser range finders have high performance and can give distance measurement directly. They are, on the other hand, expensive, sensitive to weather conditions and have problems with detecting small obstacles. The passive sensors like monocular and stereo vision can provide more information about environment, but requires more processing power [6].

P. Janku (✉) · R. Dosek · R. Jasek
Faculty of Applied Informatics, Tomas Bata University in Zlin, Nad Stranemi 4511, 760 05 Zlin, Czech Republic
e-mail: pjanku@fai.utb.cz
URL: http://www.utb.cz/fai

R. Dosek
e-mail: dosek@fai.utb.cz

R. Jasek
e-mail: jasek@fai.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_27,

There are several known principles of active obstacle detection which are based on emitting electromagnetic or mechanical waves and detecting their reflection. Among the most used sensors are: infrared sensors, ultrasonic sensors and radars. By combining several types of sensors it is possible to increase robustness of system while eliminating drawbacks of chosen sensors [7]. Ultrasonic range sensors have typically a 15–30° beam angle—to cover all directions, one of the following strategies can be employed. First strategy is to place the ultrasonic sensors in a ring of 12 or 24 sensors depending on desired accuracy [5]. The second solution lies in placing one or more sensors onto rotating part and sampling sensor results over given period to cover all directions [4].

Non autonomous robots are typically controlled via RC transmitter and receiver, generating PWM[1] signal which is simple to process. It is therefore possible to read the controlling signal directly and augment it to void collision with obstacles.

The proposed system is not intended to replace human element altogether, but to help user with robot control. Developed system should be easy to use to allow robot control by people with little or no training. System does not have to accurately pinpoint all obstacles, it's main priority is to avoid collision with large obstacles, such as walls. Because the ultimate aim is to allow it's usage in integrated rescue services, the solution shouldn't be expensive. The final solution uses combination of 4 ultrasound sensors for long range and 4 infrared light sensors placed on servos for short range detection.

2 Used Technologies

2.1 Hardware Parts

The STM32F407VE[2] microcontroller was selected as the compute base of the obstacle detection system developed in this research. It consists of Cortex-M4 core and a number of other built-in peripherals. Therefore, the developed device can achieve high computing performance, whereas the current consumption and the external size can stay very compact [2].

As was mentioned in previous section the combination of ultrasonic-base sensors and infrared sensors was used as the detection device. The ultrasonic-base sensor can detect obstacles on long range without high resolution; in contrast, the infrared sensor is more accurate on small range.

The LV-MaxSonar-EZ0[3] unit was selected as the ultrasonic-base sensor. The ultrasonic transmitter/receiver is placed on this unit as well as small compute device. Thank to this the EZ0 sonar module can provide an automatic distance

[1] Pulse Width Modulation.

[2] http://www.st.com/web/en/catalog/mmc/FM141/SC1169/SS1577

[3] http://www.maxbotix.com/Ultrasonic_Sensors/MB1000.htm

measurement; moreover, the measured distance, detected by this sonar, can by read by using it's digital interface or by reading it's analog voltage output level.

The Sharp 2Y0A02[4] device was selected as the infrared sensor. It provides combination of an infrared diode and an infrared detector for the distance measurement. The measured distance can be read as a voltage level on analog output of this sensor.

Parts described above were supplemented by SD-card holder for data storing, small I2C EEPROM memory and three communication interfaces (CAN, USB and PWM). Other parts used in the developed device design are commonly used discreet parts.

2.2 Software Parts

The FreeRTOS was selected as a base for the firmware development. It is a real-time operating system, which is distributed as an open source. Most of application parts are implemented in its "tasks" which ensures the system reaction in a final time [3].

The CMSIS[5]-base libraries supplied by the MCU manufacturer were used for internal peripherals control. These libraries contain set of easy to use procedures for setting and using internal peripherals including Cortex-M4 and DSP core [1].

3 System Design Description

The DistanceDetector system was developed in two nearly independent steps. In the first step the hardware parts of device were designed, manufactured and completed. Consequently, the internal firmware development was made in the second step.

3.1 Hardware Design

As can be seen on Fig. 1, the obstacle detection system was built as a one-board electronic device. As was discussed in the previous section, the STM32F407 was selected as the MCU.[6] This unit was connected to the cascade of ultrasonic-base

[4] http://sharp-world.com/products/device/lineup/data/pdf/datasheet/gp2y0a02_e.pdf

[5] Cortex Microcontroller Software Interface Standard.

[6] Main Compute Unit.

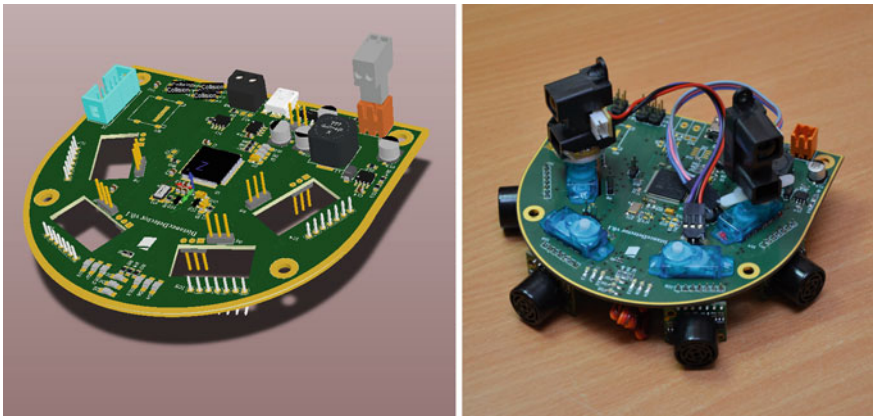


Fig. 1 *Left* Computer animation of hardware design. *Right* Manufactured device

sensors using an internal multichannel A/D converter; furthermore, the ultrasonic sensors in the cascade were connected together by using sensor's trigger inputs and outputs. Due to this, only one sensor can provide measurement in one time. As a result, there is no crosstalk between the sensors.

The infrared-base sensors were connected to the MCU by using internal an A/D converter too. Because the infrared sensors have significantly narrower detection field, they had to be mounted on servo-motors. Thank to this, they can be rotated around a vertical axis and thus they can cover the same field as the ultrasonic sensors. Servomotors were placed directly into cutouts in main board and were connected to the timer-compare outputs of MCU; therefore, the PWM signal can be generated by internal timers which has significant lower impact on firmware computation complexity than PWM signal generated by firmware.

All discussed sensors were placed on board with special attention on their position. As can be seen on Fig. 2 the final angle of detection field is exactly 180°.

3.2 Firmware Design

Because of usage of the FreeRTOS as a main part of the produced firmware, all features of this firmware were separated into these three categories: automatic procedures, user tasks and timers. The Fig. 3 shows the internal firmware structure used in the DistanceDetector device.

The automatic procedures represents all features which can be done by the internal peripherals without the main firmware attention or done in a short time in the system interrupts. These procedures are used for generating PWM signals for servo controls, converting analog sensor outputs into digital values and providing base external communications (USB, CAN).

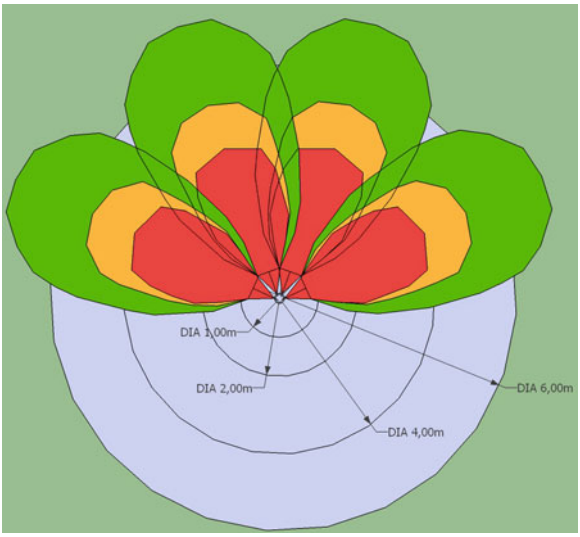


Fig. 2 The radiation diagram of ultrasonic sensors placed on the developed device

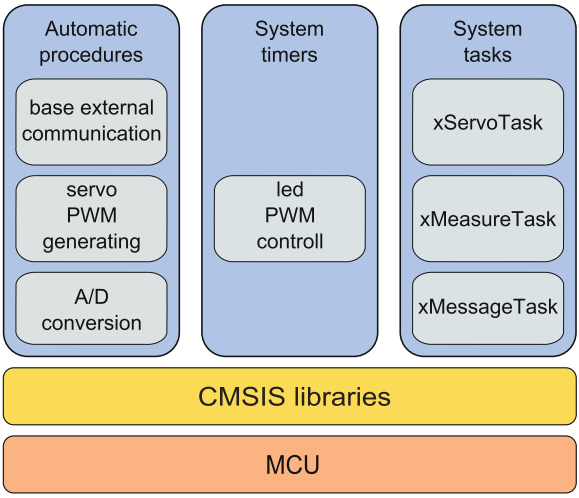


Fig. 3 Developed firmware structure

System timers includes simple firmware parts, which have to be run periodically. Developed firmware uses one system timer which implements onboard LED light in PWM mode.

System tasks represents critical operations which have to be done in final time. Therefore, three tasks are used in developed firmware.

The first task called **xServoTask** provides infrared sensors rotating. It calculates the right sensor angle, reads the measured distance and makes a stamp into internal memory or an SD card.

The second task called **xMeasureTask** collects all measured distances; moreover, it calculates all necessary informations and provides programmed reactions. For example it sets LED status, it calculates changes in PWM outputs (if PWM communication is used) or it sends informations through other interfaces.

The last task called **xMessageTask** handles large messages sent through communication interfaces. The size of these messages can be set in command line interface provided through USB. For instance, the massage can be only the shortest measured distance with corresponding angle or it can be large table periodically refreshed including all measured values.

3.3 Distance Calculation

As was mentioned in previous sections ultrasonic and infrared sensors are connected to the MCU through the internal A/D converter. This converter is set by the software to work with a 12b resolution and it's reference is connected to the $U_{cc} = 3.3$ V. The input voltage U_{in} measured by the A/D converter can be calculated as

$$U_{in} = \frac{U_{cc}}{2^{12}} * N \tag{1}$$

where N is the A/D converter output.

The ultrasonic sensor analog output voltage is generated with 9b resolution. Therefore the the voltage level is defined as

$$U_{out} = \frac{U_{cc}}{2^{9}} * D_{inch} \tag{2}$$

where D_{inch} is detected obstacle distance in inches. Because the analog output is directly connected to the A/D converter input, it can be assumed that $U_{out} = U_{in}$.

$$\frac{U_{cc}}{2^{12}} * N = \frac{U_{cc}}{2^{9}} * D_{inch} \tag{3}$$

The final relation between a measured distance in inch D_{inch} and the value measured by A/D converter N:

$$D_{inch} = \frac{2^{9}}{U_{cc}} * \frac{U_{cc}}{2^{12}} * N = \frac{N}{2^{3}} \tag{4}$$

The infrared sensor relation between measured distance and output voltage is shown in Fig. 4. For DistanceDetector purposes this graph can be separated into the three nearly linear parts in which the output voltage is defined as:

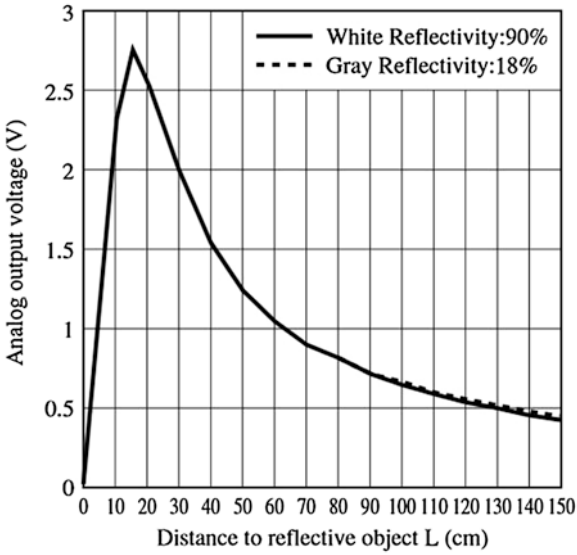


Fig. 4 Infrared sensor—analog output voltage versus distance to reflective object

$$U_o = U_{omin} + \frac{\Delta U_o}{\Delta L} * (L_{max} - D_{cm}) \tag{5}$$

where U_o is output voltage level, ΔU_o is the part voltage difference and ΔL is the part distance difference and D_{cm} is real measured distance in centimeters. Because the analog output is directly connected to the A/D converter input, it can be assumed that $U_{out} = U_{in}$.

$$\frac{U_{cc}}{2^{12}} * N = U_{omin} + \frac{\Delta U_o}{\Delta L} * (L_{max} - D_{cm}) \tag{6}$$

$$D_{cm} = \frac{\left(\frac{U_{cc}}{2^{12}} * N - U_{omin}\right)\Delta L}{\Delta U_o} - L_{max} \tag{7}$$

4 Practical Tests

4.1 Desktop Application

To test the device, we developed a desktop application in C++ language and Qt library. This application makes use of virtual COM port accessible through USB port of device. Subsequently, received data are visualized in way which allows easy orientation for users of application.

For each ultrasonic sensor there is an arc rendered with an angle of 45° and radius matching the measured distance. Furthermore, results, measured by the infrared sensors, are visualized in form of polyline.

4.2 Measured Results

The system, developed in this research, was tested in various scenarios and in many use-cases. The first group of all performed tests was aimed to review sonar sensor's obstacle detection ability. These tests revealed a significant level of noise in measured distances. This noise was induced by fake reflections inside detection field. To reduce this noise, it was necessary to implement software filtrating on measured data. Furthermore, the required filtering algorithm had to be simple enough to fit memory and processing constrains inside device. Consequently, the moving average algorithm was found sufficient for this noise level reduction. It is a very simple algorithm which compute the average from the last four measured distances. Finally, these tests recognized that the real minimal detected obstacle is tube with 4 cm radius and real maximal detection distance is nearly 5 m. These sensors are able to detect an obstacle outside this dimensions but the error probability is significant.

The second group of tests was aimed to confirm performance of infrared sensors. These tests reveals that two sensors can sufficiently cover scanned field. Therefore, next part of testing was made by using just two sensors. These measurements confirmed that the noise level in infrared distance detection is significantly lower than one in data obtained by sonar sensors. Thanks to this, the moving average using only two last values can be performed. The maximal measured distance in this case was 1.50 cm. This value is significantly lower than one obtained by the ultrasonic sensors. Furthermore, infrared sensors gives huge amount of information about detected obstacle. As can be seen in the Fig. 5, the infrared sensors can determine the obstacle position, it's scale and outline.

5 Limitations and Future Applications

The DistanceDetector is a powerful but cheap device for obstacle detection. Despite it's limitation it can bring huge amount of relevant data for further processing. This project was concentrated to basic hardware and software development for obstacle detection. Therefore, all software parts were programmed by using conservative methods.

At the other side this applications can be basic part for further research in which modern computing methods can be used. For example discrete digital filters can be used for noise reduction or neural networks can be used for obstacle recognition.

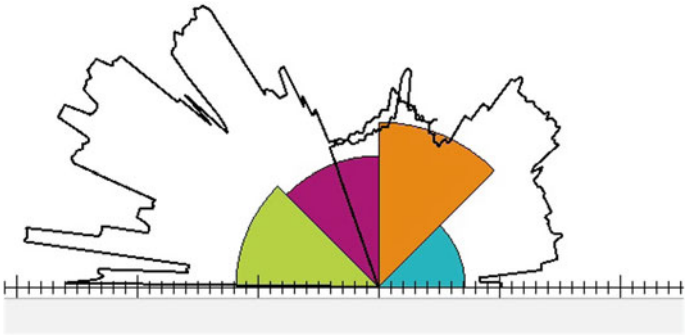


Fig. 5 DistanceDetector test result. Color sections are results from sonar sensors, polyline is result from infrared sensors

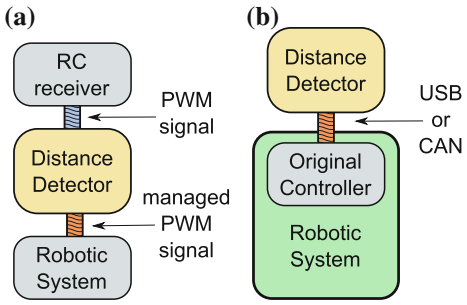


Fig. 6 Distance detector use-cases. **a** DistanceDetector inserted between classical RC receiver and robotic system. **b** Distance detector directly connected to the robotic system original controller

The data concentrated by DistanceDetector can be source of informations for interactive map creation algorithms.

All parts of the developed firmware are written in language C; therefore, all algorithms used now and in the future can be slightly divide between DistanceDetectors's compute unit and supervised system.

5.1 System Integration

The developed DistanceDetector device can be implemented into existing robotic systems in two ways. As can be seen in the Fig. 6a when the existing robotic system use standard PWM signal to transmit command (for example if it uses typical RC receiver), the distance detector can be connected into the PWM signal path. In this case, the PWM signal is sampled by the DistanceDetector, it is augmented by this device and then regenerated into robotic system. Size of this impact can be set through the USB interface.

The Fig. 6b shows the situation, when the existing robotic system uses intelligent controller with an USB or a CAN interface. In this case the Distance-Detector is connected to original controller by this interfaces and it sends all requested information immediately.

6 Conclusion

In this paper, we described an integrated board for detecting obstacles that is able to cover 180° angle. It can successfully detect obstacles that are up to 6.5 m away. When the detected object is closer than 1.5 m away, the measured distance is further refined by infrared sensor. However, due to limitations imposed by selected sensors, small obstacles may not be detected. Furthermore, the infrared sensors are limited by minimal reflectance of obstacles. The device can be easily integrated into various robotic systems such as quadracopters or robotic vehicles.

Acknowledgement This paper is supported by the Internal Grant Agency at TBU in Zlin, Project No. IGA/FAI/2013/022.

References

1. CMSIS—cortex microcontroller software interface standard—ARM. http://www.arm.com/products/processors/cortex-m/cortex-microcontroller-software-interface-standard.php
2. Cortex-m4 processor—ARM. http://www.arm.com/products/processors/cortex-m/cortex-m4-processor.php
3. FreeRTOS—market leading RTOS (real time operating system) for embedded systems with internet of things extensions. http://www.freertos.org/
4. Kalmegh, S., Samra, D., Rasegaonkar, N.: Obstacle avoidance for a mobile exploration robot using a single ultrasonic range sensor. In: Proceedings of 2010 International Conference on Emerging Trends in Robotics and Communication Technologies (INTERACT), pp. 8–11 (2010)
5. Kim, S., Kim, H.B.: High resolution mobile robot obstacle detection using low directivity ultrasonic sensor ring. In: Huang, D.S., Zhang, X., Garca, C.A.R., Zhang, L. (eds.) Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence, pp. 426–433. No. 6216 in Lecture Notes in Computer Science, Springer, Berlin Heidelberg (2011). http://link.springer.com/chapter/10.1007/978-3-642-14932-0_53
6. Miled, W., Pesquet, J.C., Parent, M.: Robust obstacle detection based on dense disparity maps. In: Daz, R.M., Pichler, F., Arencibia, A.Q. (eds.) Computer Aided Systems Theory EUROCAST 2007, pp. 1142–1150. No. 4739 in Lecture Notes in Computer Science, Springer, Berlin Heidelberg (2007). http://link.springer.com/chapter/10.1007/978-3-540-75867-9_143
7. Perrollaz, M., Labayrade, R., Royere, C., Hautiere, N., Aubert, D.: Long range obstacle detection using laser scanner and stereovision. In: 2006 IEEE Intelligent Vehicles Symposium, pp. 182–187, 00051 (2006)

Automatic Sensor Configuration for Creating Customized Sensor Network

Ketul B. Shah and Young Lee

Abstract A sensor network is expected to provide effective delivery of its services by taking an appropriate action based on one or more situations that it senses in the environment. However, due to the dynamism of application requirements and user context, it is often required to re-configure services from a sensor network to meet specific application needs. This paper is an attempt to enable dynamic adaptation of sensor network services with a web-based database consists of sensors' MAC address, vendor ID and data frame. We propose a semantics-based architecture where ASC connects with sensor and matches the received data with the database and connects the sensor with the mobile device. ASC works on full-duplex communication to observe sensors as well as control mobile devices. Wide ranges of android and java libraries are capable to manipulate embedded systems from smart-phones. It's application in the field of education, security and surveillance, environment research and military. In this paper, we represent the structure and functioning of ASC for Bluetooth devices and its applications.

Keywords Android · Java · Sensor-network · Full-duplex communication · Smart-sensors · Bluetooth · Wi-Fi · Web-server · Web-based database · Client-server communication · HTTP request · HTTP response

K. B. Shah (✉) · Y. Lee
Electrical and Computer Science Department, Texas A&M University-Kingsville,
700 University Blvd, Kingsville, TX 78363, USA
e-mail: ketul.shah@students.tamuk.edu

Y. Lee
e-mail: young.lee@tamuk.edu

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_28,

1 Introduction

Recent years have witnessed the emergence of computationally-enabled sensors, along with the rapid development and deployment of applications that use sensor data [1, 2]. A sensor network perceives the environment, monitors different parameters and gathers data according to an application purpose. The capabilities of a sensor network are not just limited to observing and forwarding raw sensor readings. Sensor networks revolutionize sensing in a wide range of application domains. They provide different services, ranging from home automation to process monitoring, healthcare analysis, weather forecasting, military situation awareness, and traffic control.

Sensor networks are often used in uncontrollable environments, whereby users may not have precise information about the sensing data or network characteristics [3]. Moreover, these properties can change over the lifetime of a network deployment, due to events such as signal quality degradation, isolated failures, resource addition, and sensor movement. As a consequence, sensor services must be re-configured to make use of the new context information and resources. There should be the provision to autonomously adapt, i.e. *auto-configure*, the sensor network services according to application scenarios and context, with enhanced accessibility and reusability.

The auto-configuration feature in a sensor network would enable autonomous structuring of contextual information and resources thus making them available to services. It is proposed as one of the means to make a system scalable and robust in the presence of changes, supporting dynamic adaptation [4]. It also allows service customization and supports employing semantic technologies. An auto-configurable sensor network service can be built through a combination of process changes, technology evolution, architecture and open industry standards.

Although there are several research works in the autonomic computing [5, 6], model-driven engineering [7, 8], overlay networks [9], service computing [10, 11], and sensor network [3, 12, 13] domains to deal with the self-properties of a system, there is the need for a detailed architecture for auto-configuration of sensor services. Our research is a step towards addressing this issue by making data frame global to web server.

We propose a rich and powerful semantics-based architecture where ASC is developing a single application with access to web-server and web-based database that connects to a sensor or robot fetches required data from the database and start communicating with the sensor or robot. So ASC can be seen as a kind of application that downloads driver software upon connected with the sensor or robot.

In Fig. 1, ASC block diagram is mentioned. It consists of mainly five building-blocks: Web-server, Sensor-array, Physical entity (Robot), Mobile device and Web-based Database. Mobile device is can be seen as a user-end. Web-server is the main entity of ASC project. Web-based database is forbidden from direct use from user-end. Only web-server is capable to communicate with the database. Sensor-array is a physical smart-sensor network. Here, note that Robotic device

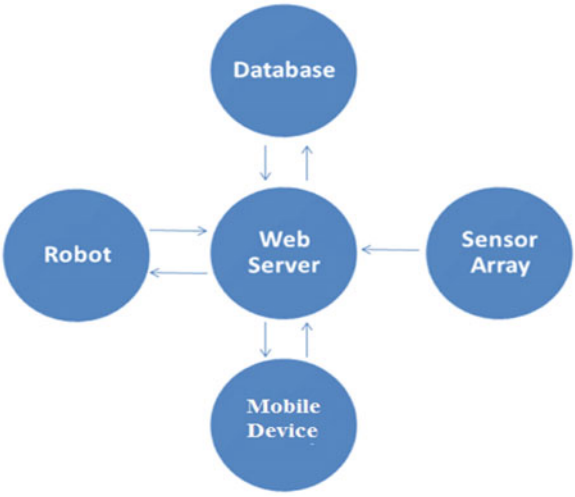


Fig. 1 Block diagram of an ASC

and sensor array are two different concepts but they can work together or separately. Further in this paper, all the building blocks are explained in detail.

The rest of this paper is structured as follows. Section 2 sets a comparative analysis of related work to highlight the novelty of our work. It is followed by sets the stage by detailing design considerations and our approach for auto-configuration of sensor services in Sect. 3. The proposed architecture and associated research challenges are presented in Sects. 4 and 5, respectively. The paper is concluded in Sect. 6.

2 Related Work

Most existing methodologies for system and service configurations are focused on a specific engineering technique. Some of them are largely not extensible and do not support dynamic adaptation to the specific characteristics of individual application scenarios and context.

Semantically-enabled Heterogeneous Service Architecture and Platforms Engineering (SHAPE) [14] provide an integrated development environment that brings together the Model-Driven Engineering (MDE) methodology with the Service-Oriented Architecture (SOA) paradigm. The Real World Internet [15] project focuses on the management, scalability, and heterogeneity of devices and users. It aims to facilitate the dynamic creation of context and actuation services from elementary sensing, actuation, processing and reuse of sensor resources for a large number of applications. The OPPORTUNITY [16] project aims to build goal-oriented sensor assemblies that are spontaneously arising and self-organizing to achieve a common activity or context recognition goal. Hydra [17] follows a semantic Web-based self-management approach, supported by a set of self-

management context ontology. The CONNECT [18] project focuses on enabling automated protocol mediation through a formalization technique to enable seamless system composition. The SANY consortium [19] provides a standard open architecture and a set of basic services for integration of sensors, sensor networks, and sensor services. It also recognizes the OGC's Sensor Web Enablement (SWE) [20] as one of the key technologies to support self-organizing and self-healing in sensor networks.

While our work is related to these research projects, we differ in that the service customization and configuration techniques in the above projects are limited to low-level technical aspects, whereas we seek to insulate client applications from the low-level details. Some of the above works use SensorML or XML-based descriptions of sensor configurations, which is not usable in our context due to not supporting reasoning and vocabulary of semantic descriptions.

In recent years, a few research work have explored the use of semantics for sensor networks and publish/subscribe systems, such as semantic-based service framework for query processing [21], sensor plug and play [22], semantic filtering in an XML-based publish/subscribe infrastructure [23], an ontology-based publish/subscribe system [24], semantic-enabled publish/subscribe middleware [25, 26], and semantic-based publish/subscribe systems [27, 28]. While they are appealing, none of them focuses on the auto-configuration aspect as we do. Many of them do not make use of reasoning or domain knowledge in anthologies to aid the identification of semantically relevant service configurations and matching them to subscribers. Moreover, most of them suffer from scalability limitations with the increase of system size, and usually do not work well under a large number of application subscriptions and a high volume of service configurations. In our work, we aim to address these limitations.

3 Auto-configuration of Sensor Network Services

The auto-configuration ability can assist a sensor network service to dynamically adapt to the changing environment, including the deployment of new components or the removal of existing ones, or sudden changes in the system characteristics. These variations can happen either as random, periodic events or gradual evolutionary changes in the system. The underlying principle for building auto-configuring sensor network services is to provide fundamental mechanisms upon which other networking and system services may be spontaneously specified and reconfigured [29]. To make a sensor network service auto-configurable, the following conditions must be met:

- The tasks involved in the configuration of a service are automated.
- The automation process is initiated based on situations that can be observed or detected in the sensor network system.
- An authoritative entity in the sensor network must possess sufficient knowledge of sensor network service configuration, resources, policies, and observed data.

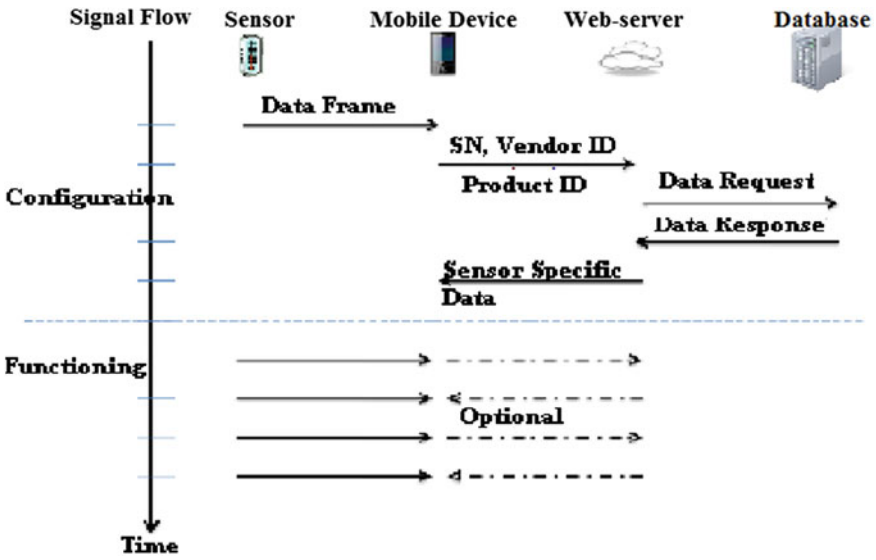


Fig. 2 Configuration process of an unknown sensor

On satisfying these conditions, it is possible to assemble the automated functions in a set of composed processes to allow a sensor network service to be auto-configurable. These processes can be governed to collect necessary details from the system, analyze them to determine the required configuration changes for a sensor service, create a plan or action sequence specifying the necessary changes, and perform those actions.

3.1 Our Approach

Semantic technologies [30] are often used to enable sensor network services and applications by providing a universally accessible platform that allows data to be shared and processed by automated tools, and by providing machine-understandable semantics of sensed data for automatic information processing and exchange. Figure 2 shows automatic configuration of smart sensor with ASC. Here, note that if sensor is not automatically configured, it is possible to configure it manually in ASC.

Each sensor in the market has one of these identities attached with it: (1) Serial Number, (2) Vendor ID, (3) Product ID or (4) MAC Address. If we have one of these three parameters, it is possible to identify the sensor and its operating characteristics. We have implemented this idea to make ASC compatible with any type of wireless digital sensors. Idea here is when sensor connects with mobile device; it gets these three parameters from the sensor and sends it to the Web-server. It is here assumed that a web-based database has all sensor data according

to these three parameters. So server sends these parameters to the database and requests the data frame. In response to this request database sends the sensor data frame and this in turn sent to the mobile device. Here we assume that a kind of driver software is installed on mobile device for that particular sensor. So the application is ready to communicate with the sensor. Following code, when simulated, finds the name and address of sensor to be connected:

```
BluetoothAdapter mBtAdapter = BluetoothAdapter.getDefaultAdapter();
mBtAdapter.cancelDiscovery();
String name = ((bluetoothDevice) v).getName().toString();
String address = name.substring(info.length() - 17);
IMOSEActivity.sensorAddress = address;
```

Once device name is obtained, it is sent to the server. Server searches for the data frame for the sensor name received and returns back the sensor data frame. If no data is found, it sends null data. In this case a manual configuration is required. Once data frame is received, following code will connect the sensor with the mobile device.

```
Intent intent = new Intent();
intent.putExtra(EXTRA_DEVICE_ADDRESS, address);
setResult(Activity.RESULT_OK, intent);
finish();
```

4 Proposed Architecture

The aim of auto-configuration is that the service will re-configure itself so as to again either satisfy the changed application context, scenario, and/or environment. The requirement specification for an auto-configurable sensor network service is not only functional behavior, but also those non-functional properties such as response time, performance, reliability, security, and that requirement may well include optimization. Based on the design considerations in Fig. 2, we present architecture for auto-configuration of sensor services, comprising two key layers.

4.1 Web-Server and Web-Based Database

Web server is the main part of the ASC application. As smart sensors and smartphones both are capable of network connectivity, web-server can be used to manipulate all ASC functions. Earlier web-servers used in other mobile applications are static and can communicate with mobile application in pre-defined

manner and in pre-defined data types [31−46]. So ASC web-server is designed to be as much flexible as possible to communicate with any type of data. It can be a sensor reading of type integer, string, single character, float, double or it might be a command dispatched to a mobile robot.

ASC web-server works as a data exchanger and transmits whatever data it receives. Data received by web-server is also stored to a web-based database. It responds to the request dispatched by either mobile device or sensors. When sensor is communicating directly with server, it stores sensor data to database and when HTTP request comes from mobile device, it gives sensor readings to the mobile device with HTTP response. In case of controlling robot, it receives commands from mobile device as HTTP request and dispatches the same command to robot as HTTP response and vice and versa.

This database works as a log register for ASC application. All data transactions done by ASC are stored by database in correct order with date and time; so required data can be pulled out from the database when needed. This database is not accessible directly by any other entity. Database is solely connected to web-server and responds only to requests dispatched by web-server.

4.2 Connecting to Physical Entity (Robot)

ASC, as mentioned earlier, also supports full-duplex communication through Bluetooth or Internet. ASC is capable of sending data to or receiving data from remote hardware (technically a robot) that is capable of receiving data through Bluetooth or Internet. In case of Bluetooth, ASC sends data directly to the Bluetooth module of the remote hardware. Web-server or Web-based database does not play any role here. But in case of Internet, ASC sends data to the web-server, which in turn passes that data to remote hardware. However, web-based database does not play any role here. Note that, Web-based Database is used only when connecting to an unknown sensor for the first time. Once driver is installed, it is not used at all in further operation.

5 Research Issues and Engineering Challenges

Several research and engineering challenges are associated with the lifecycle of an auto-configuring a sensor network service. They lie in the observed measurements, in the construction of sensor service configurations, in the generation and selection of alternative configurations and action plans, and overall in the operational activities for ongoing adaptation during the lifetime of a sensor network. This lifecycle (Fig. 3) begins with the design and implementation, proceeds to installation, configuration, monitoring, upgrading, and ceases in decommissioning of a sensor service.

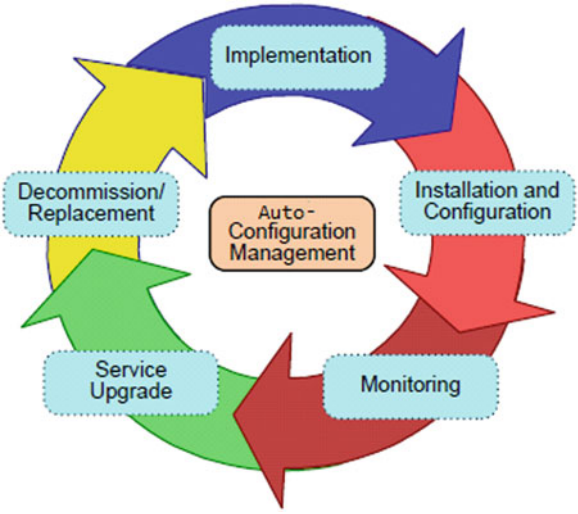


Fig. 3 Lifecycle of a self-configurable sensor network service

Implementation: This phase involves service design, testing and verification. The runtime infrastructure should leverage distributed components to create an optimal configuration according to both application requirements and context. It should also allow uniform access to data by applications, irrespective of the data format, source or location.

Installing and configuring: This phase involves bringing up a service to an operational state with minimal user involvement. It requires explicitly stating a user subscription to correspond to a published configuration in a formal request language. To enable this, a sensor service will entail a bootstrapping process that begins with the configurations registering itself with the AE. The service may also interact with AE to discover other services and dependencies it needs to complete its initial configuration.

Monitoring: This phase is included in the auto-configuration management, governed by AE. It involves monitoring configuration changes for sensor network services. This is an essential stage for enabling auto-configuration ability for a service. When coupled with event correlation or decision theory, the monitored information is useful for problem identification and recovery during system faults.

Service upgrade: The auto-configuring sensor network services may require upgrading them from time to time. They may subscribe/interact to an alert service that provides information on the availability of relevant upgrades and decide for themselves when to apply the upgrade, possibly with guidance from AE.

Decommission/replacement: Alternative to an upgrade process, sensor network services could be implemented afresh as part of a system upgrade, removing outdated services only after the new ones obtain a proper working status.

Lifecycle management: AE performs simultaneous activities to schedule and prioritize operations. A user interacts with a service via an appropriate interface provided by AE and retrieves the service description directly from the service. As

Name	MAC Address	Product ID	Vendor ID	Heart-rate	Beat Count	Battery Level
HXM-BT	xx:xx:xx:xx	xx:xx	xx:xx	12, 2	14, 2	16, 2

Fig. 4 Sample Web-based database showing Zephyr Heart rate monitor sensor details

per the node composition rule, the sensor service is configured and invoked according to application scenarios and context. During the lifetime of the system, services are dynamically reconfigured in response to sensed observations and system changes.

6 Conclusion and Future Work

ASC is successfully simulated on a heart-rate monitor sensor manufactured by Zephyr Technology Corporation [33]. This sensor takes reading of heartbeats and transmits it over Bluetooth. According to program created on ASC, application interfaces with Zephyr sensor and receives these readings from it. All of the simulations are carried out in Nexus-7 tablet manufactured by Asus Company. Simulations are carried out on Android 4.2 Jelly Bean operating System. Some functions of ASC application are dependent on Hardware on which it is running. It is assumed here that the sensor data is pre-defined in the Web-based Database.

Step-by-step Working:

1. First ASC searches for Bluetooth sensors.
2. Once searching completes, it connects to the sensor.
3. Once sensor is connected, it receives one of the following details given in first four columns.
4. ASC sends these data to Web-server which matches received data with data store in Web-based database (see Fig. 4).
5. If match is found, it returns rest of columns until a blank column is detected to the Mobile device.
6. Received data is manipulated and then sensor is successfully configured with the mobile device.

Above sample database displays the values for the Zephyr heart rate monitor sensor. This simulation is successfully conducted on ASUS Nexus 7 tablet. As shown in database first four columns are used in comparing data received from the sensor. For now, consider only first column and ignore rest three columns. Once a match is found, rests of columns (until a null column is found) are sent to the

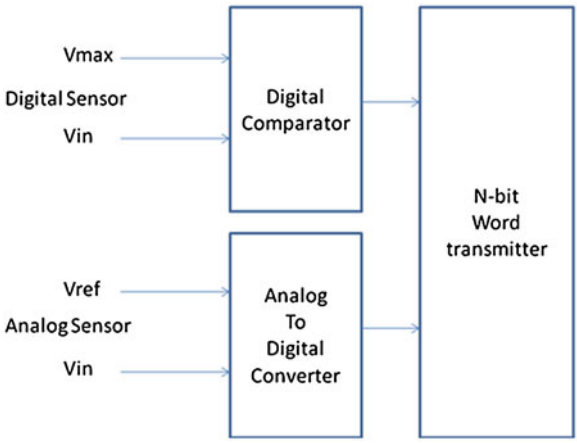


Fig. 5 ASC device

mobile device. In database first column displays the title for the data to be displayed to the user in ASC. The second row displays the location of the data. For example for Heart-rate it is "12, 2". That means whatever data frame the sensor is transmitting to the mobile device, the heart-rate information is on 12th byte and it is 2 bytes long. So ASC can manipulate the frame and find corresponding data from data-frame by fetching 12th and 13th byte.

Moreover, current version of ASC is capable to communicate with only smart sensors. In future ASC device (see Fig. 5) is to be developed which will be capable of interfacing small-scale electrical sensors -which are not smart-sensors. This device will consist of a digital comparator, Analog to Digital Converter and a wireless transmitter.

References

1. Akyildiz, I., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A survey on sensor networks. IEEE Commun. Mag. **40**(8), 102–114 (2002)
2. Yick, J., Mukherjee, B., Ghosal, D.: Wireless sensor network survey. Comput. Netw. **52**(12), 2292–2330 (2008)
3. Balani, R., Han, C.C., Rengaswamy, R.K., Tsigkogiannis, I., Srivastava, M.: Multi-level software reconfiguration for sensor networks. In: Proceedings of International Conference on Embedded Software (EMSOFT'06), pp. 121–130. ACM Press, NY (2006)
4. Kramer, J., Magee, J.: Self-managed systems: an architectural challenge. In: Proceedings of International Conference on Future of Software Engineering (FOSE'07), pp. 259–268 (2007)
5. IBM: An architectural blueprint for autonomic computing. White Paper, June 2005
6. Marsh, D., Tynan, R., O'Kane, D., O'Hare, G.P.: Autonomic wireless sensor networks. Eng. Appl. Artif. Intell. **17**(7), 741–748 (2004)
7. Stollberg, M.: SHAPE: service and software architectures, infrastructures and engineering. White Paper (2010)
8. Williams, B., Nayak, P.: A model-based approach to reactive self-configuring systems. In: Proceedings of AAAI'96, pp. 971–978 (1996)

9. Loo, B.T., Condie, T., Hellerstein, J.M., Maniatis, P., Roscoe, T., Stoica, I.: Implementing declarative overlays. ACM SIGOPS Operating Syst. Rev. **39**(5), 90 (2005)
10. Lim, A.: Distributed services for information dissemination in self-organizing sensor networks. J. Franklin Inst. **338**(6), 707–727 (2001)
11. Usländer, T.: Reference model for the ORCHESTRA architecture (RM-OA). Open Geospatial Consortium https://portal.opengeospatial.org/files (2005)
12. Bulusu, N., Heidemann, J., Estrin, D., Tran, T.: Self-configuring localization systems: design and experimental evaluation. ACM Trans. Embedded Comput. Syst. (TECS) **3**(1), 24–60 (2004)
13. Clare, L., Pottie, G., Agre, J.: Self-organizing distributed sensor networks. Proc. SPIE **3713**(229), 229–237 (1999)
14. Stollberg, M.: SHAPE: service and software architectures, infrastructures and engineering. White Paper (2010)
15. Presser, M., Daras, P., Baker, N., Karnouskos, S., Gluhak, A., Krco, S., Diaz, C., Verbauwhede, I., Naqvi, S., Alvarez, F., Fernandez-Cuesta, A.: Real world internet. Position paper. Future Internet Assembly (2010)
16. Roggen, D., Förster, K., Calatroni, A., Holleczek, T., Fang, Y., Tröster, G., Lukowicz, P., Pirkl, G., Bannach, D., Kunze, K.: OPPORTUNITY: towards opportunistic activity and context recognition systems. In: Proceedings of the Third IEEE WoWMoM Workshop on Autonomic and Opportunistic Communications (2009)
17. Zhang, W., Hansen, K.: Semantic web based self-management for a pervasive service middleware. In: Proceedings of the Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems, pp. 245–254. IEEE CS Press, Los Alamitos, CA (2008)
18. Spalazzese, R., Inverardi, P., Issarny, V.: A formalization of mediating connectors: towards on the fly interoperability. Technical Report, TRCS 004/2009, University of L'Aquila (2009)
19. Klopfer, M., Simonis, I.: SANY-an open service architecture for sensor networks: SANY Consortium, 2009
20. Botts, M., Percivall, G., Reed, C., Davidson, J.: OGC® sensor web enablement: overview and high level architecture. GeoSensor Networks, pp. 175–190 (2008)
21. Li, L., Taylor, K., A framework for semantic sensor network services. In": Proceedings of International Conference on Service Oriented Computing (ICSOC'08), pp. 347–361 (2008)
22. Bröring, A., Janowicz, K., Stasch, C., Kuhn, W.: Semantic challenges for sensor plug and play. Web Wireless Geogr. Inf. Syst., LNCS **5886**, 72–86 (2009)
23. Uschold, M., Clark, P., Dickey, F., Fung, C., Smith, S., Uczekaj, S., Wilke, M., Bechhofer, S., Horrocks, I.: A semantic infosphere. Proceedings of International Semantic Web Conference (ISWC'03), pp. 882–896 (2003)
24. Wang, J., Jin, B., Li, J.: An ontology-based publish/subscribe system. In: Proceedings of Middleware'04, pp. 232–253 (2004)
25. Facca, F., Komazec, S., Zaremba, M.: Towards a semantic enabled middleware for publish/subscribe applications. In: Proceedings of IEEE International Conference on Semantic Computing (ICSC'08), pp. 498–503. IEEE CS Press, Los Alamitos (2008)
26. Qian, J., Yin, J., Shi, D., Dong, J.: Exploring a semantic publish/subscribe middleware for event-based SOA. In: Proceedings of IEEE Asia-Pacific Services Computing Conference, pp. 1269–1275. IEEE CS Press, Los Alamitos (2008)
27. Petrovic, M., Burcea, I., Jacobsen, H.: S-ToPSS: semantic Toronto publish/subscribe system. In: Proceedings of the 29th VLDB Conference, pp. 1104–1107 (2003)
28. Zeng, L., Lei, H.: A semantic publish/subscribe system. In: Proceedings of IEEE International Conference on E-Commerce Technology for Dynamic E-Business (CEC-East'04), pp. 32–39. IEEE CS Press, Los Alamitos (2004)
29. Lim, A.: Distributed services for information dissemination in self-organizing sensor networks. J. Franklin Inst. **338**(6), 707–727 (2001)
30. Rajan, D., Spanias, A., Ranganath, S., Banavar, M.K., Spanias, P.: Health monitoring laboratories by interfacing physiological sensors to mobile android devices. In: Frontiers in Education Conference, 2013 IEEE, pp. 1049–1055. ISSN: 0190-5848

31. Kanoun, K., Mamaghanian, H., Khaled, N., Atienza, D.: A real-time compressed sensing based personal cardiogram monitoring system. In: Design, Automation and Test in Europe Conference and Exhibition (DATE), 2011. ISSN: 1530-1591, ISBN: 978-1-61284-208-0
32. Pelegris, P., Banitsas, K., Orbach, T., Marias, K.: A novel method to detect heart beat rate using a mobile phone. In: Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE, pp. 5488 – 5491. ISSN: 1557-170X
33. Spanias, A., Atti, V.: Interactive online undergraduate laboratories using J-DSP. IEEE Trans. Educ. **48**, 735–749. ISSN: 0018-9359
34. Mohammad, B., Elgabra, H., Ashour, R., Saleh, H.: Portable wireless biomedical temperature monitoring system: architecture and implementation. In: Innovations in Information Technology (IIT), 2013 9th International Conference. INSPEC Accession number: 13583533, pp. 95–100
35. Bande, V., Pop, S., Ciascai, I., Pitica, D.: Real-time sensor acquisition interfacing using MatLAB, pp. 189–192. ISBN: 978-1-4673-4757-0
36. Gervais, S.: Next smart sensors generation. Sensors Applications Symposium (SAS), 2011 IEEE, pp. 193-196. ISBN: 978-1-4244-8063-0
37. Kofi, A., Makinwa, A.: Smart CMOS temperature sensors. 3rd International workshop on advances in sensors and interfaces (IWASI 2009), p. 87. ISBN: 978-1-4244-4708-4
38. Blockly programming environment on https://code.google.com/p/blockly/
39. Chun, S.-J., Seoul National University of Education: Development and Application of a Web-based Programming Learning System with LED Display Kits, chunsj@snue.ac.kr
40. Ryoo, J., The Pennsylvania State University-Altoona: Development and Application of a Web-based Programming Learning System with LED Display Kits, jryoo@psu.edu
41. Zephyr HXM BT—Bluetooth heart rate monitor, Zephyr Technologies, 2012 on http://www.zephyranywhere.com/products/hxm-bluetooth-heart-rate-monitor/
42. Zhang, P., Chen, M., He, P.-J.: The study of interfacing wireless sensor networks to grid computing based on web service. Second international workshop on education technology and computer science (ETCS), 2010, vol. 1, pp. 439–442. ISBN: 978-1-4244-6388-6
43. Al-Ali, A.R., Aji, Y.R., Othman, H.F., Fakhreddin, F.T.: Wireless smart sensors networks overview. In: Second IFIP International Conference on Wireless and Optical Communications Networks (WOCN 2005), pp. 536–540. ISBN: 0-7803-9019-9
44. Ukil, A.: Towards networked smart digital sensors: a review. Industrial Electronics, 2008, (IECON 2008), 34th Annual Conference of IEEE, pp. 1798–1802. ISSN: 1553-572X
45. Barowski, L.A.: Chief Software Engineer, jGrasp. http://www.jgrasp.org/index.html
46. Cross II, J.H.: Project Director, jGrasp. http://www.jgrasp.org/index.html

Adapting User's Context to Understand Mobile Information Needs

Sondess Missaoui and Rim Faiz

Abstract The use of the user's environmental and physical context can reveal important information to enhance Mobile Information Retrieval. However the typical mobile search process integrates all gathered information about the user's context. These approaches do not take into account user's intention behind the query, which decreases their reliability and effectiveness in terms of leading to the appropriate user's information need. In this paper, we study the problem of finding a set of user's context information allow to disambiguate user's query. These contextual informations, that we call relevant dimensions, can help to personalize the mobile search process. To this aim we develop a context filtering approach CFA. The problem of finding such set of dimensions can be assimilated to a context filtering problem. We propose a novel measure that directly precises the relevance degree of each contextual dimension, which leads to finally filter the user's context by retaining only relevant. Our experiments show that our measure can analyze the real user's context of up to 6,000 of dimensions related to more than 2,000 of user's queries. We also show experimentally the quality of the set of contextual dimensions proposed, and the interest of the measure to understand mobile user's needs and to filter his context.

Keywords Information retrieval · Mobile information · Relevance analysis · Personalise web search

S. Missaoui (✉)
LARODEC, ISG University of Tunis, Le Bardo, Tunisia
e-mail: sondes.missaoui@yahoo.fr

R. Faiz
LARODEC, IHEC University of Carthage, Carthage, Tunisia
e-mail: Rim.Faiz@ihec.rnu.tn

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_29,

1 Introduction

Mobile computing becomes an interesting topic of research driven by the innovation. In fact, there is a growing market of Internet connected mobile devices and web search are the most used service. However, this explosive growth has lead to a new challenge facing mobile users. That is a need for Information Retrieval enhancement taking into account the mobile context and mobile devices characteristics. Mobile individuals using PDA, iPad, iPhone or others devices to search for information differ from users of desktop computers, they have some distinct characteristics: less patience and typically use 1–2 keywords maximum per web search. Thereby, their queries are often short and ambiguous that traditional search engines cannot guess their information demands accurately. For such reason, an interesting aspect emerging in Mobile Information Retrieval (Mobile IR) appeared recently, that is related to the several contextual dimensions that can be considered as new features to enhance the user's request and solve "the mismatch query problem" [1].

While, in the mobile information environment, the context is a strong trigger for information needs. So, the question is "What contextual dimensions reflect better the information need and lead to the appropriate search results?" In this paper, we focus our research efforts on an area that has received less attention which is the context filtering. We have brought a new approach that has addressed two main problems: how to identify the user's context dependency of mobile queries? And how to filter this user's context and select the most relevant contextual dimensions? In fact, our hypothesis is that an accurate and relevant contextual dimension is the dimension that provides an interesting improvement in retrieved results (cf. Sect. 4). Those dimensions can improve the quality of search. They help to propose to the user results tailored to his current context.

The remainder of this paper is organized as follows. In Sect. 2, we give an overview of related work which address Context-centered mobile web search. We describe in Sect. 3, our context model. Then in Sect. 4, we present our Filtering approach. Then, in Sect. 5, we discuss experiments and obtained results. Finally, by Sect. 6, we conclude this paper and outline future work.

2 Context-Centered Approaches for Mobile Information Retrieval

The work on context-aware approaches focuses on the adaptation of Mobile IR systems to users needs and tasks. These approaches modelize the user's current context, and exploit it in the retrieval process. The Related work in the domain can be summarized in terms of three categories. Firstly, approaches which are characterized as "one dimension fits all". They use one same contextual dimension to personalize all search queries. Secondly, approaches such as [2, 3] that exploit a set of predefined dimensions for all queries even these latter are submitted by

different users in different contexts. Finally, approaches that are performed to the aim of filtering the user's context and exploit only the relevant information to personalize the mobile search. In this category, our work has proceeded in terms of filtering the mobile context and identifying relevant dimensions to be latter using in contextual ranking approach.

The "one dimension fits all" approaches consider user's mobile context as one dimension at a time sessions. In this category, several research efforts are builded to modelize the current user's situation, where location is probably the most commonly used variable in context recognition. Some of these approaches such as Bouidghaghen [4], Welch and Cho [5], Chirita et al. [6], Vadrevu et al. [7] and Gravano et al. [8] have build models able to categorize queries according to their geographic intent.

With the aim of recognizing user's intention behind the search, the use of a unique predefining context's dimension is not accurate. For example, when a mobile user is a passenger at the airport and he is late for check-in, the relevant information often depends in more than time or localization. It is a complex searching task. So, it needs some additional context dimensions (e.g., flight number inferred from the user's personal calendar or numeric agenda). For such reason, the second category of approaches propose to use a set of contextual dimensions for all queries and do not offer any context adaptation models to the specific goals of the users. Several works in this category use 'Here' and 'now', both as the main important concepts for mobile search. Thus, Coppola et al. [9] and Castelli et al. [10] projects operate including Time and Location as main dimensions besides others. Most of these approaches use classification, machine learning techniques for context modeling. A few studies have tried to use semantics and ontological modeling techniques for context such as Gross and Klemke [11], Jarke et al. [12] and Aréchiga et al. [13].

While all aspects of the operational mobile environment have the potential to influence the outcome search results, only a subset is actually relevant. For such reason, the last category of approaches such as [14, 15] are proposed to identify the appropriate contextual information in order to tailor the search engine results to the specific user's context. Kessler [14] approach is built to automatically identify relevant context parameters. He proposes a cognitively plausible dissimilarity measure "DIR". Another research effort, Stefanidis et al. [15], specify context as a set of multidimensional attributes. They identify user's preferences in terms of their ability to tailor with the context of a query.

In order to improve understanding the user's needs and to satisfy them by providing relevant responses, we propose a novel model inspired by the last category of approaches. It allows to define the most relevant and influential user's context dimensions for each search situation. Comparing to the previously discussed approaches, our main contribution is to filter the mobile user's context in order to tailor search results with the intention behind his query. We formulate the context filtering problem as the problem of identifying those contextual dimensions which are eligible to encompass the user's preferences. We provide a new score that allows to compute the relevance degree of each dimension. The idea is:

"the more relevant the context dimension is, the more effective the personalized answer can be".

3 Context Model

Within Mobile IR research, context information provides an important basis for identifying and understanding user's information needs. The key notion of context may have multiple interpretations.

In this paper, we adopt the definition of [16] in which the context is: "Any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves". In our work, the context is modeled through a finite set of special purpose attributes, called context dimensions c_i, where $c_i \in C$ and C is a set of n dimensions $\{c_1, c_2, \ldots, c_n\}$.

For instance we adopt a context model that represents the user's context by only three dimensions Location, Time, Activity. In our model, the user's current context can be considered as the current state at the time of the query submission. For example, when a query such as "Restaurant" is formulated by a parent, his current situation can be definite as Location: Sousse—Tunisia; Time: Evening-12/09/2012; Activity: Outing with family. We present in the next section our filtering model including the main features that allow to filter the user's current context and specify the most relevant dimensions to narrow the search.

4 Context Filtering Process: CFA Approach

A user's context is multidimensional and contains lots of dimensions. All those dimensions are changing from one situation to another and may have an important effect for a query and haven't the same importance for another. Hence, in order to identify contextual dimension efficiency, we propose to measure their ability to enhance retrieved results. Thus, we measure their ability to enhance the query languages models (cf. Sect. 4.2). To this aim, we execute the following steps.

Step 1: We begin by selecting the top N (cf. Sect. 5) search results of initial user's query (Q_{in}).

Step 2: In the second step, we refine the query by adding one contextual dimension c_i at time. We obtain a refined query (Q_{c_i}) for which we select also the top N search results.

Step 3: We measure the effect of the dimension c_i on the query outcomes by comparing the search results of both initial query (Q_{in}) and refined query (Q_{c_i}) using two features which are the Content and the Preferences Relevance (cf. Sect. 4.1). The assumption is that the more the dimension enhance the Content and Preferences Query profiles the more it is relevant.

Step 4: We introduce a new metric measure (cf. Sect. 4.2) that combines both Content and Preferences Relevance to define a Relevance score that allows to specify the type of a dimension (Relevant/Irrelevant).

Step 5: After repeating the previous steps for all contextual dimensions $c_i \in C$, finally, we select the relevant dimensions that have the higher relevance measure. Those will be used in personalization process as the relevant current user's context.

4.1 Features Set

According to Diaz and Jones [17]: "One way to analyze a query is to look at the type of documents it retrieves". On basis of this rules, we infer that the best way to analyze a context dimension is to look at its effect on the query. Thence, its effect on the type of documents the query retrieves. In our work we use the language model approach [18] to filter the context by examining the dimension effects on the top N documents (cf. Sect. 5) of retrieved results. We offer two types of filtering features as follows.

Content Relevance Features

Content Query Profile: According to Lavrenko and Croft [19], in a language modeling context, we rank the documents in a collection according to their likelihood of having generated the query. Given a query Q and a document D, this likelihood is presented by the following equation:

$$P(Q|D) = \prod_{w \in Q} P(w|D)^{q_w} \tag{1}$$

We denote, q_w as the number of times the word w occurs in query Q which was restricted to 0 or 1. A document language model $P(w|D)$ is estimated using the words in the document. This ranking allows to build a query language model, $P(w|Q)$, out of the top N documents as follows.

$$P(w|Q) = \sum_{D \in R} P(w|D) \frac{P(Q|D)}{\sum_{D \in R} P(Q|D')} \tag{2}$$

Where R is the set of top N documents (cf. Sect. 5). This query language model, computed over all the query terms, is called "**the Content Profile of the Query** Q".

Content Relevance of Dimension c_i: The Content Relevance Feature is a Kullback-Leibler divergence, between the Content Query Profiles (unigram distributions) of (Q_{in}) and (Q_{c_i}). (Q_{in}) is the initial query submitted by the user. (Q_{c_i}) is the refined query by adding the contextual dimension c_i to (Q_{in}). The Content Relevance Feature is a gap presented as follows.

$$D_{kl}(P(w|Q_{c_i}), P(w|Q_{in})) = \sum_{w \in Q} P(w|Q_{c_i}) \log \frac{P(w|Q_{c_i})}{P(w|Q_{in})} \quad (3)$$

where $P(w|Q_{in})$, is the language model of the initial query, used as a background distribution. Thus, $P(w|Q_{c_i})$ is the language model of the refined query.

Preferences Relevance Feature

Preferences Query Profile: We are interested in describing the effectiveness of a query to return results related to user's preferences. E.g., Searching for some "events", the mobile search system must take into account the user's preference "Art". Hence relevant retrieved results must contain cultural or musical events. By analogy with the "Content Query Profile", we create a "Preferences Query Profile" described as the maximum likelihood estimate of the preference profile model.

$$\hat{P}(Pre|Q) = \sum_{D \in R} \hat{P}(Pre|D) \frac{P(Q|D)}{\sum_{D \in R} P(Q|D)} \quad (4)$$

Where "*Pre*" is a term that describes a user preferences category from a data base containing all user's preferences (his profile). For example if a user is interested by "Music" a set of terms such as (Classical songs, Opera, Piano, Saxophone) are defined as "*Pre*". The maximum likelihood estimate of the probability "*Pre*" under the term distribution for document D is:

$$\hat{P}(Pre|D) = \begin{cases} 1 & \text{if } Pre \in Pre_D \\ 0 & \text{Otherwise} \end{cases} \quad (5)$$

Where Pre_D is the set of categories names of interests contained in document D (e.g. Art, Music, News, Cinema, Horoscope …). A very helpful step is about smoothing maximum likelihood models such as $\hat{P}(Pre|Q)$. We used Jelinek-Mercer process [20] to smooth $\hat{P}(Pre|Q)$ with a background model. Such background smoothing is often helpful to handle potential irregularities in the collection distribution over interests. Also, it replaces zero probability events with a very small probability. Our aim is to assign a very small likelihood of a topic where we have no explicit evidence. This reference-model is defined by:

$$\hat{P}(Pre \backslash Q_{in}) = \frac{1}{|N|} \sum_{D} \hat{P}(Pre|D) \quad (6)$$

Our estimation can then be linearly interpolated with this reference model such that:

$$P'(Pre|Q) = \lambda \hat{P}(Pre|Q) + (1 - \lambda) \hat{P}(Pre|Q_{in}) \quad (7)$$

Given λ as a smoothing parameter.

Preferences Relevance Feature of Contextual Dimension c_i: Once the Preferences Profile is calculated for both the initial query (Q_{in}) and the refined query (Q_{c_i}), we calculate the Preferences relevancy of c_i using Kullback-Leibler divergence which leads to calculate the rate between both profiles. It is initially defined as:

$$D_{kl}(P(Pre|Q_{c_i}), P(Pre|Q_{in})) = \sum_{Pre \in Pre_D} P(Pre|Q_{c_i}) \log \frac{P(Pre|Q_{c_i})}{P(Pre|Q_{in})} \quad (8)$$

where $P(Pre|Q_{c_i})$ is preferences profile for the refined query Q_{c_i} using a contextual dimension c_i. Thus, $P(Pre|Q_{in})$ is the Preferences Profile for the initial query Q_{in}.

4.2 The New Measure of Context Filtering Process

We introduce a new measure that combines both the Content Relevance and the Preferences Relevance of the dimension. The objective of this combination is to select the most relevant dimensions. After normalizing each features, these are combined linearly using the following formula:

$$Relevance\,(c_i, Q) = [D_{kl}(P(Pre|Q_{c_i}), P(Pre|Q_{in})) + D_{kl}(P(w|Q_{c_i}), P(w|Q_{in}))] \quad (9)$$

with *Relevance* (c_i, Q) on [0, 1]. Where c_i and C represent respectively, contextual dimension and user's current context. Once this Relevance score is calculated, we define experimentally a threshold value γ. A relevant dimension c_i must have a relevance degree that goes beyond γ, otherwise it is considered irrelevant and will be not including in the personalization as an element of the accurate user's current context C. In the next section, we will evaluate the effectiveness of our metric measure "Relevance score" to classify the contextual dimensions.

5 Experimental Evaluation

5.1 Dataset

For the experiments reported in this work, we used a real-world dataset which is a portion of the 2006 query log of AOL.[1] We had relied on some experts in the field of Information Retrieval to pick manually 2000 queries based on the signification of their terms which may be related to the user's environmental and physical context. Where three contextual dimensions (Time, Location and Activity) are

[1] http://www.gregsadetsky.com/aol-data/

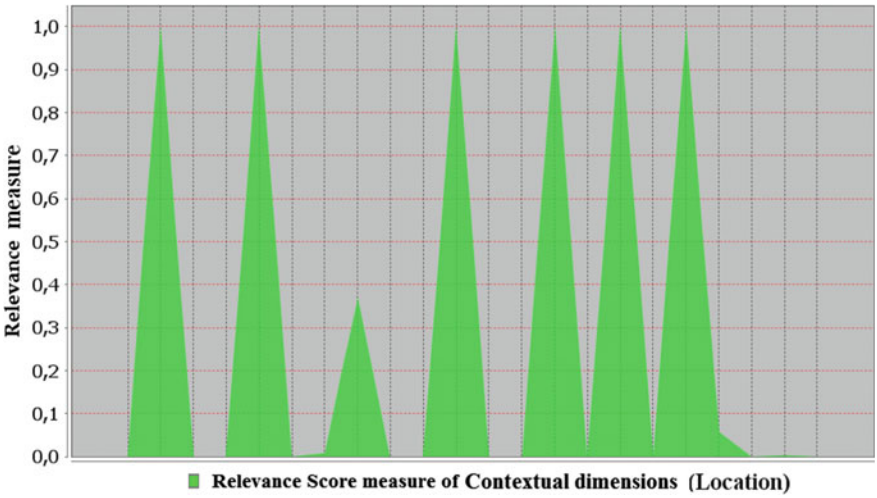


Fig. 1 Distribution of relevance measure for geographic dimension (Location)

assigned to each query to indicate the user's current situation. To classify contextual dimensions, experts assign a pertinence degree to each dimension according to their related queries. These steps left us, in our sample test queries, with 34 % irrelevant dimensions and 65.6 % relevant. To obtain the top N Web pages that match each query, we use the Google Custom Search API.[2] We considered only the first 10 retrieved results, which is reasonable for a mobile browser, because mobile users aren't likely to scroll through long lists of retrieved results. To evaluate the effectiveness of our technique to identify user's relevant contextual dimensions, we build a context intent classifier using "Relevance score" as a classification feature. In order to compute the performance of the classifiers in predicting the dimension class, we use standard precision, recall and F-measure measures. We also Compare our classifier to several supervised individual classifiers (Decision trees, Naive Bayes, SVM, and a Rule-Based Classifier) implemented as part of the Weka[3] software.

5.2 Results and Discussion

5.2.1 Analysis of Relevance Score Measure

At this level we analyze the "Relevance score" distribution for each category of contextual dimensions. Figure 1 shows distribution of this measure over different

[2] https://developers.google.com/custom-search/

[3] http://www.cs.waikato.ac.nz/ml/weka/

Table 1 Classification performance obtained using a classifier with relevance score feature

Classifier	Class	Precision	Recall	F-measure	Accuracy (%)
SVM	Relevant	0.978	0.989	0.981	99
	Irrelevant	1	1	1	
	Average	0.991	0.99	0.99	
JRIP rules	Relevant	0.911	0.953	0.924	96.3
	Irrelevant	1	1	1	
	Average	0.965	0.962	0.962	
Bayes	Relevant	1	0.933	0.966	97
	Irrelevant	1	1	1	
	Average	0.973	0.971	0.971	
J48	Relevant	1	0.933	0.966	97
	Irrelevant	1	1	1	
	Average	0.973	0.971	0.971	

values of Location dimension for different queries. In this figure we notice that there are remarkable drops and peaks in the value of "Relevance score". Indeed, the relevance of a contextual dimension is independent on his type or value but it depends on the query and the intention of mobile user behind such query. Hence, Relevance score measure hasn't a uniform distribution for those dimensions. We can conclude that the measure based on language model approach succeeds to measure the sensitivity of a mobile query to each contextual dimension.

5.3 Effectiveness of Contextual Parameter Classification

Our goal in this evaluation is to assess the effectiveness of our classification attribute "Relevance score" to identify the type of contextual dimension from classes relevant and irrelevant. As discussed above, we tested different types of classifiers and Table 1 presents the values of the evaluation metrics obtained by each one. However, all the classifiers were able to distinguish between the both classes. "SVM" classifier achieves the highest accuracy with 99 % for the F-measure. This first experiment implies the effectiveness of our approach to accurately distinguish the both types of user's current contextual information. It especially allows to correctly identify irrelevant dimension with an evaluation measure over 1. When relevant achieving over 97 % classification accuracy.

5.3.1 Comparison of CFA approach with DIR Approach

In a second experiment, we evaluated the classification effectiveness of our approach comparatively to DIR approach [14]. The DIR measure enables distinguishing between irrelevant and relevant context. We implemented the DIR

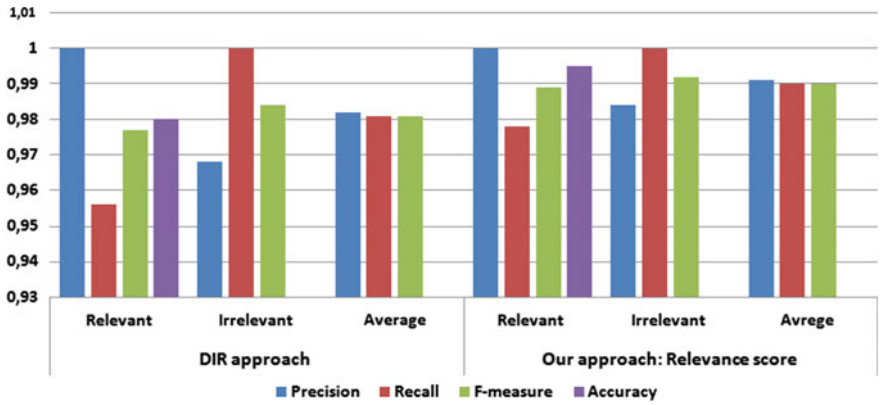


Fig. 2 Comparison between our CFA approach based relevance score and DIR measure approach

Table 2 Classification performance on relevant and irrelevant classes: comparison between CFA and DIR measure approaches

Approach	DIR approach			CFA approach					
Class	Relevant	Irrelevant	Average	Relevant	Impro (%)	Irrelevant	Impro (%)	Average	Impro (%)
Precision	1	0.968	0.982	1	0	0.984	1.7	0.991	1
Recall	0.956	1	0.981	0.978	2.3	1	0	0.99	1
F-measure	0.977	0.984	0.981	0.989	1.3	0.992	0.9	0.99	1
Accuracy (%)	98			99.5					1.5

approach using the SVM classifier that achieves one of the best classification performance using one simple rule: relevant contextual information must have an impact that goes beyond a threshold value. Then, we compare our experimental results with outcomes from DIR approach on this basis.

Figure 2 presents the result of this comparison. Looking in depth on this graph of Fig. 2, we can see the difference between the performance of both measures. We can see a clear improvement in the classification of Relevant and Irrelevant dimensions using Relevance score. We used the most commonly used evaluation measures Precision, Recall and F-measure, which prove the reliability of CFA approach to distinguish between dimensions classes. Table 2 explains in more details the comparison results.

Table 2 presents the results of comparison through the precision, recall, F-measure and accuracy achieved by the SVM classifier according to the both approaches. The result of comparison show that, our approach gives higher classification performance than DIR approach with an improvement of 1.5 % at accuracy. This improvement is mainly over relevant context dimensions with 1 % at Recall.

6 Conclusion

This paper focused on an essential challenge on Mobile Information Retrieval by developing a new approach to solve it. The challenge is finding the most interesting and relevant contextual dimensions to personalize mobile web search. In fact, we suggest CFA approach to filter the user's context and to select the most relevant dimensions. These dimensions (e.g., Time, Location, Activity, …) will improve retrieval process to produce in context results. Our approach is based on building a measure namely Relevance score. This measure allows to effectively classify contextual dimensions into relevant and irrelevant class. We have evaluated the classification performance of our metric measure comparatively to a cognitively plausible dissimilarity measure namely DIR. In future we plan to exploit our proposed CFA approach for identifying relevant contextual information as an evidence to personalize mobile Web search.

References

1. Mario, A., Cantera, J.M., Fuente, P., Llamas, C., Vegas, J.: Knowledge-based thesaurus recommender system in mobile web search (2010)
2. Varma, V., Sriharsha, N., Pingali, P.: Personalized web search engine for mobile devices. In: International Workshop on Intelligent Information Access (2006)
3. Yau, S., Liu, H., Huang, D., Yao, Y.: Situation-aware personalized information retrieval for mobile internet. In: The 27th Annual International Computer Software and Applications Conference (2003)
4. Bouidghaghen, O.: Accés contextuel à l'information dans un environnement mobile : approche basée sur l'utilisation d'un profil situationnel de l'utilisateur et d'un profil de localisation des requêtes. Thesis of Paul Sabatier University (2011)
5. Welch, M., Cho, J.: Automatically identifying localizable queries. In: Proceedings of 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 1185–1186 (2008)
6. Chirita, P., Firan, C., Nejdl, W.: Summarizing local context to personalize global Web search. In: Proceedings of the Annual International Conference on Information and Knowledge Management, pp. 287–296 (2006)
7. Vadrevu, S., Zhang, Y., Tseng, B., Sun, G., Li, X.: Identifying regional sensitive queries in web search. In: WWW '08 Proceedings of the 17th international conference on World Wide Web, pp. 1185–1186 (2008)
8. Gravano, L., Hatzivassiloglou, V., Lichtenstein, R.: Categorizing web queries according to geographical locality. In: Proceedings of the twelfth international conference on Information and knowledge management, pp. 325–333 (2003)
9. Coppola, P., Della Mea, V., Di Gaspero, L., Menegon, D., Mischis, D., Mizzaro, S., Scagnetto, I., Vassena, L.: CAB: the context-aware browser. IEEE Intell. Syst. **25**(1), 38–47 (2010)
10. Castelli, G., Mamei, M., Rosi, A.: The Whereabouts Diary, pp 175–192. Springer, Berlin (2007)
11. Gross, T., Klemke, R.: Context modelling for information retrieval: requirements and approaches. J. WWW/Internet **1**, 29–42 (2003)
12. Jarke, M., Klemke, R., Nicki, A.: An Environment for Multi-Dimensional User-Adaptive Knowledge Management. IEEE Computer Society Press (2001)

13. Aréchiga, D., Vegas, J., Redondo, P.F.: Mymose: ontology supported personalized search for mobile devices. In: Proceedings of ONTOSE (2009)
14. Kessler, C.: What is the difference? A cognitive dissimilarity measure for information retrieval result sets. Knowl. Inf. Syst. **30**(2), 319–340 (2012)
15. Stefanidis, K., Pitoura, E., Vassiliadis, P.: Adding context to preferences. In: Proceedings of the 23rd International Conference on Data Engineering (ICDE), p. 23 (2007)
16. Dey, A.K., Abowd, G.D.: Towards a better understanding of context and context-awareness. CHI 2000 Workshop on the What, Who, Where, When, Why and How of Context-Awareness (2000)
17. Diaz, F., Jones, R.: Using temporal profiles of queries for precision prediction. SIGIR'04 ACM J. **4** (2004)
18. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 275–281 (1998). Knowledge Information Systems J, 1–34 (2010)
19. Lavrenko, V., Croft, W.B.: Relevance-based language models. In: Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 120–127 (2001)
20. Jelinek, F., Mercer, R.: Interpolated estimation of Markov source parameters from sparse data. In: Proceedings of the Workshop on Pattern Recognition in Practice, Amsterdam (1980)

Web Service Based Data Collection Technique for Education System

Ruchika Thukral and Anita Goel

Abstract This paper presents a web service data collection technique that facilitates observing data collection process for education system. Our technique uses web service for collection of education data. Education Management Information System (EMIS) as a centralized data collection system consumes web service of regional centers for collecting data from application to application. Our data collection technique is divided in two phases. First phase collects data from education providers at regional level centers and second phase collects data for EMIS from all regional levels centers using web service. First phase is divided in four components to register contact addresses of education providers, send data collection format (DCF) and collect filled DCF from web addresses. Thus data is stored at regional levels which can be later collected by EMIS when required. Regional level centers are required to give access of data using web service. Web service can be easily consumed by EMIS using WSDL information to collect data which does not need any modifications in information systems at both regional and central level. Data can be entered directly by education providers to be stored at regional centers, as against centralized system for data entry. So our technique enables platform-independent, computer to computer collection of data. We illustrate our technique by a case study for data collection from schools.

Keywords Data collection format (DCF) · Dissemination · Data consumer · Data provider · Education Management Information System · Web service · XML

R. Thukral (✉)
Department of Computer Science, University of Delhi, New Delhi, India
e-mail: ruchikathukral2203@gmail.com

A. Goel
Department of Computer Science, Dyal Singh College, University of Delhi, New Delhi, India
e-mail: agoel@dsc.du.ac.in

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_30,

1 Introduction

The features of web service to provide interoperability between different computing machines introduces new paradigm for data collection in education system. It facilitates data exchange over the World Wide Web using XML and requires no alterations in hardware and software of computing machines. Web service is platform independent and can access data stored in XML format from any platform. The process of online data collection in education system introduces new kinds of issues in the collection process. Some issues involved in data collection process are different from old paper-pen data collection process. In order to handle issues in online data collection process, there is a need of improvisation or new technique needs to be developed.

During data collection process, data is collected by education system from education institutes. Education system uses Education Management Information System (EMIS) to send data collection formant (DCF) to all educational institutes which is collected back at regional centers. Regional centers are supposed to enter data in centralized data entry system one by one for every institute. This involved few issues which are listed as: data entry is done only at regional centers manually, regional centers need continuous internet connection during the data entry process, voluminous data entry increases the chances of errors, EMIS has to collect data from all over the country at a time which increases the processing time, different geographical regions demands different policies and planning according to their weather conditions at different time but current system lays policies for entire country irrespective of requirements, regional centers have no access of data to do further analysis for improvisation of educational conditions in their respective regions, different education institutes have different information management system thus application to application data collection requires lots of modification.

In this paper we focus on online data collection and we use web service for our data collection technique. In our technique we have used two user groups to address main actors in data collection process:—(1) Data consumer, and (2) Data provider. Data consumer collects data from the educational institutes, like, different departments in education ministry and data providers are the educational institutes from where data is collected, like, schools, colleges, universities, institutes etc. Data is collected from the data providers in a uniform format i.e. Data Collection Format (DCF). DCF is designed according to the requirements of data consumers and disseminated to data providers using online methods like emails, download from website, or online forms. In our technique we have divided data collection in two phases:

- *collection of data from data providers at regional level centers*
- *collection of data from regional to centralized data collection system of EMIS.*

In first phase of data collection, we have four components. Components are designed to first register online contact addresses of data providers and DCF. Then

DCF is sent and collected back using registered contact addresses at regional centers. In our technique, the data collection cycle includes

- *Accepting DCF and list of data provider's email-ids and web address (URL) from data consumers,*
- *Sending DCF to data providers via email,*
- *Collecting filled DCF from specified data providers web address and*
- *Storing collected data.*

In second phase data collected at regional level centers is collected by centralized system using web service interface. EMIS through web service interface of regional centers can collect data region wise for further analysis when required. Use of web service has enabled application to application data collection i.e. from regional level centers to central EMIS.

Our technique using web service in data collection process has many advantages over current data collection process used by education system. Data is collected from application to application at regional level centers instead of centralized data collection. Data is saved at regional centers without human intervention in comparison of data entry of each and every data provider separately in current system. Chances of errors have been reduced due to data entry at origin. The data is collected directly from data provider, thereby doing away with CD's or hand filled printed forms, thus supporting green computing. Data collection with our technique speeds up the process, resulting in timely data collection. The proposed technique enables platform independent, computer to computer collection of data from heterogeneous environments. There is no requirement of any alterations in hardware or software of computing systems at any level of data collection. It also has an advantage over direct data entry in centralized system where regional centers could not use data directly for further planning. Our technique provides data access on both levels for regional as well as central planning. Thus web service interface for data collection process has speed up the system with fewer chances of errors which required no modification in information management system at any level of administration.

The paper is divided in 6 sections. Section 2 discusses data collection in education system. Section 3 presents the architecture of web service based data collection technique for Education system. Section 4 describes a case study of using proposed data collection techniques to collect data from schools of India. Finally, Sects. 5 and 6 state related work and conclusion, respectively.

2 Data Collection in Education System

Data collection requires collecting uniform data from data providers by education system. Online data collection system followed by education system is time consuming and cumbersome. The data collected from data providers is required to

be relevant and uniform to provide information to education planners and decision makers. Unified format i.e. Data Capture Format (DCF) is used to collect relevant data from the data providers. DCF is disseminated region wise by downloading from web site and then distributed personally or via post mail. DCF is disseminated to data providers using dissemination channel which follows hierarchy of administration levels. Same hierarchy is followed to collect DCF from data providers. DCF collected is required to be verified by special team at regional level centers. After verification, all DCF are entered one by one in the centralized data collection system of EMIS.

DCF used for data collection from institutes only collects statistical data like number of infrastructural objects, number of classes, number of teachers, teacher student ratio per class etc. Special trainings sessions are organized for better understanding of long DCF and for filling the DCF with correct information by the head of institutes. Despite special trainings, inconsistencies in data and incomplete entries are affecting data entry process in centralized system. Here, our technique for data collection in education system is used to disseminate and collect DCF from data providers directly. DCF is designed using XML based schema which is easy to understand and is user friendly to collect both statistical as well as detailed data.

Data entry in centralized system is a tedious and time consuming process. Voluminous data entry increases the chances of errors. Even though significant number of institutes has access to computer and Internet, data cannot be entered from source because of software incompatibility, different operating system and hardware, lengthy DCF and lack of skilled personnel for direct data entry.

As we are collecting data at regional centers, data is collected by centralized system by using web service interface. Web service interface provides interoperability, leverages existing web standards for data exchange. Web service uses XML as the standard language for communication between applications as its interoperability feature supports messaging system which is regardless of syntax and contents. In Web service, WSDL (Web Service Description Language) uses XML based documentation to describe the service it provides. With the help of WSDL information of regional data transfer web services, EMIS can consume web service for all regional level centers to collect data from application to application.

3 Web Service Based Data Collection Technique

Data collection technique focuses on online collecting of data from data providers. In the proposed technique, we are using web service to collect data from data providers which first collects data at regional level. We have targeted two major groups:

- *Data consumer*: who collects data from educational institutes, and
- *Data provider*: who provides data to data consumer.

As data collection involves different levels of administration, so we are introducing two levels for data collection. Levels of administration used in our technique are:

- *Regional level*
- *Central Level*.

Regional level of administration is state or district that collects data from data providers of respective states and central level is education ministry that collects data from all regional levels when required using regional level web service. Data transfer web services are consumed by EMIS using WSDL information. To cater both the levels of data collection, our technique works in two phases that are as follows:

- *Collection of data from data providers at regional level centers*: here collection process includes four components from registering of contact addresses to collection of DCF.
- *Collection of data from regional level centers to centralized data collection system of EMIS*: here collection process includes collecting of data from regional level centers to central level directly from application to application using web service interface.

First phase of proposed technique to collect data from data providers at a regional center consists of four components:

- *Data-Request*: This component initializes the system with list of data providers' email-ids, URLs (web address) and DCF.
- *DCF-Dissemination*: This component disseminates DCF to data providers in their email-ids.
- *DCF-DataFill*: This component provides guidelines for data providers to fill up DCF and uploads the filled DCF on the data provider's registered web address.
- *DCF-Collection*: This component collects filled DCF as XML document from data provider's web address and stores data in database.

As shown in Fig. 1, data collection technique has four components in first phase. Data Request initializes the list of email-ids and URLs (web addresses) of data providers along with the DCF. DCF-Dissemination component disseminates DCF to all registered email-ids of data providers. In DCF-DataFill component, data provider is required to fill in data and convert it into XML document. Filled DCF is required to be uploaded on the registered web addresses (URL) by data providers. DCF-Collecting component used to collect filled DCF which is uploaded on the data provider's registered web addresses. Filled DCF is collected as XML document and stored in the database.

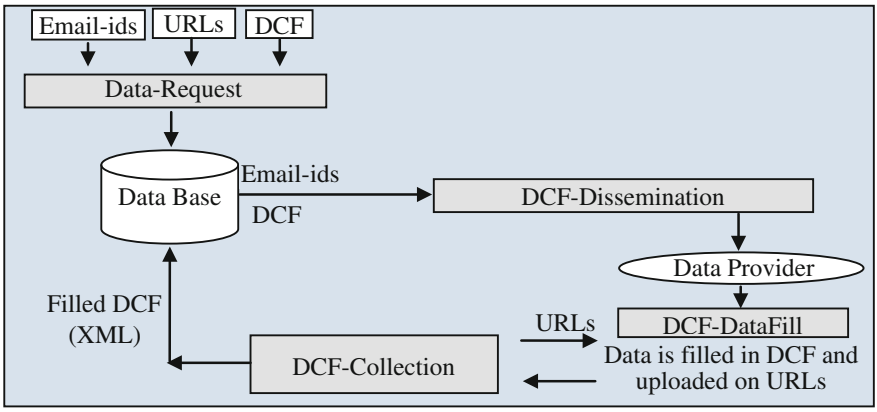


Fig. 1 Data collection technique in phase one

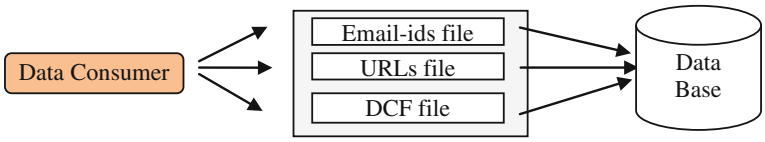


Fig. 2 Data-request component

3.1 Data Request Component

This component is designed to register three files by data consumer. First file contains list of email-ids of data providers, second file contains web addresses from where filled DCF will be collected and third file is Data Collection Format (DCF) as XML document. In database, separate tables have been created to store data from files uploaded files by data consumer as shown in Fig. 2.

3.2 DCF-Dissemination Component

This component disseminates DCF to all the registered email-ids. List of email-ids is selected from the data base to send DCF one by one.

Figure 3 illustrates that data consumer using DCF-Dissemination component disseminates DCF to data providers. DCF and Email-ids will be selected from the data base to send XML schema of DCF. Each and every email-id is taken one by one to send DCF.

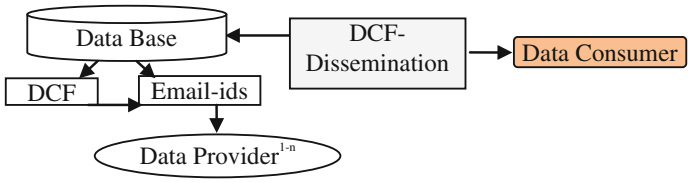


Fig. 3 DCF-dissemination component

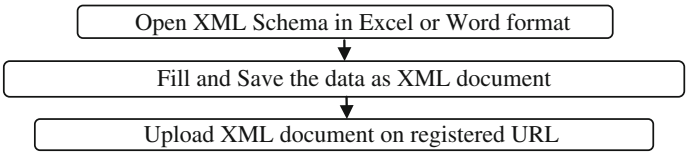


Fig. 4 DCF-datafill component

3.3 DCF-DataFill Component

This component describes the data provider's end where they are supposed to fill the DCF and upload on web address (registered URL).

In this component Data providers get DCF as XML schema in their email-ids which they are required to open in Excel sheet or Word document. DCF in Excl sheet or Word document can be filled by copying data from data provider's information system or manually as shown in Fig. 4. After filling up the data, data provider is supposed to export data to XML document and save as filledDCF.xml file. FilledDCF.xml document is required to be uploaded on same web address (URL) which was registered by data consumer in first component.

3.4 DCF-Collection Component

This component is designed for collecting DCF directly from the data provider's web addresses (URL's) i.e. application to application or machine to machine. For the functioning of this component, it is required to upload filledDCF.XML file by data providers on their web addresses. As shown in Fig. 5 data consumer is required to use DCF-Collection component to collect filled DCF from registered URLs taken from database one by one. If filledDCF.xml is not available on web address, it will be searched again for data during data collection period.

In second phase of data collection, data collection of first phase is required to be completed. Collected data is stored in information management system of every regional centre. Here we use web service interface to collect data from regional level centers to central level.

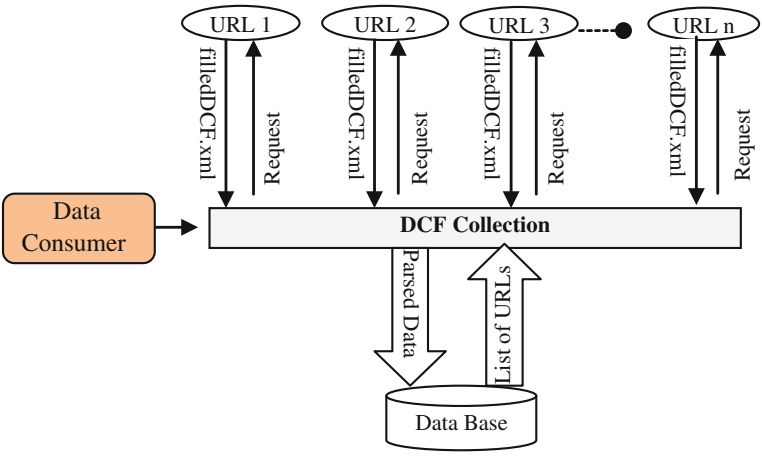


Fig. 5 Data collection component

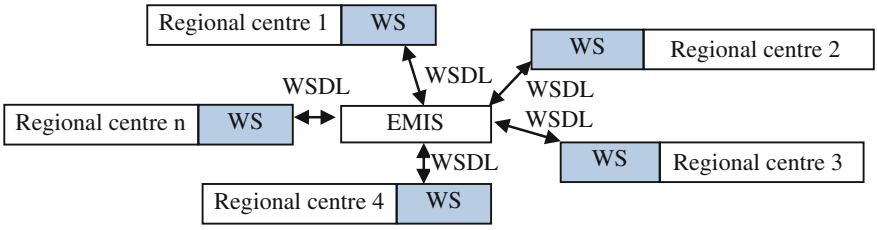


Fig. 6 Data collection in phase second

Web service interface for data collection:—This phase requires regional centres to provide data as web service. Web services used by all regional centers use query to transfer data. Regional level center is required to register their web service (WS) with EMIS. EMIS will consume web service (WS) which is defined by WSDL to collect data directly from application to application i.e. from regional level centers to central level EMIS as shown in Fig. 6.

4 Case Study: Use of Web Service in Data Collection for EMIS

In India, schools form a vast network with 13,00,000 primary and 2,00,000 senior secondary schools spread across more than 600 districts in 28 states and 7 Union territories [1]. Most schools in urban areas are not fully equipped with latest computing facilities. Schools in rural areas are completely deprived of computing system. To collect uniform data from all the schools, uniform format (DCF) is send

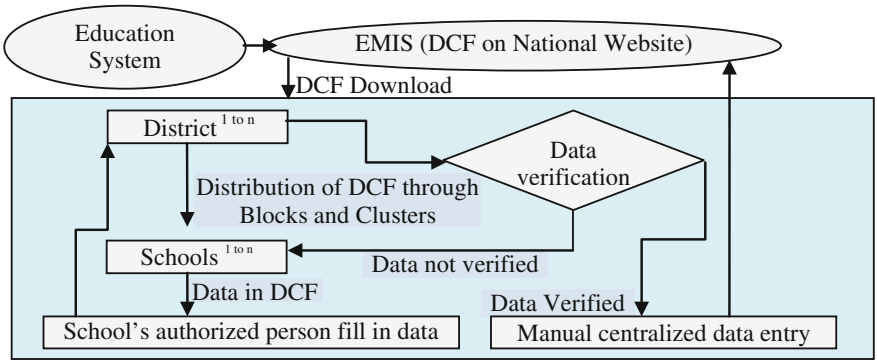


Fig. 7 Working of existing EMIS

to all the schools of India as hard copy of DCF in printed form. DCF used by education system collects only statistical information about schools. Present DCF is approximately 20–25 pages long and is disseminated to schools by a proper hierarchy i.e. from national administration to districts, Blocks, Clusters and schools. Data is collected back via the same hierarchy in reverse as shown in Fig. 7.

After collection of data, it is verified by district officers. If they find any unfilled columns or inconsistency, the DCF is sent back otherwise the data is entered in an online form. But factors like internet connection, power supply and lack of skills lead to delayed data entry. Also, the District has to bear with extra cost for hardware and software setup to fill the online DCF in centralized data entry system. Providing special training to the heads of the schools to fill data in DCF and appointing special skilled staff for data entry adds to the data collection budget.

Our data collection technique has been implemented in schools of two states i.e. New Delhi and Haryana. Earlier the schools received the data collection form for statistical data from the Education department of Government of India. Schools send their data to the district either in CDs or as printed form. At district level, there is a direct online data entry to the Education Management Information System, where the data gets freeze after it has been filled online. Thereafter making any changes to the data involves a series of steps. Using first phase of our technique, schools received the DCF as XML schema in their email-ids and they are required to fill it in Excel format. After filling it is needed to be saved as XML document for direct communication between the different information systems of institutes or schools. In this way, we can collect DCF directly from schools without setting up special computing infrastructure. The proposed technique interacts with two groups:

- *Education department at two levels* i.e. district and central. Center collects data from district using web service interface
- *Schools* who perform data entry in DCF and upload it.

Education department at district level interacted with Data-Request component to upload address of data providers (schools), e-contacts and DCF. DCF-Dissemination component has disseminated DCF to all school in their email-ids. DCF-Collection component is used to collect data directly from data providers web addresses to store data at every district.

Deliverables to education department are:

- *A component to upload email-ids, URLs and DCF as text files.*
- *A component to disseminate DCF as email to schools.*
- *A component to collect data from schools in DCF, over the internet.*

Deliverables to schools

- *DCF for entering data, regardless of hardware, software and operating system used for school's information system.*
- *Easy to fill data entry format*.

Centralized data entry of existing data collection system requires same computing infrastructure at all the schools and special training sessions for entering data at different levels (district, state etc.) which increases the cost of data collection. The proposed technique uses web service technology in second phase of data collection which allows data collection form computer to computer i.e. from all districts to central data collection system (EMIS). Use of web service to collect data from all districts does away with the modifications required for data exchange from application to application. Major benefits are time saving and low or no expenditure on infrastructure and trainings.

Benefits of proposed technique over current system are:

- Collecting data from school spread across different geographical locations
- Computer to computer data collection is possible irrespective of heterogeneous computing environments thus give away the use of printed forms, hand filled forms, also formal trainings for data entry thus support—Green computing.
- Data collection format is easy to understand and fill, thus school's authorized persons have shown interest in providing the data with their own responsibility.
- Data is collected at source has reduced the chances of errors.
- Data collection is faster facilitating decision makers in forming policies on educational front of a country in time.
- Voluminous detailed data is collected with the help of easy to fill DCF.
- Major cost savings are done as no special trainings or infrastructure setup is required to collect data from all districts.
- Districts can use collected data for their planning to improve education system.
- Education ministry can collect data district wise for further analysis instead of collecting voluminous data from entire country at a time using centralized data entry system (Table 1).

Table 1 Comparison of different data collection techniques

Benefits of using data collection technique	Use of paper-pen	Use of online methods	Use of web service
Infrastructural cost	×	✔	×
Internet connection	×	✔	✔
Ease of data collection	×	×	✔
Accessibility of data	×	×	✔
Personnel training	✔	✔	×
Geographical segregation	×	×	✔
Collecting data in details	×	×	✔
Environment friendly	×	×	✔
Data collection region wise	×	×	✔
Hardware and software updates	×	✔	×

5 Related Work

Different methods of data collection have been used over the years like face to face interview, questionnaires, online surveys, EDI etc. for combination of online and offline data collection methods [2, 3]. Online data collection is being done at District level where hardware and software components have been setup for DCF submission. Thus all schools are required to submit DCF either in printed form or soft copy in CDs or flash drives. District officer hire special team for verification and online submission of DCF which is really a cumbersome task.

Dissemination of DCF [4–6] to education data providers is done to collect uniform data. Education data consumers upload DCF on their website which can be downloaded by data providers but due to lack of proper computing infrastructure, DCF is downloaded by District officers and distributed to schools through a channel in printed form or in CDs for offline entering the data in DCF. In our technique we propose a web service based dissemination system where the institutes can use the DCF irrespective of hardware or software infrastructure directly from the website.

Web Service [4, 7–9], characteristics such as interoperability, leveraging web standards and protocols for data transfer, has made it widely used over the internet. Web service is also being used by Australian government for the collection of student's feedback [10]. Data collection using Web service is possible in XML document as Web service exchange methods in same [11]. The proposed technique uses web service based data collection method to collect data from educational institute. Use of web service allowed collecting the data from heterogeneous information systems of educational institutes without any alteration in the information system [12, 13]. XML submission with the help of Excel [14] is used by European Food Safety and Authority for data collection through Web service. In our technique, we are proposing Web service for data collection.

EMIS [4–6, 14–19], is a system to collect data from educational institutes to provide information to the policy makers. Decision making process is dependent

on relevant and reliable data collected in time. The DCF which has been used for data collection only captures statistical information like number of rooms, number of water resources, number of fans, number of teachers, number of students etc. in a school. It does not capture students or teachers name, age, subject etc. Thus in our paper, we are proposing portable and easy to understand DCF which would collect both statistical and detailed information.

6 Conclusion

This paper presents web service based online data collection technique for education system. Education Management Information System collects data from data providers, like, educational institutes, and process data for providing information to data consumers, like, researchers, planners and decision makers. Online methods for data collection have limitations like hardware and software incompatibility, information systems of institutes are on heterogeneous platforms etc. The proposed web service based data collection technique is used for data collection over the web, irrespective of heterogeneous information systems. The data is collected directly from the regional level centers via web service without altering their hardware or software components. Since data is collected directly from its origin by regional centers using first phase of our data collection technique, chances of errors in the collected data are highly reduced. The data is collected electronically, thereby doing away with CDs, hand-filled or printed forms for data collection, thus supporting green computing. The electronic data collection speeds up the process, resulting in timely collection of data.

References

1. Goel, R.A.: Using web service interface for data collection. Int. J. Comput. Sci. Issues (IJCSI-2012-9-2-1757) (2012)
2. Granello, D.H., Wheaton, J.E.: Online data collection: strategies for research. JCD Alexandria **82**(4), 387–394 (2004)
3. McDonald, H., Adam, S.: A comparison of online and postal data collection methods in marketing research. Market. Intell. Plann. **21**(2), 85–95 (2003)
4. Almonaies, A.A., Alalfi, M.H., Cordy, J.R., Dean, T.R.: Towards a framework for migrating web applications to web services. In: Proceedings of the 2011 Conference of the Center for Advanced Studies on Collaborative Research (2011)
5. Connal, C., Sauvageot, C.: NFE-MIS handbook: developing a Sub-National Non-Formal Education Management System. Paris: UNESCO. http://unesdoc.unesco.org/images/0014/001457/145791e.pdf (2005)
6. Wako, T.N.: Education Management Information Systems (EMIS) a Guide for Young Managers. NESIS/UNESCO, vol. 57 (2003)
7. Arsanjani, A., Hailpern, B., Martin, J., Tarr, P.L.: IBM research report web services: promises and compromises, RC22494 (W0206-107), 20 June 2002

8. Nezhad, H.R.M., Benatallah, B.: Web services interoperability specifications, 0018-9162/06, IEEE (2006)
9. Al-Jaroodi, J., Mohamed, N., Aziz, J.: Service oriented middleware: trends and challenges. In: 7th International Conference on Information Technology, IEEE, 978-0-7695-3984-3/10 (2010)
10. Australian Government: Higher education in formation management system (heims) heims web services interface technical specification chessn functions, Commonwealth of Australia (2008)
11. Hansen, M., Madnick, S., Siegel, M.: Data integration using web services. Working paper 4406-02 CISL 2002-14 May 2002
12. Shoab, M., Jain, K., Shashi, M.: Development and implementation of web service for logging and retrieving real time train location information. (IJSCE) ISSN: 2231-2307, **2**(6), January 2013
13. Andrade, R.B., Nunes, L., Barbosa, B.M.D.M., Vijaykumar, N.L., Santos, R.D.C.: A web service-based framework for temporal/spatial environmental data access. In: 12th International Conference on Computational Science and Its Applications, 2012 IEEE
14. European Food Safety Authority: Technical report on the use of Excel/XML files for submission of data to the Zoonoses system. European Food Safety Authority: EN-233. [33 pp.], http://www.efsa.europa.eu/en/ publications (2012)
15. Mehta, A.C.: DISE II: Higher Education Management Information System (HE-MIS), ACM/HE-MIS/Sept 14, 2007
16. Mohamed, A., Kadir, N.A.N.A., May-Lin, Y., Rahman, S.A., Arshad, N.H.: Data completeness analysis in the Malaysian Educational Management Information System. IJEDICT **5**(2), 106–122 (2009)
17. Hua, H., Herstein, J.: Education Management Information System (EMIS): integrated data and information systems and their implications in educational management. In: Annual Conference of Comparative and International Education Society March 2003
18. Powell, M.: Rethinking Education Management Information Systems: lessons from and options for less developed countries. Working paper no. 6 (2006)
19. Cassidy, T.: Education Management Information System (EMIS) Development in Latin America and the Caribbean: Lessons and Challenges, Jan 2005

Approximate Dynamic Programming for Traffic Signal Control at Isolated Intersection

Biao Yin, Mahjoub Dridi and Abdellah El Moudni

Abstract As a new optimization technique for discrete dynamic systems, approximate dynamic programming (ADP) for the optimization control of a simple traffic signalized intersection is proposed. ADP combines the concepts of reinforcement learning and dynamic programming, and it is an effective and practical approach for real-time traffic signal control. This paper aims at minimizing the average number of vehicles waiting in the queue or the vehicles average waiting time at isolated intersection by using the action-dependent ADP (ADHDP). ADHDP signal controller is designed with neural networks to learn and achieve a near optimal traffic control policy by measuring the traffic states. As shown by the comparison with other traffic control methods, the simulation results indicate that the approach is efficient to improve traffic control at a simple intersection.

Keywords Approximate dynamic control (ADP) · Dynamic programming · Neural networks · Traffic signal control policy

1 Introduction

Nowadays urban traffic congestion becomes more and more serious, which means excess delay, safety and pollution problems. In traffic signal control system, some research is based on the isolated intersection. In order to reduce traffic delay or

B. Yin (✉) · M. Dridi · A. El Moudni
Université de Technologie de Belfort-Montbéliard, Belfort, France
e-mail: biao.yin@utbm.fr; yihu580124@gmail.com

M. Dridi
e-mail: mahjoub.dridi@utbm.fr

A. El Moudni
e-mail: abdellah.el-moudni@utbm.fr

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_31,

queue lengths of vehicles waiting at the approaches, it is necessary to design an optimal control policy to make vehicles pass through the intersection efficiently.

Normally, there are three traffic control methods at isolated intersection, pre-timed control, actuated control and adaptive control. Pre-timed control also called fixed-time control is the traditional approach for traffic signal control. It is the most basic type of control logic implemented. The cycle length and the phase splits are set by fixed values as well as the duration of each interval within each phase. It is easy to achieve the control in placid traffic flow rather than the complex and changeable traffic flow condition. Actuated control is another type control using the demand-responsive logic to set signal timings based on traffic demand as registered by detectors or other traffic sensors on the intersection approaches. Cycle length, phase split and even phase sequence can vary in response to current traffic demand. The most common feature of actuated control is the ability to extend the length of the green interval for a particular phase. This approach only considers the traffic flow in current phase, without taking the flow in other phases into account. Like actuated control, adaptive control responds to traffic demand in real time, realizing the adjustment of states parameters such as traffic volume, stop times, delay and queue length. What's more, it can also change or adjust the allocation of the cycle time to the various phases in different intersections to make them cooperative. Adaptive traffic control systems are becoming more widespread.

In this paper, on account of vehicle actuated control or traffic responsive control at isolated intersection, we develop an adaptive control strategy based on approximate dynamic programming (ADP) to provide efficiency in operation of traffic control field. The traffic responsive control problem can be expressed as a multi-stage optimization process. Robertson and Bretherton [1], Gartner [2] use dynamic programming (DP) approach to solve this problem. The results show that using DP can reduce about 56 % of vehicle delays from the best fixed-time plans. Nevertheless, with a large dimension, the DP's implication for real-time traffic signal control is limited. As much research focuses on the ADP theory and its applications in traffic control problems [3–5], it shows that ADP supports an effective way to solve the dimension problem of complex dynamic systems.

The formulation of ADP is first proposed by Werbos [6]. And then, he further proposed two basic ADP versions which are heuristic dynamic programming (HDP) and dual heuristic programming (DHP) [7]. In recent years, ADP has been developed by many researchers, such as Si et al. [8] and Powell [9]. The main idea of ADP is to use a structure of approximation function, such as neural network, fuzzy model, polynomial and so on, to estimate the cost-to-go value function in dynamic programming. So, it can effectively avoid the "curse of dimensionality" caused by large state space in the recursive calculation of Bellman's equation.

In this paper, we use the "action-dependent critics", namely ADHDP as the structure of our traffic control model to verify the practicability of the ADP theory in traffic control. The artificial neural networks are adopted as a function approximation structure to approximate the cost-to-go function.

The paper is organized as following. Section 2 describes the intersection configuration and traffic signal control model. Section 3 discusses the traffic signal

controller designed based on ADHDP. Two three-layer neural networks served as action network and critic network are presented. Section 4 tests the control approach in simulation with different traffic flow rates and in comparison with the presented control methods. Some conclusions are given in Sect. 5.

2 Problem Description and Traffic Control Model

2.1 Configuration of Two Phases Intersection

In the signalized intersection, there are a conflict zone and four approaches with lanes of arriving and departure. The traffic flows can be partitioned into disjointed combinations of non-conflicting flows that will have the right-of-way to occupy the conflict zone. We analyze a simple traffic intersection of 4-lane combined with two conflicting movements (Fig. 1).

In this simple intersection, the incompatible phases are either on green or red. To avoid interference between antagonistic streams, the intergreen time also called a red clearance interval is necessarily considered when two conflicting movements alternately have right-of-way to access the conflict zone. As the traffic lights indications are formulated in discrete time, each phase has minimum green time, maximum green time and extended green time. Denote the time increment by Δd when extension green time is required.

As we know, the actuated control method only considers the traffic flow in current phase, without taking the flows in other phases into account. Our work is trying to adjust the parameters to optimize the traffic control policy by using intelligent algorithms. At first, the traffic control modeling will be introduced.

2.2 Traffic Control Modeling

2.2.1 System State Variables

In traffic control system at time t, $k_i(t)$ denotes the traffic state which means the actual number of vehicles queuing on lane i ($i = 1,\ldots,I$). I is the total number of lanes. $x_i(t)$ denotes the signal state on lane i and it is a binary variable depending on the traffic signal indication such that:

$$x_i(t) = \begin{cases} 1 & \text{if signal is green for lane } i \\ 0 & \text{if signal is red for lane } i \end{cases} \tag{1}$$

In order to reduce the number of state variables collection ($k_i(t)$, $x_i(t)$), the system state $s(t)$ which will be used as the input variables in the algorithm in

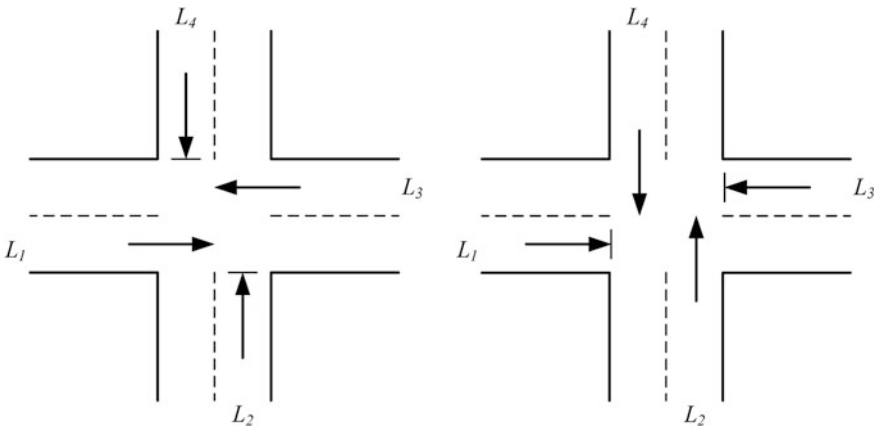


Fig. 1 A two-phase signalized intersection

Sect. 3, is defined as the collection of two elements $s_1(t)$, $s_2(t)$. $s_1(t)$ denotes the sum of maximums of all queue lengths measured in the same combinations, and $s_2(t)$ denotes the green time of the current phase which is equal to the least minimum green $g_{\min}$ with the addition of extension interval green time. The system state is expressed as

$$s(t) = (s_1(t), s_2(t)) \tag{2}$$

$$s_1(t) = \max(k_1(t), k_3(t)) + \max(k_2(t), k_4(t)) \tag{3}$$

$$s_2(t) = g_{\min} + \sum_{x_i(t)=1} x_i(t) \cdot \Delta d, \ (i = 1, 3 \text{ or } 2, 4) \tag{4}$$

During $(t, t + \triangle d)$, denote the arrival traffic on lane i by $w_i(t)$ which can be obtained by the traffic arrival pattern prediction. Denote the departing traffic rate by $y_i(t)$ which can be obtained by the three conditions:

$$y_i(t) = \begin{cases} 0 & \text{if on red or intergreen interval} \\ \Delta d \cdot S & \text{if on green and } k_i(t) + w_i(t) \geq \Delta d \cdot S \\ k_i(t) + w_i(t) & \text{if on green and } k_i(t) + w_i(t) < \Delta d \cdot S \end{cases} \tag{5}$$

where S is the saturation flow rate (veh/s) of a single traffic lane. Assume that the rates of all lanes are the same.

2.2.2 System Decision

At the start of each time t, the decision of the system is to switch the green phase to the next phase or unchanged, namely extending the current green phase. Let $u_i(t)$ denote the system decision during $(t, t + \triangle d)$ on lane i.

$$u_i(t) = \begin{cases} 1 & \text{for signal changed} \\ 0 & \text{unchanged} \end{cases} \tag{6}$$

2.2.3 Transition of System State

Once the system has made a decision on signal status, the state of the intersection will be changed. The transition of the signal state $x_i(t)$ and traffic state $k_i(t)$ are transferred by (7) and (8), respectively.

$$x_i(t + \Delta d) = (x_i(t) + u_i(t)) \bmod 2 \tag{7}$$

$$k_i(t + \Delta d) = k_i(t) + w_i(t) - y_i(t) \tag{8}$$

Actually, the capacity of each lane is limited. The maximum queue length that each lane could hold is defined as K_{limit}. So, the queue length in each lane should be constricted by

$$0 \le k_i(t) \le K_{\text{limit}} \tag{9}$$

2.2.4 Average Waiting Time

In the model, the objective is to minimize the vehicle average waiting time during planning horizon T by the optimized control policy. Let $\triangle t$ denote the phase time and it is equivalent to the sum of intergreen time t_{int} and the green time including minimum green time $g_{\min}$ and extended green time. Additionally, the maximum green time is $g_{\max}$. As we can see,

$$\Delta t = t_{\text{int}} + g_{\min} + n \cdot \Delta d \tag{10}$$

where n is the total increments of extension green time before the signal is changed. Further, in planning horizon T, $n(m)$ denotes the total increments of extension green time during the phase time $\triangle t(m)$ at phase alternation m. Thus, we have the constraint as following (Fig. 2):

$$\sum_{m=1}^{M} \Delta t(m) = M(t_{\text{int}} + g_{\min}) + \sum_{m=1}^{M} n(m)\Delta d \le T \tag{11}$$

In fact, vehicles total waiting time in all lanes can be divided into three parts *TI*, *TG* and *TR* to obtain by (12) respectively. *TI* denotes vehicles total waiting time during intergreen time $(t,\ t + t_{\text{int}})$ in all lanes of all red signals. *TG* denotes

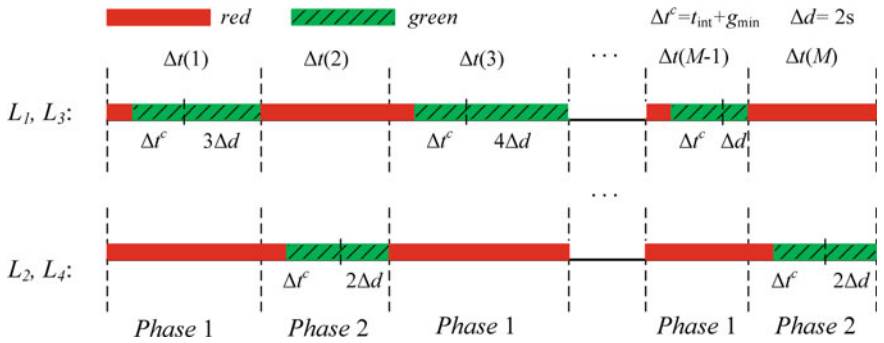


Fig. 2 Example of two-phase traffic signal diagram

vehicles total waiting time during $(t + t_{\text{int}}, t + \triangle t)$ in all lanes of green phase. *TR* denotes vehicles total waiting time during $(t + t_{\text{int}}, t + \triangle t)$ in all lanes of red phase.

$$\begin{cases} TI = \sum_{i=1}^{I} \sum_{\tau=1}^{\tau=t_{\text{int}}} k_i(t+\tau) \\ TG = \sum_{i=1}^{I} \sum_{g=t_{\text{int}}+1}^{\Delta t} x_i(t+g) \times k_i(t+g) \\ TR = \sum_{i=1}^{I} \sum_{e=t_{\text{int}}+1}^{\Delta t} (1 - x_i(t+e)) \times k_i(t+e) \end{cases} \tag{12}$$

So, the average waiting time during $(t, t + \triangle t)$ is defined as:

$$T_w = (TG + TR + TI) / \left(\sum_{i=1}^{I} \Delta t_i \right) \tag{13}$$

where $\Delta t_i \leq \Delta t$, and it equals to the during time from the start point of the phase until the time that no vehicle is waiting.

3 Approximate Dynamic Programming

In principle, ADP system should be able to approximate the solution to any problem in control or planning which can be formulated as an optimization problem [10]. So, we seek an approximation of the true value function in dynamic programming for solving traffic signals control problem.

3.1 Notation

k	is time step ($k = 0,1,\ldots,K$). In the model, the duration of one step is $\triangle d$ or $\Delta t^c = t_{\text{int}} + g_{\min}$ according to action u. $t(k)$ denotes the time when step k is occurred.
$f(\cdot)$	is a function of states transfer according to the current state and action.
T	is a time horizon, namely the total time steps.
γ	is a discount of utility function, $\gamma = 0.9$ is used in this paper.
$U(\cdot)$	is a utility or cost function.
$J(\cdot)$	is the true value performance index function of dynamic programming.
$J^*(\cdot)$	is the optimal value of $J(\cdot)$.
$\hat{J}(\cdot)$	is an approximate function of $J(\cdot)$.
$K^f_{\max}(\cdot)$	is the maximum queue length in phase f.
$E_a(\cdot)$	is objective training function in action network.
$E_c(\cdot)$	is objective training function in critic network.
$W_a(\cdot)$	is the weight vector in action network.
$W_c(\cdot)$	is the weight vector in critic network.
$l_a(\cdot)$	is the learning rate in action network.
$l_c(\cdot)$	is the learning rate in critic network.

3.2 Dynamic Programming

Suppose that traffic control system is a discrete-time nonlinear (time-varying) dynamical system. State transition equation is expressed as:

$$s(k+1) = f[s(k),\, u(k),\, k] \tag{14}$$

The objective of traffic signal control is to minimize the overall average waiting time per vehicle. According to the Little's law, this is equivalent to minimizing the average number of vehicles. So, In the case of two-phase intersection, the sum of maximums of two queue lengths measured in every phase, in this paper, is chosen as the utility function.

$$K^1_{\max}(k) = \max\{k_1(k),\, k_3(k)\} \tag{15}$$

$$K^2_{\max}(k) = \max\{k_2(k),\, k_4(k)\} \tag{16}$$

$$U(k) = K^1_{\max}(k+1) + K^2_{\max}(k+1) \tag{17}$$

Then we have objective to minimize the total discounted length of vehicles within a time horizon T with total K steps, which can be express as:

$$\min\left\{\sum_{k=0}^{K}\gamma^{k}U[s(k),\,u(k),\,k]\right\} \tag{18}$$

We can write it in the way of the system performance index (or cost):

$$J(s(j)) = \sum_{k=j}^{K}\gamma^{k-j}U[s(k),\,u(k),\,k] \tag{19}$$

The objective of dynamic programming problem is to choose a control sequence $u(k)$, $k = j, j+1,\ldots$ ($j \in [0, K]$) so that the function in (19) is minimized. According to Bellman, the optimal cost from step k is equal to

$$J^{*}(k) = \min_{u(k)}\{U(k) + \gamma J^{*}(k+1)\} \tag{20}$$

The optimal control $u^{*}(k)$ at step k is the $u(k)$ which achieves this minimum, i.e.,

$$u^{*}(k) = \arg\ \min_{u(k)}\{U(k) + \gamma J^{*}(k+1)\} \tag{21}$$

3.3 Formulation of Approximation Dynamic Programming

3.3.1 The Overview of ADP Controller

ADP generally consists of three modules: model module, action module, and critic module. Traditionally, model module and action module are the system model and controller, respectively. And critic module is used to guide the optimization of the parameters of action module quantitatively. That is, by changing the parameters of action module, maximum or minimum of the output of critic module can make the optimal or near optimal control signal from action module. Additionally, if the output of action module acts as the input of critic module, it is another form of ADP, namely "action-dependent" ADP (ADHDP). In this paper, the ADHDP is adapted to learn a near optimal control policy in the traffic control system.

Normally, a neural network is used to estimate the performance index value function defined in dynamic programming as neural network can approximate the nonlinear system. ADHDP control can adopt two neural networks to complete the action module and the critic module. Structures of these two networks are shown in Fig. 3.

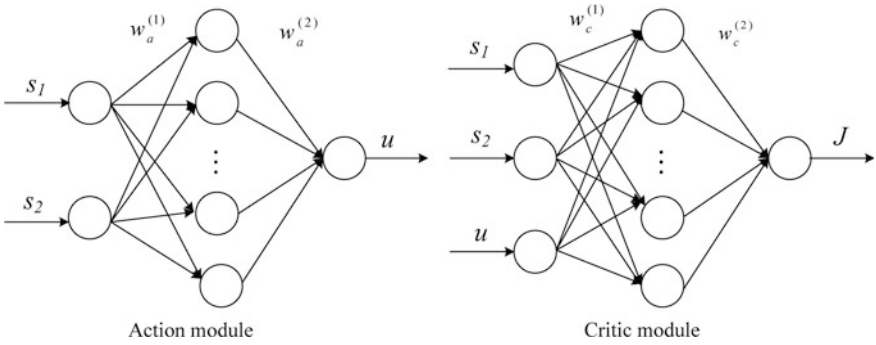


Fig. 3 Structures of action network and critic network

3.3.2 Action Network

The structure of the action network is chosen as a three layer feed-forward neural network with two inputs, a single hidden layer with four neurons, and a single output neuron. In the input layer, it has two inputs of system states, which in this paper represent the sum of maximum of two queue lengths measured in each combination, and the green time in the current phase. In the hidden layer, the neural transfer function is hyperbolic tangent sigmoid transfer function. The output of the action network is defined as $u(k)$. The control output is defined as

$$u^*(k) = \begin{cases} 1 & u(k) > 0 \\ 0 & u(k) \leq 0 \end{cases} \tag{22}$$

It is the same definition as (6), when $u^*(k) = 1$, the signal is changed, otherwise, the current green phase will be extended.

The adaption of action network is done by minimizing the following error measure over time. $U_c(k)$ is chosen as the minimum of $J(k-1)$ saved in the last control cycle.

$$E_a(k) = \frac{1}{2}[\hat{J}(k) - U_c(k)]^2 \tag{23}$$

The weight update rule for the action network is a gradient-based adaptation given by

$$W_a(k+1) = W_a(k) + \Delta W_a(k) \tag{24}$$

$$\Delta W_a(k) = l_a(k)\left[-\frac{\partial E_a(k)}{\partial W_a(k)}\right] \tag{25}$$

3.3.3 Critic Network

The structure of critic network is chosen as a three layer feed-forward neural network with three inputs, a single hidden layer with four neurons, and a single output neuron. In the input layer, it has the same inputs with the action network and the third input is just the output of the action network. In the hidden layer, the transfer function is the same as the action network. The adaption of critic network is done by minimizing the following error measure over time.

$$E_c(k) = \frac{1}{2}[\hat{J}(k) - U(k) - \gamma\hat{J}(k+1)]^2 \tag{26}$$

The weight update rule for the critic network is a gradient-based adaptation given by

$$W_c(k+1) = W_c(k) + \Delta W_c(k) \tag{27}$$

$$\Delta W_c(k) = l_c(k)\left[-\frac{\partial E_c(k)}{\partial W_c(k)}\right] \tag{28}$$

3.4 Algorithm

The traffic signal control algorithm using ADP can be summarized as the following.

- Step0: Initialization.
  - Choose an initial system state s_0; Set step index $k = 0$;
  - Initialize the parameters for the action network and the critic network;
  - Give t_{int} and $g_{\min}$ to the original phase.
- Step1: Receive new information of the maximum number of vehicles in every phase and the green duration of the current phase.
- Step2: Control decisions u and criteria J are obtained according to action network and critic network.
- Step3: Update the parameters of critic network and action network according to the input variables $s(k)$ and utility function $U(k)$.
- Step4: Implement optimal option decision $u^*(k)$ for the time $t(k)$ of the planning horizon.
  - If $u^*(k) = 1$, that means changing the current green phase, otherwise, the current green phase being extended with increment $\triangle d$;

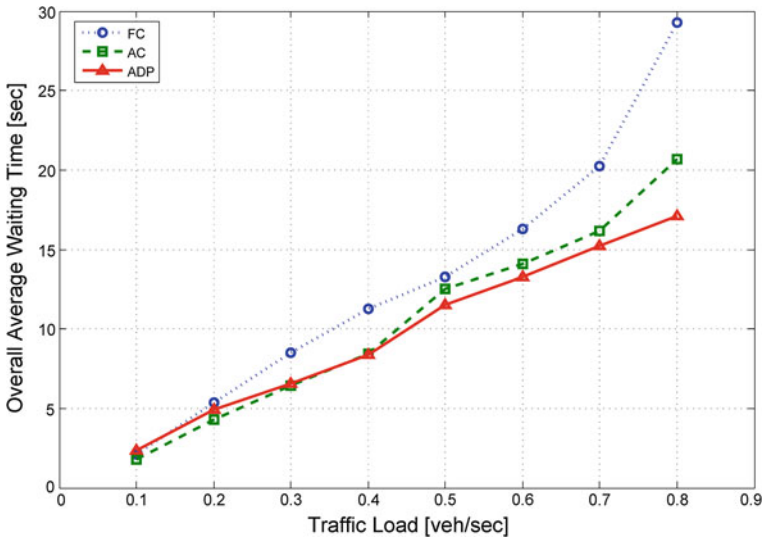


Fig. 4 Simulation results of average waiting time

– Complete the system state transition from $s(k)$ to $s(k+1)$. And set step index $k = k + 1$.

- Step5: if time consumption $t(k) < T$, then goes back to Step1; otherwise stop.

4 Simulation

In this section, a case of two-phase isolated intersection is simulated. The simulation parameters are summarized as following. The extension green interval $\triangle d$ is 2 s, minimum green time is 5 s and maximum green time is 20 s for arrival rates ranging from 0.05 to 0.2 veh/s per lane; minimum green time is 10 s and maximum green time is 50 s for arrival rates ranging from 0.25 to 0.4 veh/s per lane. These parameters are set according to the optimal green time in fix-timed cycle control with different arrival rates. The intergreen time is 3 s. The limit number of vehicles in each lane is 20. The departure rate is 1 veh/s. The traffic load ρ is defined by the sum of maximum rates of every phase. For symmetric arrival rates of two-phase system, ρ is ranging from 0.1 to 0.8 veh/s.

For comparison, the fix-timed cycle control (FC) and actuated control (AC) with the overall average delay and queue length are simulated. The simulation runs traffic flows in 2,000 s. Results are shown as follows. In Fig. 5, we can see that, the ADP method perfects well, especially during the high traffic load 0.5–0.8 veh/s. During the low traffic load 0.1–0.4 veh/s, the results of ADP method are not as good as AC method. But both of ADP and AC control methods are better than FC method (Fig. 4).

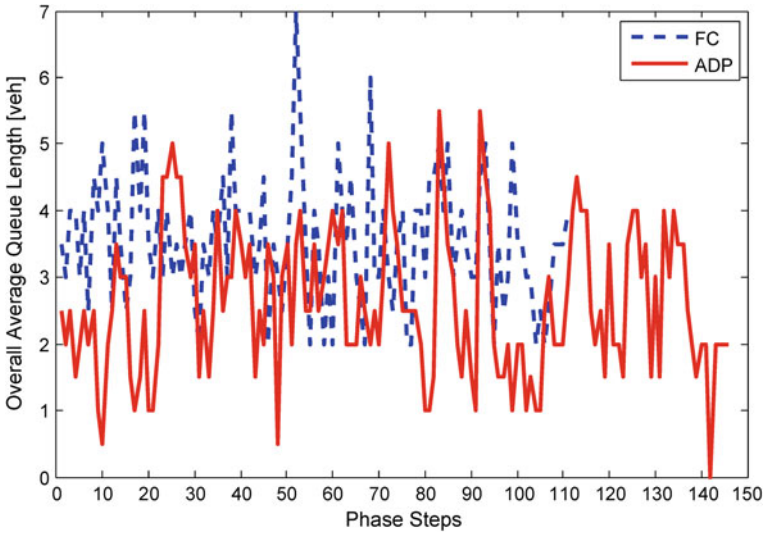


Fig. 5 Simulation results of average queue length of ADP and FC ($\rho = 0.6$)

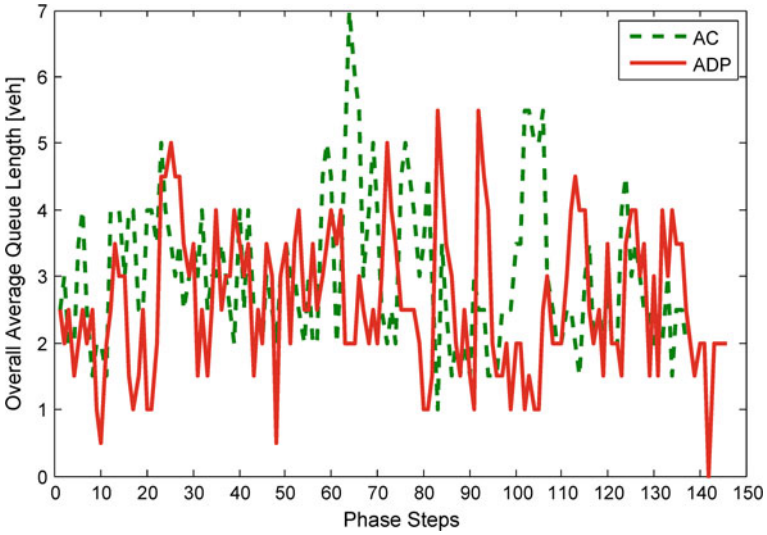


Fig. 6 Simulation results of average queue length of ADP and AC ($\rho = 0.6$)

In Figs. 5 and 6, the overall average queue length, namely the average maximum number of vehicles of every phase when the phase is changed, of ADP-FC and ADP-AC are compared. In this aspect, obviously, ADP method outperforms very well among them, e.g., $\rho = 0.6$ veh/s.

5 Conclusion

We have presented the application of approximate dynamic programming (ADP) to the field of traffic signal control. In ADP controller, two artificial neural networks are applied as the approximation structures. The overall average of vehicle waiting time was adapted as the evaluation criterion. Simulation results for different traffic flow rates are quite good and outperform existing strategies, such as fixed-time control and actuated control.

Further, a comprehensive sensitivity analysis needs to be studied, including the learning rate of convergence and the stability in stochastic dynamic systems. As for the application of traffic signal control, formulation of the ADP controller can further be used to complicated stochastic dynamic systems, such as the complicated intersections with more restrictions and states, the traffic network control, etc.

References

1. Robertson, D.I., Bretherton, R.D.: Optimum control of an intersection for any known sequence of vehicle arrivals. In: Proceedings of the 2nd IFAC/IFIP/IFORS Symposium on Traffic Control and Transportation Systems (1974)
2. Gartner, N.H.: OPAC: a demand-responsive strategy for traffic signal control. Transp. Res. Rec. (906), 75–81 (1983)
3. Cai, C., Wong, C.K., Heydecker, B.G.: Adaptive traffic signal control using approximate dynamic programming. Transp. Res. Part C **17**(5), 456–474 (2009)
4. Li, T., Zhao, D.B., Yi, J.Q.: Application of ADP to intersection signal control. Advances in Neural Networks 2007. LNCS, vol. 4491, pp. 374–379. Springer, Heidelberg (2007)
5. Li, T., Zhao, D.B., Yi, J.Q.: Adaptive dynamic neuro-fuzzy system for traffic signal control. In: IEEE International Joint Conference on Neural Networks, 2008, IEEE World Congress on Computational Intelligence, pp. 1840–1846 (2008)
6. Werbos, P.J.: Advanced forecasting methods for global crisis warning and models of intelligence. General Systems Yearbook, **22**, 25–38 (1977)
7. Werbos, P.J.: Approximate dynamic programming for real-time control and neural modeling. In: Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches, vol. 15, pp. 493–525 (1992)
8. Si, J., Barto, A.G., Powell, W.B., Wunsch, D.C.: Handbook of Learning and Approximate Dynamic Programming. IEEE Press Series on Computational Intelligence. Wiley-IEEE Press (2004)
9. Powell, W.B. Approximate Dynamic Programming: Solving the Curses of Dimensionality, vol. 703. Wiley, New York (2007)
10. Werbos, P.J., Pang X.: Generalized maze navigation: SRN critics solve what feed-forward or Hebbian nets cannot. In: IEEE International Conference on Systems, Man, and Cybernetics, 1996, vol. 3, pp. 1764–1769 (1996)

An Approach to Semantic Text Similarity Computing

Imen Akermi and Rim Faiz

Abstract The use of text similarity plays an important role in many applications in Computational Linguistics, such as Text Classification and Information Extraction and Retrieval. Besides, there are several tasks that require computing the similarity between two short segments of text. In this work, we propose a sentence similarity computing approach that takes account of the semantic and the syntactic information contained in the sentences. The proposed method can be applied in a variety of applications to mention, text knowledge representation and discovery. Experiments on a set of sentence pairs show that our approach presents a similarity measure that illustrates a considerable correlation to human judgment.

Keywords Natural language processing · Semantic similarity · Computational linguistics

1 Introduction

Natural Language Processing forms an integral part of Computational Intelligence. Indeed, with the rapid development of the computer's computational technologies, the need to rely on linguistic techniques to facilitate human–machine communication has become essential. Language processing took benefit of the power of computers to acquire a new dimension and to open the way to interesting areas of research to mention the semantic similarity calculation. Indeed, Text semantic

I. Akermi (✉)
University of Tunis—ISG, LARODEC 2000, Bardo, Tunisia
e-mail: imenakermi@yahoo.fr

R. Faiz
University of Carthage—IHEC, LARODEC 2016, Carthage, Tunisia
e-mail: rim.faiz@ihec.rnu.tn

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_32,

similarity measures have been the central concern of taxonomists of the previous century [1–3]. The increasing complexity of data requires the development of measures able to keep a semantic relevance for Information Processing related applications, such as text summarization [4], machine translation [5] and image retrieval from the Web [6]. In fact, it has been shown that short text enveloping the images can help to reach a higher retrieval precision instead of using the whole document containing the images [6]. Furthermore, text similarity is beneficial for relevance feedback, text categorization [7, 8] and evaluation of text coherence [9]. In this same context, we propose an approach that uses Web content to measure semantic similarity between a pair of short text segments. The rest of the document is organized as follows: Sect. 2 introduces the text similarity related work. In Sect. 3, we present our approach for measuring semantic similarity between sentences and we evaluate our approach in order to demonstrate its ability. In Sect. 4, we conclude with few notes and some perspectives.

2 Related Work

There are two categories of similarity calculation between sentences: statistical and semantic methods. Statistical similarity between sentences, as defined by Zhang [10], takes only into account the words in the two sentences without any former knowledge such as syntactical parsing or lexicon dictionary. They also noticed that the cost of computing statistic similarity is lower than the cost of computing semantic similarity [10].

2.1 Statistic Similarity Between Sentences

Zhang [10] present five measures of statistical similarity between sentences:

- Word set based sentence similarity: using the two sets of words of the two sentences.
- Sentence similarity based on vector: using the vectors representing the two sentences. There are two ways for assigning weights of words: the first one appoints the weight of words averagely; the second uses term frequency-inverse document frequency (TF-IDF) approach to assign the words weights.
- Sentence similarity based on edits distance: measured by the edit distances between two sentences.
- Word order based sentence similarity: employs the word pairs' orders in the sentences.
- Word distance based sentence similarity: considers the distances between word pairs in the same sentences.

The first three sentence similarity metrics are considered as symbolic similarity, while the latter ones are structural similarity. The symbolic similarity between sentences takes only into account the spelling of words disregarding the meanings of words. The structural similarity includes word orders, word distances and the structure of the sentence. For the following sections we denote:

S1: a sentence with length L1 (L1≥2).
$S1 = w_{11}w_{12}w_{13}\ldots w_{1L1}$
w_{1i} (i ∈[1,L1]) are the words or separators in S1.

S2: a sentence with length L2 (L2 ≥ 2).
$S2 = w_{21}w_{22}w_{23}\ldots w_{2L2}$
w_{2i} (i ∈[1,L2]) are the words or separators in S2.

w(S1) : the set of words enclosing all the words w1i (i ∈[1,L1]).
w(S2) : the set of words enclosing all the words w2i (i ∈[1,L2]).

Word Set based similarity. In order to measure the word set based sentence similarity, one should construct first the word sets of sentences. Bearing in mind that the sentences might embrace different voices and tenses, there exist two methods to calculate word based sentence similarity. The first one consists in calculating sentence similarity with all the words in sentences; the second one only deals with stemmed words in sentences. However, the stemming can skip the sentence tense and voice information [10].

The Jaccard similarity coefficient, as defined by Achananuparp et al. [11]: "*is a similarity measure that compares the similarity between two feature sets*". For the sentence similarity task, it is calculated as the size of the intersection of the words contained in the two sentences divided by the size of their union.

After formulating the word sets of two sentences, the Jaccard coefficient can be calculated by:

$$\text{Jaccard}(S1, S2) = \frac{|w(S1) \cap w(S2)|}{|w(S1) \cup w(S2)|} \tag{1}$$

Dice similarity is another similarity metric based on the word set and is calculated by:

$$\text{Dice}(S1, S2) = \frac{2|w(S1) \cap w(S2)|}{|w(S1)| + |w(S2)|} \tag{2}$$

Edit distance based similarity. The edit distance uses the spelling of words in two sentences. There are several kinds of edit distance: Hamming distance, Levenshtein distance, Damerau-Levenshtein distance, etc.

In the following, we give the definition of the Levenshtein distance.

(*Levenshtein Edit Distance*). "*The edit-distance of two strings is the minimal cost of a sequence of symbol insertions, deletions, or substitutions transforming one string into the other*" [12].

The sentence similarity based on the edit distance is calculated by:

$$\text{Edit}_{\text{sim}} = \frac{1}{1 + \text{Edit\_distance}} \tag{3}$$

Edit distance based similarity is widely used in measuring similarity of sequences such as strings, languages and biological sequences. However, it only involves the substitutions, deletion and insertion of characters and separators; which makes difficult to capture the meaning of words [10].

Word order based similarity. This measure is based on the orders between word pairs which are determined according to the positions of words in a sentence. The sequential relations between words formulate a sequential network of words.

The distances between words vary from 1 to |sentence| −1.

$$\begin{cases} L(S1) = \{(w11, w12); (w11, w13); \ldots; (w1(L1-1), w1L1)\} \\ L(S2) = \{(w21, w22); (w21, w23); \ldots; (w2(L2-1), w2L2)\} \end{cases}$$

We can, then, calculate the similarity between S1 and S2 based on the orders of words by:

$$\text{Set}_{\text{sim(S1,S2)}} = \frac{|\text{L(S1)} \cap \text{L(S2)}|}{|\text{L(S1)} \cup \text{L(S2)}|} \tag{4}$$

2.2 Semantic Similarity Between Sentences

Li et al. [13] developed a method that extracts text similarity from semantic and syntactic information contained in the compared sentences. Employing the words contained in the pairs of sentences, they dynamically form a joint word set. For each sentence, they derive a raw semantic vector with the help of the WordNet lexical database [14]. Li et al. [13] noticed that, the weight of a word is appropriately identified by using information content extracted from a corpus given that each word in a sentence has its own contribution to the meaning of the whole sentence. Then, a semantic vector is determined for each of the two sentences by associating the information content derived from the corpus with the raw semantic vector, and consequently, the computation of the semantic similarity is based on the two semantic vectors. Finally, the overall sentence similarity is calculated by

combining semantic similarity and the order similarity computed using the two order vectors [13].

Mihalcea et al. [15] introduced a combined method for measuring the semantic similarity of sentences by taking advantage of the information that can be deduced from the similarity of the component words. They apply two corpus based measures, Pointwise Mutual Information-Information Retrieval (PMI-IR) [16] and Latent Semantic Analysis (LSA) [17] and six knowledge-based measures [11, 18–22] of word semantic similarity, and combine the results to demonstrate the way these measures can be used to determine text similarity. They used a paraphrase recognition task to evaluate their method. According to Islam and Inkpen [23], the major issue behind this method is that it employs eight different methods to compute the similarity of words, which is not computationally efficient. Besides, Islam and Inkpen [15] noticed that the measures presented in [13] and [15] ignore the string similarity, which can be significant in some cases. Islam and Inkpen [24] proposed a method that determines the similarity of two sentences from semantic and syntactic information that they contain. They relied on three similarity functions to define a more generalized text similarity approach. As a first step, they calculate string similarity and semantic word similarity and then they apply a common-word order similarity function to include syntactic information in their method. Finally, they derive the text similarity, combining semantic similarity, string similarity and common-word order similarity, with normalization. They call their proposed method the Semantic Text Similarity (STS) method. Inkpen [25] also presented another method for computing the similarity of two short texts, based on the similarities of their words. She used the Second-Order Co-occurrence PMI (SOC-PMI) corpus-based similarity for two words which is a similarity measure that uses second order co-occurrences [26]. The method selects a word from the first text and a word from the second text, which have the highest similarity. The similarity value is stored, and the two words are taken out. The method continues until there are no more words. At the end, the similarity scores are added and normalized.

The approach we propose is different from those already mentioned in that we tried to combine several techniques taking into account the semantic and the syntactic information that the sentences may contain. The different components of our approach will be detailed in the following section.

3 A New Approach for Measuring Semantic Similarity Between Sentences

We propose a method which combines semantic and syntactic information that a sentence might contain in order to measure similarity between two sentences.

3.1 Proposed Method

Our method consists in 3 phases:

- Phase 1: Calculating the semantic similarity between the two sentences.
- Phase 2: Calculating the syntactic similarity between the two sentences.
- Phase 3: Combine the semantic and the syntactic information.

Phase 1: The semantic similarity between the two sentences.
In this phase, we start by eliminating the function words such as the, a, where, etc., and the punctuation from the two sentences, obtaining thus two sets of the terms expressing respectively the semantics of the two sentences:

$$Set_{S1} = w_1, w_2, \ldots, w_{ls1}; \quad ls1 : \textit{the number of terms of } Set_{S1}$$
$$Set_{S2} = w_1, w_2, \ldots, w_{ls2}; \quad ls2 : \textit{the number of terms of } Set_{S2}$$

Then, we:

- Select a word w_i from Set_{S1} and a word w_j from Set_{S2} having the highest similarity, which includes the computation of the similarity scores between all the pairs (w_i, w_j) using our word similarity measure Sim_{FA} presented in previous works [27]. The Sim_{FA} uses, on one hand, an online English dictionary provided by the Semantic Atlas project (SA)[1] and on the other hand, page counts returned by a social website whose content is generated by users.
- Store the similarity value of the 2 words and take the 2 words out of the sets Set_{S1} and Set_{S2}.

We continue to do so until there are no more words left in the two sets. At the end, we add the similarity scores and we normalize:

$$\text{SemSim}(S1, S2) = \frac{\sum \text{StoredScores}}{\text{Minimum}(ls1, ls2)} \tag{5}$$

Phase 2: The syntactic similarity between the two sentences.
In this phase, we form two sets out of the two sentences including the function words:

$$Set_{S1} = w_1, w_2, \ldots, w_{ls1}; \quad ls1 : \textit{the number of terms of } Set_{S1}$$
$$Set_{S2} = w_1, w_2, \ldots, w_{ls2}; \quad ls2 : \textit{the number of terms of } Set_{S2}$$

[1] http://dico.isc.cnrs.fr: belongs to the French National Center for Scientific Research's domain (CNRS), one of the major research bodies in France.

Then, we employ the Jaccard coefficient to calculate the intersection of the two words sets compared to their union:

$$\text{Jaccard}(S1, S2) = \frac{m_c}{ls1 + ls2 - m_c} \tag{6}$$

where

m_c The number of common words between the two sets.
ls1 The number of words in the set Set_{S1}.
ls2 The number of words in the set Set_{S2}.

In addition, we calculate the word order similarity measure between the two sentences. This measure is based on the orders between word pairs. For every sentence, we construct its corresponding word order set. As shown by Achananuparp et al. [11], similarity bases on word order can help to differentiate the meaning of two sentences. This is considered as crucial in many text similarity metrics since without the syntactic information, it is impossible to set apart the sentences sharing the same representation of the corresponding bag-of-word [11].

Let us take for example a sentence S = "*Jack is dancing*":

$$\text{Word}_{\text{order}}(S) = \{(\text{Jack}, \text{is}); (\text{Jack}, \text{dancing}); (\text{is}, \text{dancing})\}$$

Once we construct the word order sets $Word_{order}(S1)$ and $Word_{order}(S2)$ for the two sentences, we calculate the following score:

$$\text{Sim}_{\text{wo}}(S1, S2) = \frac{|\text{Word}_{\text{order}}(S1) \cap \text{Word}_{\text{order}}(S2)|}{|\text{Word}_{\text{order}}(S1) \cup \text{Word}_{\text{order}}(S2)|} \tag{7}$$

At the end, we add the Jaccard coefficient and the word order similarity previously calculated in order to obtain the overall syntactic similarity measure:

$$\text{SynSim}(S1, S2) = \text{Jaccard}(S1, S2) + \text{Sim}_{\text{wo}}(S1, S2) \tag{8}$$

Phase 3: The overall sentence similarity measure.
In this last phase, we incorporate both measures previously calculated by the following formula:

$$\text{SenSim}_{\text{FA}}(S1, S2) = \alpha \times \text{SemSim}(S1, S2) + (1 - \alpha) \times \text{SynSim}(S1, S2) \tag{9}$$

$\alpha \in [0,1]$

First experiments on Li et al. dataset [13] have shown that our measure performs better with $\alpha = 0, 7$.

Table 1 Results on Li et al. sentence data set

RG no.	R-G word pair in the sentence	Human sim. (mean)	Li et al. sim. method	Our method
1	Cord-smile	0.01	0.33	0.13
5	Autograph-shore	0.01	0.29	0.24
9	Asylum-fruit	0.01	0.21	0.02
13	Boy-rooster	0.11	0.53	0.16
17	Coast-forest	0.13	0.36	0.18
21	Boy-sage	0.04	0.51	0.07
25	Forest-graveyard	0.07	0.55	0.23
29	Bird-woodland	0.01	0.33	0.07
33	Hill-woodland	0.15	0.59	0.39
37	Magician-oracle	0.13	0.44	0.12
41	Oracle-sage	0.28	0.43	0.06
47	Furnace-stove	0.35	0.72	0.17
48	Magician-wizard	0.36	0.65	0.33
49	Hill-mound	0.29	0.74	0.15
50	Cord-string	0.47	0.68	0.35
51	Glass-tumbler	0.14	0.65	0.21
52	Grin-smile	0.49	0.49	0.30
53	Serf-slave	0.48	0.39	0.27
54	Journey-voyage	0.36	0.52	0.29
55	Autographsignature	0.41	0.55	0.14
56	Coast-shore	0.59	0.76	0.57
57	Forest-woodland	0.63	0.7	0.37
58	Implement-tool	0.59	0.75	0.62
59	Cock-rooster	0.86	1	0.87
60	Boy-lad	0.58	0.66	0.48
61	Cushion-pillow	0.52	0.66	0.20
62	Cemetery-graveyard	0.77	0.73	0.53
63	Automobile-car	0.56	0.64	0.45
64	Midday-noon	0.96	1	0.94
65	Gem-jewel	0.65	0.83	0.74

3.2 Evaluation

For evaluation, we used a data set of 30 sentence pairs which similarity values were computed by human judges [13]. Li et al. [13] employed the Rubenstein and Goodenough 65 noun pairs [28] and redefined them with their definitions from the Collins Cobuild dictionary [29]. These definitions were written in full sentences with a well defined grammatical structure. The participants were asked to complete a questionnaire, rating the sentence pairs (each presented on a separate page) similarity from 0.0 (min similarity) to 4.0 (maxi similarity). In each questionnaire the sentence pair sheets and the order of the two sentences composing each pair were presented randomly. This questionnaire was organized in a way to prevent any bias that can be inducted by the order of presentation. All of the 65 sentence

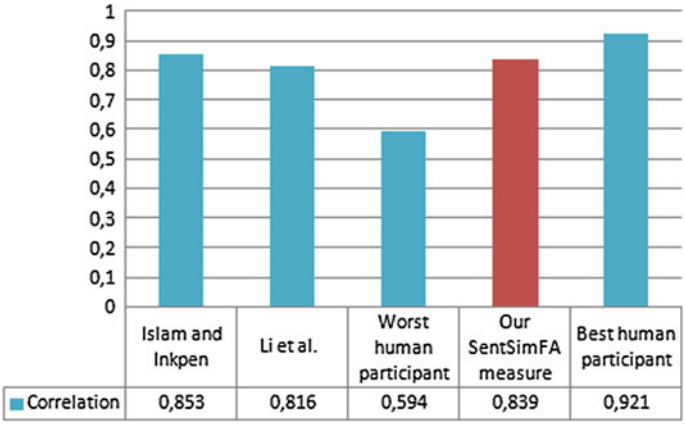


Fig. 1 The $SenSim_{FA}$ similarity measure compared to baselines on Li et al.

pairs were assigned a semantic similarity score computed as the mean of the participants' judgments. So, for an even similarity distribution, a subset of 30 sentence pairs was chosen.

The following pair of sentences is an example of Li et al. dataset [13]:

13. boy:rooster

S1 *A boy is a child who will grow up to be a man.*
S2 *A rooster is an adult male chicken.*

Table 1 presents the mean of the human similarity scores along with Li et al. similarity method scores [13] and our proposed sentence similarity scores.

Figure 1 presents the correlation between the scores produced by our method and the average of the scores given by the human judges. According to Fig. 1, our results are better than the results of the method of Li et al. [13], based on a lexical co-occurrence network and it is comparable with Islam and Inkpen method [24]. The third and the last bars in the figure show how much the human judges varied from their mean.

4 Conclusion and Perspectives

Text similarity is fundamental to various fields such as Cognitive Science and Artificial Intelligence. With the increasing complexity of data it became necessary to develop similarity measures able to keep a semantic relevance with respect to a certain application domain such as Computational Intelligence and related areas. In fact, several studies on Natural Language Processing were motivated by text semantic similarity measures, such as the work of Hirst and Budanitsky [30] in

which they investigated the usefulness of the semantic similarity in the problem of spelling correction, where actual spelling errors are detected and corrected automatically. This accentuates the importance of relying on a reliable and robust similarity measure.

In this paper, we proposed a novel approach for measuring semantic similarity between short text segments. The experimental results are promising. There are several lines of future work that we intend to work on, to mention, using our text similarity measure for image retrieval from the Web. Besides, we will proceed with the evaluation of our approach on other datasets in order to confirm its performance.

References

1. McDonald, S.: Exploring the validity of corpus-derived measures of semantic similarity. In: 9th Annual CCS/HCRC Postgraduate Conference, University of Edinburgh (1997)
2. Miller, G.A., Charles, W.G.: Contextual correlates of semantic similarity. Lang. Cogn. Proc. **6**(1), 1–28 (1991)
3. Elkhlifi, A., Bouchlaghem, R., Faiz, R.: Opinion extraction and classification based on semantic similarities. In: 24th International Florida Artificial Intelligence Research Society Conference. AAAI Press, Palm Beach, Florida, USA (2011)
4. Erkan, G., Radev, D.R.: Lexrank: graph-based lexical centrality as salience in text summarization. J. Artif. Intell. Res. **22**(1), 457–479 (2004)
5. Somers, H.: Review article: example-based machine translation. Mach. Transl. **14**(2), 113–157 (1999)
6. Coelho, T.A.S., Calado, P.P., Souza, L.V., Ribeiro-Neto, B., Muntz, R.: Image retrieval using multiple evidence ranking. IEEE Trans. Knowl. Data Eng. **16**(4), 408–417 (2004)
7. Ko, Y., Park, J., Seo, J.: Improving text categorization using the importance of sentences. Inf. Process. Manage. **40**(1), 65–79 (2004)
8. Liu, T., Guo, J.: Text similarity computing based on standard deviation. In: International Conference on Advances in Intelligent Computing: Part I, pp. 456–464, Hefei, China (2005)
9. Wegrzyn-Wolska, K., Szczepaniak, P.: Classification of RSS-formatted documents using full text similarity measures. In: 5th International Conference on Web Engineering, pp. 400–405, Sydney, Australia, (2005)
10. Zhang, J.: Calculating statistical similarity between sentences. Convergence **6**(2), 22–34 (2011)
11. Achananuparp, P., Hu, X., Shen, X.: The evaluation of sentence similarity measures. In: 10th International Conference on Data Warehousing and Knowledge Discovery, pp. 305–316. Springer, Heidelberg (2008)
12. Mohri, M.: Edit-distance of weighted automata. In: Champarnaud, J.-M., Maurel, D. (eds.) 7th International Conference, pp. 1–23, CIAA (2002)
13. Li, Y., McLean, D., Bandar, Z., O'Shea, J., Crockett, K.: Sentence similarity based on semantic nets and corpus statistics. IEEE Trans. Knowl. Data Eng. **18**(8), 1138–1150 (2006)
14. Miller, G., Beckwith, R., Fellbaum, C., Gross, D., Miller, K.: Introduction to WordNet: an on-line lexical database. Int. J. Lexicogr. **3**(4), 235–244 (1993)
15. Mihalcea, R.: Corpus-based and knowledge-based measures of text semantic similarity. In: 21st National Conference on Artificial Intelligence, vol. 1, pp. 775–780, Boston, Massachusetts (2006)
16. Turney, P.: Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In: 12th European Conference on Machine Learning, pp. 491–502, London, UK (2001)

17. Landauer, T., Dumais, S.: A solution to Plato's problem: the latent semantic analysis theory of acquisition, induction, and representation of knowledge. Psychol. Rev. **104**(2), 211–240 (1997)
18. Jiang, J.J., Conrath, D.W.: Semantic similarity based on corpus statistics and lexical taxonomy. In: 10th International Conference on Research on Computational Linguistics, pp. 19–33 (1997)
19. Leacock, C., Chodorow, M.: Combining local context and WordNet similarity for word sense identification. In: Fellbaum, C. (ed.) WordNet: An Electronic Lexical Database, pp. 265–283. MIT Press, Cambridge (1998)
20. Lesk, M.: Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In: 5th ACM Annual International Conference on Systems Documentation, pp. 24–26 (1986)
21. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: 14th International Joint Conference on Artificial Intelligence, pp. 448–453, Montreal, Quebec, Canada (1995)
22. Wu, Z., Palmer, M.: Verbs semantics and lexical selection. In: 32nd Annual Meeting on Association for Computational Linguistics, pp. 133–138, Las Cruces, New Mexico, (1994)
23. Islam, A., Inkpen, D.: Semantic similarity of short text. In International Conference on Recent Advances in Natural Language Processing, Borovets, Bulgaria (2007)
24. Islam, A., Inkpen, D.: Semantic text similarity using corpus-based word similarity and string similarity. ACM Trans. Knowl. Discovery Data **2**(2), 1–25 (2008)
25. Inkpen, D.: Semantic similarity knowledge and its applications. Studia Universitatis BabesBolyai Informatica **LII**(1), 11–22 (2007)
26. Islam, A., Inkpen, D.: Second order co-occurrence PMI for determining the semantic similarity of words. In: 5th International Conference on Language Resources and Evaluation, pp. 1033–1038 (2006)
27. Akermi, I., Faiz, R.: Hybrid method for computing word-pair similarity based on web content. In: 2nd International Conference on Web Intelligence, Mining and Semantics, Craiova, Romania (2012)
28. Rubenstein, H., Goodenough, J.: Contextual correlates of synonymy. Commun. ACM **8**(10), 627–633 (1965)
29. Sinclair, J.: Collins Cobuild English Dictionary for Advanced Learners. HarperCollins, New York (2001)
30. Hirst, G., Budanitsky, A.: Correcting real-word spelling errors by restoring lexical cohesion. J. Nat. Lang. Eng. **11**, 87–111 (2005)

Object Recognition with the Higher-Order Singular Value Decomposition of the Multi-dimensional Prototype Tensors

Bogusław Cyganek

Abstract In the paper an extension of object recognition based on the Higher-Order Singular Value Decomposition (HOSVD) to the 4th dimension is discussed. HOSVD based object recognition expands the concept of object recognition in the pattern spaces spanned by the PCA decomposition of vector patterns into the higher dimensions. However, contrary to the PCA, in the HOSVD the bases of the pattern space are tensors rather than 1D vectors. Nevertheless, the already presented works on HOSVD recognition were limited to the images with only scalar valued pixels. In the proposed framework images are allowed to contain multi-dimensional pixels, which adds an additional dimension to the pattern tensor. The proposed method opens new possibility of the HOSVD based recognition to color or other multi-valued images. Experimental results show improved accuracy as compared to the scalar valued data, as well as fast execution time.

Keywords Higher order singular value decomposition (HOSVD) · PCA · Recognition · Multi-dimensional data · Tensor classifier

1 Introduction

Recently, tensor based methods found great interest in pattern recognition domain. In computer vision these were also shown to provide excellent results in object recognition [1, 5, 16, 18]. Their success lie in the fact that tensor based methods explicitly account for multidimensional nature of processed data.

In this paper an extension of object recognition based on the Higher-Order Singular Value Decomposition (HOSVD) to the 4th dimension is presented.

B. Cyganek (✉)
AGH University of Science and Technology, Al. Mickiewicza 30, 30-059 Kraków, Poland
e-mail: cyganek@agh.edu.pl

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_33,

HOSVD method of object recognition exploits the concept of object recognition in the pattern spaces. The best known method in this category is the PCA decomposition. However, PCA operates with vector-like data. Also, the bases of the spaces spanned by PCA are vectors. However, contrary to the PCA in the HOSVD the bases of the pattern spaces are tensors rather than 1D vectors.

The HOSVD classifier shows good results when applied to multi dimensional data, such as images [5, 16]. This is due to tensor processing which allows separate control of all intrinsic dimensions of data. Let us recall that in the classical PCA-based classification method, images are first vectorized and, in the result, the obtained subspaces are spanned by vector bases [17]. However, this also leads to the lost of information on spatial relations among pixels. Contrary to this, in the HOSVD method the bases of the orthogonal pattern subspace are spanned by the higher-dimensional tensors. Nevertheless, to the best of our knowledge, the reported works on HOSVD recognition were always limited to the images with only scalar valued pixels [3, 4, 16]. In the case of scalar valued images, the HOSVD bases are two-dimensional. Contrary to these, in the proposed framework images are allowed to contain multi-dimensional pixels, which adds an additional dimension to the pattern tensor. However, thanks to this, the proposed method opens new possibility for the HOSVD based recognition of color or any-length pixel images. In other words, due to the proposed extension to multi-valued pixels, the base tensors are three-dimensional, as will be discussed. When classifying an unknown pattern, the tested patters are projected onto the subspaces of each of the trained classes and the best fitting projection is returned. However, in the tensor case the bases are multidimensional, as already mentioned. The proposed method can be also extended to higher dimensions, leading to the 5th, 6th, and higher dimensional bases, depending on a type of the input signals.

The method was tested on the problem of face recognition in the difficult set of color face images. Experimental results show improved accuracy as compared to only scalar valued, i.e. gray-valued, images. The proposed method can be compared to the methods reported by other researchers [10, 12, 14], although it was not optimized particularly for the face recognition problem.

Apart from the above, in the proposed system a parallel version of the HOSVD algorithm is applied. Concurrency is obtained through the functional and data decompositions on different levels of computations. Parallel operation is also possible at the response time of the system, since each subspace projection can be computed independently.

The paper is organized as follows. In the next section the tensor based pattern recognition framework is presented. Experimental results are presented and discussed in Sect. 3. In this section also implementation details are provided. Conclusions are presented in Sect. 4 of this paper.

2 Multi-valued Image Recognition in the Tensor Subspaces

Tensors in data mining can be interpreted as multidimensional data-cubes. Processing and analysis of multi-dimensional data, such as images, fits well into this framework. An example of image representation in a tensor form is presented in Fig. 1. However, an analysis of data content requires proper decomposition of pattern tensors. In this respect, HOSVD is one of the most powerful tensor decomposition methods [1, 5–7, 13]. As shown, HOSVD can be used to build orthogonal spaces which can be then used for pattern recognition in a way similar to the subspace projection methods [8, 17]. This procedure is briefly outlined in this section. More information on tensors in signal processing can be found in literature, e.g. [1, 5–7, 13].

Let us briefly present the underlying theory behind multi-dimensional data representation and analysis by means of tensors and their decompositions. In this respect, the first concept is the *k-mode vector* of a *P*th order tensor $\mathcal{T} \in \Re^{N_1 \times N_2 \times \ldots N_P}$. It is a vector obtained from the elements of $\mathcal{T}$ by changing only one index n_k, and keeping all other fixed. The second important concept is the operation of the *k*-mode flattening of a tensor. For a tensor $\mathcal{T}$, a result of its *k*-mode flattening is the following matrix [6, 13].

$$\mathbf{T}_{(k)} \in \Re^{N_k \times (N_1 N_2 \ldots N_{k-1} N_{k+1} \ldots N_P)}. \tag{1}$$

Now we can define the HOSVD decomposition for pattern tensors constructed of a series of 3D images, that is, each having two spatial and one pixel-value coordinates. Therefore our pattern tensors will be four-dimensional (4D). Thus, the further discussion is confined to the 4D tensors.

As already mentioned, important information about the pattern space are revealed after its HOSVD decomposition. That is, any 4D tensor can be represented as the following tensor product [6, 13]

$$\mathcal{T} = \mathcal{Z} \times_1 \mathbf{S}_1 \times_2 \mathbf{S}_2 \times_3 \mathbf{S}_3 \times_4 \mathbf{S}_4. \tag{2}$$

In the above formula $\mathbf{S}_k$ are *unitary* matrices of dimensions $N_k \times N_k$, (called mode matrices), and $\times_j$ denotes the so called *j*-mode product of a tensor and a matrix. The mode matrices $\mathbf{S}_k$ are responsible for representation of column spaces related to each different index (dimension) of a tensor. On the other hand, the tensor $\mathcal{Z} \in \Re^{N_1 \times N_2 \times N_3 \times N_4}$ is called a core tensor, and fulfills properties of the subtensor orthogonality and decreasing energy value [6, 13].

De Lathauwer [6] proposed a method of computation of the HOSVD which is based on successive application of the matrix SVD decompositions to the flattened matrices of a given tensor. The HOSVD decomposition algorithm for a 4-dimensional tensor $\mathcal{T}$ is outlined in Fig. 3. It can be easily observed that computation of the HOSVD requires a series of computations of the SVD decompositions on the flattened tensor representations (i.e. matrices). These are

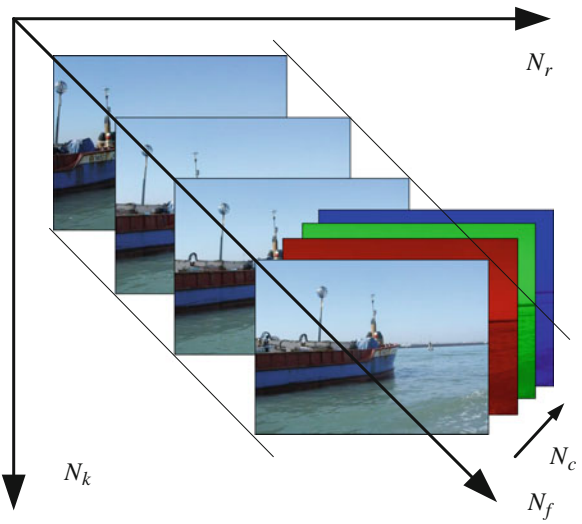


Fig. 1 A series of color images represented as a four-dimensional data cube. This can be seen as a 4-dimensional tensor

independent versions (different modality) of the input tensor. Therefore it is possible to run all these SVD decompositions concurrently, which must be synchronized on a barrier just before computation of the core tensor in (8), however. Figure 3 shows the algorithm for computation of the HOSVD. Its grayed area can be run concurrently, as discussed.

Let us now observe that, thanks to the commutative properties of the *k*-mode multiplication, for each mode matrix $\mathbf{S}_i$ in (2) the following sum can be constructed

$$\mathcal{T} = \sum_{h=1}^{N_P} \mathcal{T}_h \times_4 \mathbf{s}_4^h. \tag{3}$$

Further, it can be shown that tensors

$$\mathcal{T}_h = \mathcal{Z} \times_1 \mathbf{S}_1 \times_2 \mathbf{S}_2 \times_3 \mathbf{S}_3 \tag{4}$$

in (3) constitute the basis tensors and $\mathbf{s}_P^h$ are columns of the unitary matrix $\mathbf{S}_P$ [6, 13]. Thus, they form an orthogonal basis which spans a subspace. This property is used to construct a HOSVD based classifier [3, 16]. However, in this case they are *3D tensors*, as shown in Fig. 2. This constitutes *a novelty* of the proposed method.

In each subspace spanned by tensors $\mathcal{T}_\mathrm{h}$, object recognition can be formulated as a testing of a distance of a given test pattern $\mathbf{P}_x$ to its projections in each of the spaces spanned by the set of the bases $\mathcal{T}_\mathrm{h}$ in (4). That is, the following optimization process needs to be solved [16]:

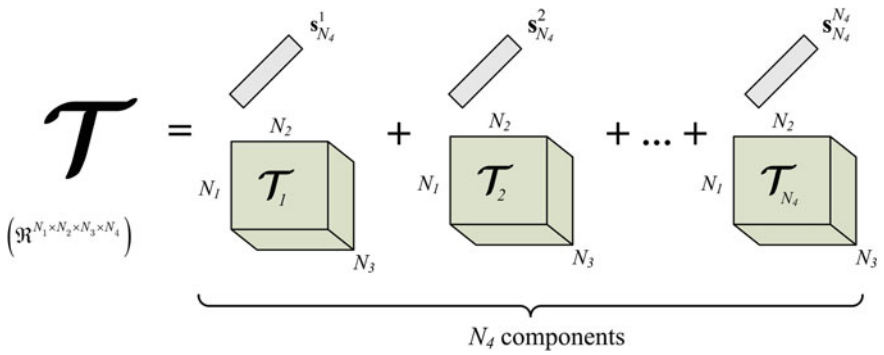


Fig. 2 Decomposition of the pattern tensor into a sum of products of the 3D base tensors and mode vectors. The base tensors form an orthonormal subspace used for pattern recognition

$$\min_{i,c_h^i} \underbrace{\left\| \mathbf{P}_x - \sum_{h=1}^{K} c_h^i \mathcal{T}_h^i \right\|^2}_{Q_i}, \tag{5}$$

where the scalars c_h^i denote unknown coordinates of the pattern $\mathbf{P}_x$ in the space spanned by $\mathcal{T}_h^i$, and $K \leq N_P$ denotes a number of chosen dominating components. It can be further shown that to minimize (5) we need to maximize the following value [5, 16]

$$\hat{\rho}_i = \sum_{h=1}^{K} \left\langle \hat{\mathcal{T}}_h^i, \hat{P}_x \right\rangle^2, \tag{6}$$

where $\left\langle \hat{\mathcal{T}}_h^i, \hat{P}_x \right\rangle$ denotes the inner product operation. In other words, the (single) HOSVD based classifier returns a class i for which its ρ_i from (6) is the largest. It is worth recalling that in our framework the base tensors $\mathcal{T}_\mathrm{h}$ are 3D. However, in the response time, computation of the inner product in accordance with (6) is very fast.

The main difference of the tensor based approach to building the spanning pattern subspaces thus lies in 4-times computed column space, whereas in the PCA method this is computed once on a vectorized data, no matter what dimensionality they had originally.

More details on implementation of the HOSVD decomposition can be found in [5]. Figure 3 contains pseudo-code of the four-dimensional HOSVD decomposition. The grayed area represents the part of the algorithm which can be run concurrently. This can lead to the computation speed-up.

begin

for each k=1, …, 4 **do**

1. From Eq.(1) compute k-mode flattened matrix $\mathbf{T}_k$ of tensor $\mathcal{T}$

2. Compute $\mathbf{S}_k$ from the SVD decomposition of $\mathbf{T}_k$

$$\mathbf{T}_k = \mathbf{S}_k \mathbf{V}_k \mathbf{D}_k^T \tag{7}$$

end

Compute the core tensor from all matrices $\mathbf{S}_k$

$$\mathcal{Z} = \mathcal{T} \times_1 \mathbf{S}_1^T \times_2 \mathbf{S}_2^T \times_3 \mathbf{S}_3^T \times_4 \mathbf{S}_4^T \tag{8}$$

end

Fig. 3 Algorithm for computation of the Higher-Order Singular Value Decomposition of tensors of 4th dimensions. The shaded steps can be executed concurrently

3 Experimental Results

The presented method was implemented in C++, supported by the *DeRecLib* software from [5] and the *OpenMP* library for the multicore processing [2, 15]. The experiments were carried out on the computer with 8 GB RAM and the Pentium® Quad Core Q 820 microprocessor (eight cores due to the hyper-threading technology [11]).

In order to evaluate the method a database with multi-valued features is required. For this purpose the Georgia Tech Face Database (GTFD) [9] was employed which contains color images. Images of persons in the GTFD are acquired in different sessions, various poses and illuminations. Some of the photographed persons in some sessions wear glasses, as well as many persons were photographed from different viewpoints. Therefore, this database is known as highly demanding for the face recognition algorithms [10, 14]. It contains images of 50 persons taken at multiple sessions. There are 750 images with 15 images per person. The images for each person contain the frontal pose, as well as different facial expressions, various illuminations, and scale. Exemplary faces from this database are shown in Fig. 4. However, contrary to other works in our experiments the images are not preprocessed.

Thus, for each 15 available exemplars, the experiment was carried out always randomly taking 12 images of a person for training, and then testing on the remaining 3 images. Such tests were run ten times and the average results are reported in Table 1.

Figures 5 and 6 depict slices of the 3D base tensors $\mathcal{T}_\mathrm{h}$ computed in accordance with the formula (4) for two subjects shown in Fig. 4.

Fig. 4 Examples of the test images from the Georgia Tech Face Database [9]. There are 50 subjects, for each there are 15 images from which 10 were randomly selected for training and the remaining 5 for testing in different runs of the system

Table 1 Average accuracy of face recognition with the multi-dimensional 3D and 4D HOSVD based classifier (first column)

Experiment conditions	Accuracy (a) (%)	Accuracy (b) (%)
3D tensors (scalar)	83.4	87.2
4D tensors (multi-valued)	87.3	91.9

Accuracy measured with a condition on best match separation of at least 1 % (second column)

In our experiments parallelism on different levels of computations were measured. Also, each parallel realization was analyzed in the context of memory requirements. The parallel implementation allows up to two times speed-up in computations, as compared to a serial version.

To verify our assumptions the experiments were performed the same number of times for the monochrome, as well as color versions of the same images. Results show that utilization of color information, in the form of a 4th dimension of the input pattern tensor, leads to better accuracy. The number of components used in (6) was 7 in all experiments. Lower values led to slightly smaller accuracy, although even for 3 first components the differences do not exceed 1 % in overall accuracy. On the other hand, higher values resulted in no higher accuracy, requiring more computations at the same time.

It is worth noticing the difference of the proposed HOSVD based method compared with the PCA approach. The proposed methods works better since multi-dimensional data (color faces in our case) are decomposed independently in each dimension (four in our experiments), whereas PCA does decomposition only

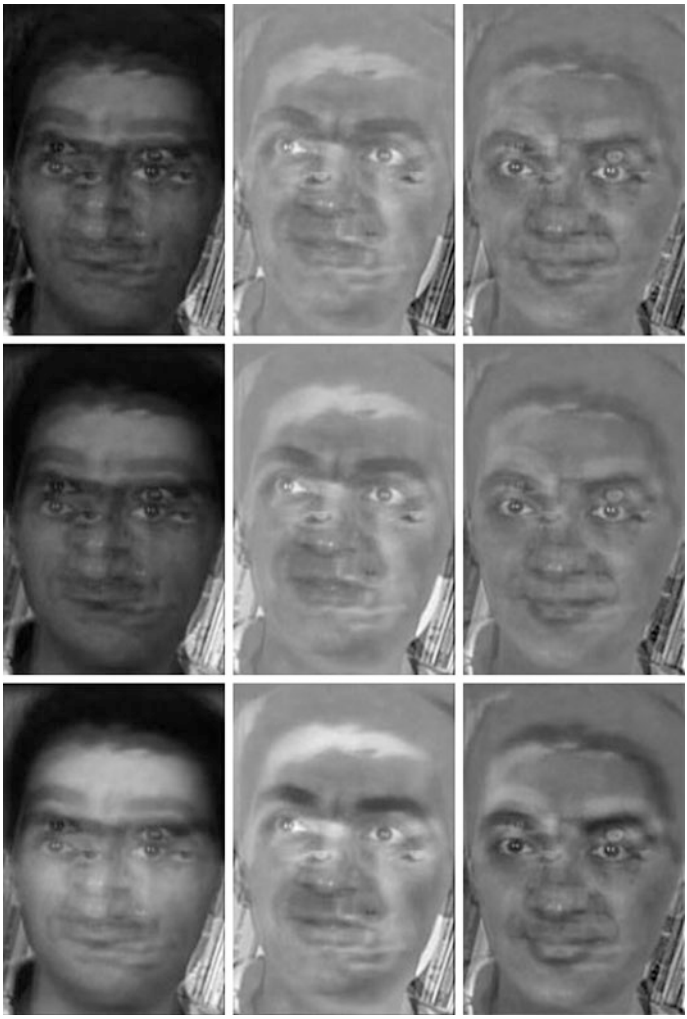


Fig. 5 Slices of the three base tensors of the second subject shown in Fig. 4

in one dimension regardless of the internal dimensionality of data. Thus, with the HOSVD a more in-depth information of the contained patterns is extracted which leads to higher accuracy. However, we would like to emphasize that the proposed method is not the best face recognition algorithm. Especially problematic is recognition of multi class patterns, containing dozens of classes. In this case the variability between classes can be even lower than within a single class which leads to lowered accuracy. To remedy the situation we added an additional constraint on the best match value, as well as the second best match [i.e. the value in formula (6)]. More precisely, the following condition is checked

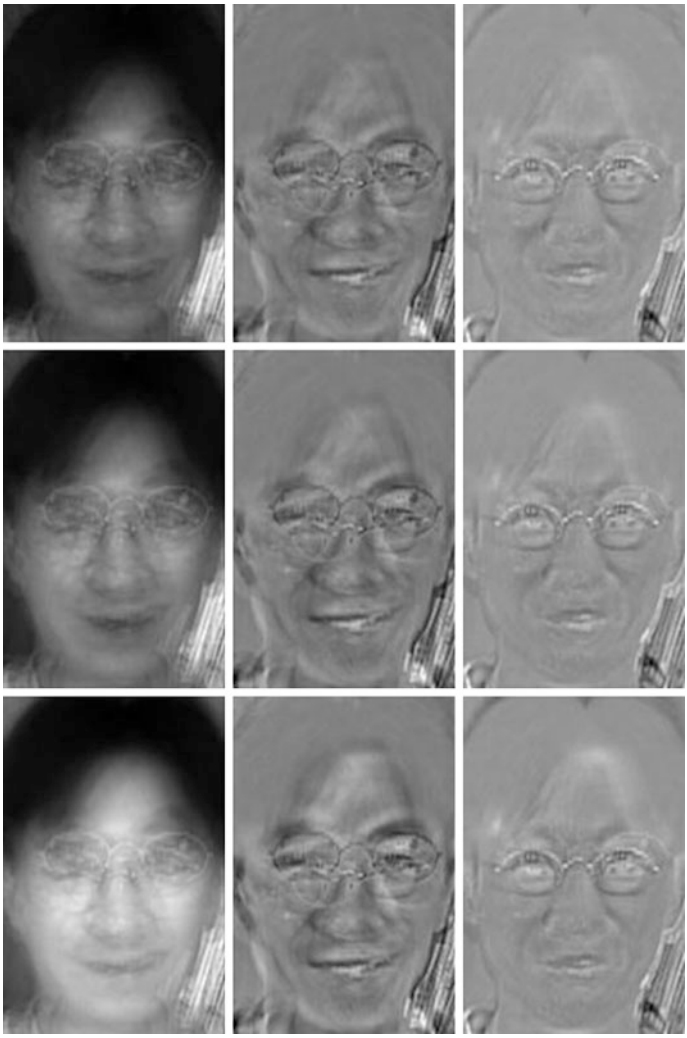


Fig. 6 Slices of the three base tensors of the fourth subject shown in Fig. 4

$$1 - \frac{\hat{\rho}_{2nd}}{\hat{\rho}_{1st}} > \tau, \tag{9}$$

where $\hat{\rho}_{1st}$, $\hat{\rho}_{2nd}$ denote the 1st largest and the 2nd largest value of the residuum computed in accordance with (6), respectively, and τ denotes a threshold value. In our experiments the latter was set to 1 %. Application of (9) allowed an increase of accuracy at a cost of some missing recognitions (false negatives), as shown in Table 1.

However, our purpose was to show the difference between the HOSVD operating in different dimensions. That is, a difference between the 3D and 4D pattern tensors. Our experimental results show that operations in the higher dimensional space lead to better results, at a negligible additional computations in the response stage allowing real-time operations.

4 Conclusions

In this paper a new version of the HOSVD based tensor classifier is presented. This is an extension of the highly successful HOSVD classifier to the 4th dimension, representing multi-valued pixels of the input images. Thanks to this, the input prototypes can contain other than scalar values. Thus, the proposed method allows recognition of color or other multi-valued signals. Experimental results in color face recognition show improved accuracy as compared to the scalar valued representations. Summarizing, the key features of the presented method are as follows:

- The method achieves high accuracy.
- The proposed pattern recognition method can be used to any patterns (not only images).
- The method can be easily extended to higher dimensional "cubes" of data (such as video, hyperspectral, etc.).
- The parallel algorithms for training and testing were outlined.
- The method allows real-time operation even in software implementation (simple inner product computation).

Nevertheless it is in order to mention some problems associated with the proposed method. First, size of the input tensor very frequently is too high to fit into the memory. This also concerns time necessary for the HOSVD decomposition. Therefore, future research is to develop methods which allow partial computation of the HOSVD. The second problem is threshold necessary to distinguish the in-class from the ext-class patterns. In the presented experiments this is achieved by comparison with the external face classes. However, in some practical situations such ext-class examples are not available. Further research will focus also on testing the proposed method with different datasets, application of image transformations, such as computation of the extended structural tensor, as well as further extension to classification of the video patterns, i.e. processing of the 5th order pattern tensors. Also interesting is application of the presented method to data other than images.

Acknowledgments The work was supported in the years 2013–2014 from the funds of the Polish National Science Centre NCN, contract no. DEC-2011/01/B/ST6/01994.

References

1. Chapman, B., Jost, G., Van Der Pas, A.R.: Using OpenMP: Portable Shared Memory Parallel Programming. MIT Press, Cambridge (2008)
2. Cichocki, A., Zdunek, R., Amari, S.: Nonnegative matrix and tensor factorization. IEEE Signal Process. Mag. **25**(1), 142–145 (2009)
3. Cyganek, B.: An Analysis of the Road Signs Classification Based on the Higher-Order Singular Value Decomposition of the Deformable Pattern Tensors. Lecture Notes in Computer Science, vol. 6475, pp. 191–202. Springer, Berlin (2010)
4. Cyganek, B.: Adding parallelism to the hybrid image processing library in multi-threading and multi-core systems. IEEE International Conference on Networked Embedded Systems for Enterprise Applications, pp. 103–110 (2011)
5. Cyganek, B.: Object Detection in Digital Images. Theory and Practice. Wiley, Hoboken (2013)
6. de Lathauwer, L.: Signal processing based on multilinear algebra. Ph.D dissertation, Katholieke Universiteit Leuve (1997)
7. de Lathauwer, L., de Moor, B., Vandewalle, J.: A multilinear singular value decomposition. SIAM J. Matrix Anal. Appl. **21**(4), 1253–1278 (2000)
8. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley, Hoboken (2000)
9. Georgia Tech Face Database. http://www.anefian.com/research/face_reco.htm (2013)
10. Goel, N., Bebis, G., Nefian, A.V.: Face recognition experiments with random projections. SPIE Conference on Biometric Technology for Human Identification (2005)
11. Intel. www.intel.com (2013)
12. Jiang, X., Mandal, B., Kot, A.: Eigenfeature regularization and extraction in face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **30**(3), 383–394 (2008)
13. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM Rev. **51**(3), 455–500 (2008)
14. Nefian, A.V.: Embedded Bayesian networks for face recognition. IEEE International Conference on Multimedia and Expo, August (2002)
15. OpenMP. www.openmp.org (2013)
16. Savas, B., Eldén, L.: Handwritten digit classification using higher order singular value. Pattern Recogn. **40**(3), 993–1003 (2007)
17. Turk, M., Pentland, A.: Eigenfaces for recognition. J. Cogn. Neurosci. **3**(1), 71–86 (1991)
18. Vasilescu, M.A.O., Terzopoulos, D.: Multilinear Analysis of Image Ensembles: TensorFaces. Lecture Notes in Computer Science, vol. 2350, 447–460. Springer, Berlin (2002)

A Quality Driven Approach for Provisioning Context Information to Adaptive Context-Aware Services

Elarbi Badidi

Abstract The growing adoption of the Service Oriented Architecture (SOA) for provisioning services and the proliferation of Internet-enabled handheld devices are changing the services landscape. Users are increasingly demanding services that can adapt to their current context. In this paper, we propose a framework for provisioning context information to adaptive services. The framework relies on negotiated Context Level Agreements (CLAs) between context-consumers (adaptive services) and context-providers by means of a context broker. The CLA specifies the context information and the agreed upon level of quality-of-context (QoC) that the context-provider shall deliver. We describe the components of the framework and the CLA negotiation process. One of the advantages of the approach is that context-providers can provide several types of context information at different QoC levels. Moreover, the publish/subscribe model allows the broker to be aware of significant variations in QoC offerings; and consequently, be able to monitor the execution of CLAs.

Keywords Context-aware services · QoC · Context broker · Context negotiation

1 Introduction

As a result of the phenomenal proliferation of Internet-enabled handheld devices—such as iPhone, iPad, and Android-based smartphones and tablets—, and the growing adoption of SOA by businesses for implementing and deploying their business applications on the Web, users increasingly require services that can meet their functional requirements and adapt to their current context. Context

E. Badidi (✉)
College of Information Technology, United Arab Emirates University,
15551Al-Ain, United Arab Emirates
e-mail: ebadidi@uaeu.ac.ae

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_34,

information determines behavior and strategies of context-aware systems. Several definitions of the notion of context have been provided in the literature [1, 2]. The amount of information that can be categorized as context information is extremely wide. Geo-location, time, temperature, humidity, pressure, and mobile user activity are the most common context indicators.

The development of adaptive context-aware services requires two main components: context management and dynamic provisioning of context information. Context management deals essentially with sensing, storing raw context data, aggregating and reasoning to infer high-level context information. Adaptive context-aware services typically obtain high-level context information from various context-providers (or services) that aggregate raw context data sensed by various sensors and mobile devices. Many works proposed, designed, and implemented frameworks and middleware infrastructures for managing context information and providing users with context-aware services [3–7]. Likewise, many surveys investigated the features and shortcomings of existing systems [8–10].

One of the challenging issues regarding the provisioning of context information is assessing the *quality of context information* (QoC). The QoC concept is introduced in Sect. 2.2. Furthermore, two key aspects of context provisioning, which are not addressed by most context management systems, are context negotiation and the management of the continual variations in QoC delivered by context-providers. These variations in delivered QoC are mainly due to discrepancies in the process of obtaining high-level context information from raw context information. They are also due to differences in the quality of sensing devices. Context-consumers may be notified of significant QoC changes only after a certain period and can suffer degradation in the QoC they get from context-providers.

To cope with the issues of QoC-driven selection of context-providers, context information negotiation, and CLA compliance, we propose in this work a framework for brokered CLA negotiation and monitoring. The main component of the framework is the *Context Broker*, which mediates between context-consumers and context-providers to reach agreements with respect to context information and QoC levels to deliver. The context broker has the following components: *Profile Manager*, *Context Request Dispatcher*, *CLA Manager*, *Notification Manager*, and the *Coordinator*. The Notification Manager implements a publish/subscribe model to deal with notifications on significant variations in QoC offerings of context-providers.

The remainder of the paper is organized as follows. Section 2 presents background information on the concepts of Context Level Agreements and Quality-of-Context as well as related work on the issues of context negotiation and provisioning. Section 3 gives an overview of the proposed framework. Section 4 describes the CLA negotiation protocol. Section 5 describes the interactions among the framework's components. Finally, Sect. 6 concludes the paper and describes future work.

2 Background and Related Work

2.1 Context Level Agreements

A CLA is an arrangement between a context-provider and a context-consumer concerning the guarantees of delivered context information. It describes common understandings and expectations between the two parties. The guarantees concern the context information and the QoC levels to be delivered. The typical sections of a CLA are:

- *Parties*: represents the parties involved in the CLA and their respective roles (context-consumer and context-provider).
- *Activation time*: represents the period of time at which the CLA will be valid.
- *Scope*: defines the types of context information covered by the agreement.
- *Context-level objectives* (*CLOs*): represents the levels of QoC that both parties agree on, and habitually include a number of quality indicators such as accuracy and freshness.
- *Penalties*: specifies the penalties for not meeting the stated context level objectives, such as getting discount or having the right to terminate the contract in light of unsatisfactory QoC levels.
- *Exclusions*: specifies what is not covered in the CLA.
- *Administration*: defines the processes to assess the CLA objectives, and describes the responsibility of the context-provider regarding the control of each of these processes.

Achieving the quality objectives may require from the context-provider to establish and manage a number of CLAs, all with potentially varying provisioning requirements. Context level is a performance measure of how well the context-provider is responding to incoming requests for context information. CLOs are the goals of the context-provider, such as the freshness or precision of delivered context information that the context-provider can guarantee. They represent a commitment of the context-provider to maintain a particular level of context delivery in a predefined period of time. A typical CLA may have the following CLOs: context precision, degree of freshness of context information, and probability of correctness.

2.2 Quality-of-Context

Existing context-aware systems implicitly consider that context information used to adapt their services is correct and reliable. This hypothesis is obviously not well matched when considering the effective conditions in real pervasive situations, where raw context data are obtained using various, and possibly unreliable sensors.

To cope with this reliability issue, context information is characterized by some properties referred in literature as QoC indicators. Buchholz et al. [11] defined the QoC as: "*Quality of Context (QoC) is any information that describes the quality of information that is used as context information. Thus, QoC refers to information and not to the process nor the hardware component that possibly provide the information.*"

Buchholz et al. [11] and Sheikh et al. [12] identified the following QoC indicators: *precision*, *freshness*, *temporal resolution*, *spatial resolution*, and *probability of correctness*. Precision represents the granularity with which context information describes a real world situation. Freshness represents the time that elapses between the determination of context information and its delivery to the requester. Spatial resolution represents the precision with which the physical area, to which an instance of context information is applicable, is expressed. Temporal resolution is the period of time during which a single instance of context information is relevant. Probability of correctness represents the probability that a piece of context information is correct.

Taking into consideration QoC in both the design and management of context-aware systems has been recognized in many works on context-awareness [3, 11–14]. Few works investigated the issues of modeling and measuring QoC. Krause and Hochstatter [15] described the requirements for modeling QoC through the analysis of the context dissemination process. Filho et al. [16] described a OWL-DL QoC model and methods for measuring QoC by taking into account the fact that context information might be modified after sensing and described into a high semantic level. Manzoor et al. [17] considered QoC to be composed of two components, QoC sources and QoC parameters. QoC sources represent the information concerning the sources, which collect context information, the subjects about which context information is collected, and the environment where context information is sensed and collected.

2.3 Context Negotiation and Provisioning

Only a small number of research works in the area of context-awareness have investigated the context negotiation issue. Many works designed and implemented middleware infrastructures to manage context information. Other works investigated the design and implementation of adaptive context-aware services. Furthermore, many surveys were conducted to understand and compare the features and shortcomings of existing systems [8–10]. Baldauf et al. [8] provided a survey on many context-aware systems and compared them in terms of sensing support, their context models, context information processing, security, and privacy. Henricksen et al. [9] conducted a survey on middleware-based context-aware systems and compared the features of few systems such as The Context Toolkit, CFN solar, The Context Fabric, Gaia, and RCSM in terms of their support of heterogeneity, mobility, scalability, privacy, traceability and control, tolerance for

failures, and ease of development and configuration. Truong et al. [10] presented a survey on context-aware Web service-based systems. Bettini et al. [18] described current context modeling and reasoning techniques. Modeling approaches mainly include key-values models, markup scheme models, graphical models, object-oriented models, logic-based models, and ontology-based models.

With the advent of service oriented computing, a number of research works investigated the design and implementation of specialized services, called context services, to capture, store, analyze, and aggregate raw data to deduce high-level context information. Schmidt et al. [19] designed and implemented a generic context service with a modular architecture that allows for context collection, discovery and monitoring. This context service provides a Web service interface that allows its integration in heterogeneous environments. The implementation uses OWL to describe context information and SPARQL to query and monitor context information. Coronato et al. [20] proposed a semantic context service to support smart offices using rules and ontologies to infer high-level context information, such as lighting and sound level, from low-level raw information acquired from context sources.

To the best of our knowledge, the most significant work that investigated the issue of context negotiation and establishing and negotiating CLAs is the work of Khedr and Karmouch [21]. In this work, the authors described a multi-agent middleware, which uses a negotiation protocol, to facilitate the development of adaptive context-aware personalized applications, and an ontology model to represent context information.

We believe that given that clients increasingly require quality adaptive services, context-aware services have to negotiate CLAs with context-providers to guarantee some levels of the quality of context information they obtain from these providers if they want to remain competitive. Our proposed framework provides the means to negotiate the CLOs and the other terms of CLAs between context-consumers and context-providers. Our approach differs from Khedr and Karmouch approach in that our framework relies on a context broker instead of agents to negotiate CLAs on behalf of context-consumers and a publish/subscribe model for monitoring QoC provisioning.

3 Framework Overview

In the proposed framework, the Context Broker decouples consumers from context-providers. It is mainly in charge of reaching CLAs between context-consumers and context-providers. Figure 1 depicts our CLA-based framework for context provisioning. The main components of the framework are context-consumers, the Context Broker, and context-providers.

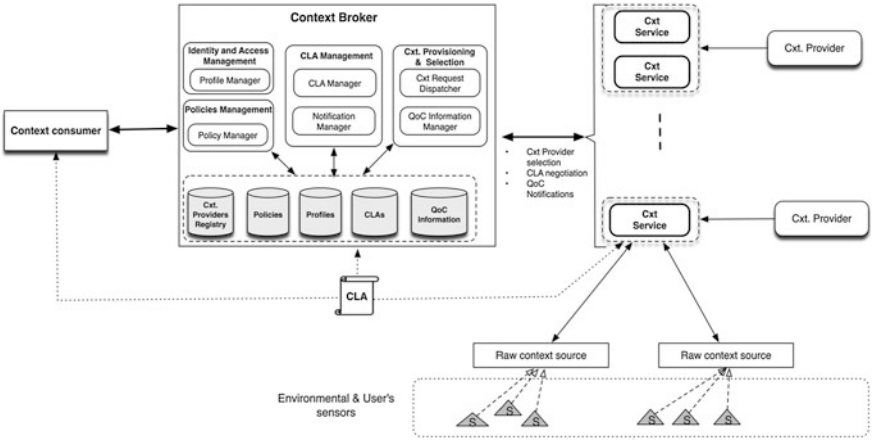


Fig. 1 CLA-based framework for QoC-aware context provisioning

3.1 Context-Consumers

In our framework, context-aware services (CAS) are the consumers of context information obtained from context-providers. A CAS can be implemented as a Web service that can understand situational context and can adapt its behavior according the changing conditions as context data may change rapidly. It produces dynamic results according to the 5 WH questions: who, where, when, what, and why it was invoked. A CAS can be responsive to various situational conditions, such as:

- The location of the client.
- The time at which the client invokes the service.
- The activity that the client is carrying out at the time it invokes the service.
- The preferences that the client may have defined prior to invoking the service.
- The security and privacy policies associated with the client of this service.
- The device (laptop, tablet, smartphone, etc.) that the client is using to invoke the service.

3.2 Context Broker

The Context Broker is a mediator service that decouples context-consumers from context-providers. It is in charge of handling subscriptions of consumers in which they express their interest to consume some types of context information, and registration of context-providers that are willing to provide some types of context information. Given that context-consumers do not usually have the capabilities to

negotiate, manage, and monitor QoC, they delegate management tasks, such as context-provider selection and QoC negotiation, to the Context Broker. The Context Broker is aware of the current QoC of context-providers through the Notification Manager that implements a topic-based publish-subscribe system.

The Context Broker includes several components that cooperate in order to deliver personalized services to their clients. These components are the *Context Request Dispatcher* (*CRD*), the *CLA Manager*, the *QoC Information Manager*, the *Notification Manager*, the *Profile Manager*, and the *Policy Manager*. These components are under the control of the *Coordinator* component. They allow carrying out various management operations such as QoC-based context-provider selection, CLA negotiation, user profile management, and policies management. The back-end databases maintain information about context-providers' policies, customers' profiles and preferences, CLAs, and dynamic QoC information.

The CRD is in charge of implementing different policies for the selection of context-providers, based on the context-consumer's requirements in terms of context information and QoC, and the context-providers' QoC offerings. In our previous work [22], we described an algorithm for context-provider selection that allows ranking potential providers according to their level of satisfaction of the client's QoC requirements. The CLA Manager is in charge of carrying out the negotiation process between a context-consumer and a context-provider in order to reach an agreement as to the service terms and conditions. The negotiation protocol is described in Sect. 4. The Profile Manager is responsible for managing context-consumers' profiles, including their preferences in terms of context information and required QoC. The Policy Manager is in charge of managing rules and policies such as authorization policies and QoC-aware selection policies of context-providers.

The Notification Manager implements a topic-based publish-subscribe system in which context-providers are the publishers and context-consumers are the subscribers. Figure 2 depicts this model. QoC offerings of context information, requested by clients, represent the topics of the system. If there is a significant change in the current QoC of particular context information, the context-provider notifies the Notification Manager about the change in its QoC offering. Then, the Notification Manager notifies any subscriber to the corresponding QoC offering of that change. In addition to this model for getting QoC updates, the QoC Notification Manager also implements a regular on-demand request/response model.

3.3 Context-Providers

Context-providers expose interfaces that allow context-consumers to get context information, and enable the context broker to negotiate CLAs on behalf of context-consumers. They can offer several types of context information (location, temperature, user-activity, etc.). Figure 3 depicts a typical architecture of the context-provider platform.

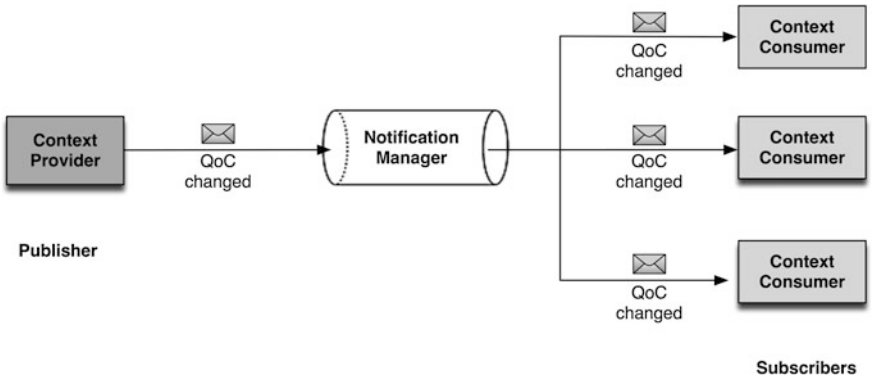


Fig. 2 Topic-based publish/subscribe system

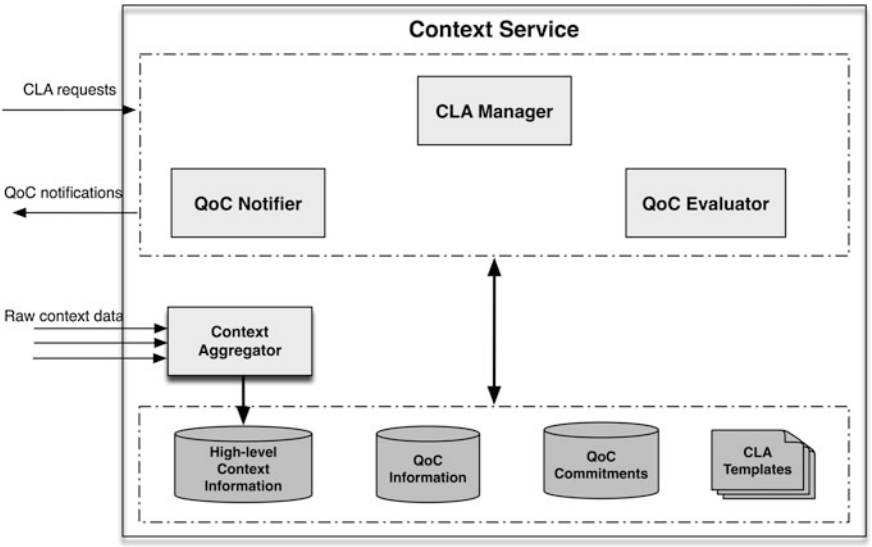


Fig. 3 Context-provider architecture

Raw context data collected from various context sources is aggregated by the Context Aggregator to determine high-level context information that is stored in a database. It is then used by the QoC Evaluator to estimate the current QoC offerings. QoC Information, which is also stored in a database, is made available to the CLA Manager and the QoC Notifier. The CLA Manager component is responsible for negotiating with the Context Broker, or directly with context-consumers, context-level agreements that specify the context information and the QoC level to be delivered. The QoC Notifier is in charge of notifying the Notification Manager of substantial changes in the QoC offering of a given context information.

4 CLA Negotiation Process

Once a context-consumer needs particular context information, it sends a CLA Request to the Context Broker. This latter tries, then, to find a suitable context-provider that can meet the consumer's context requirements. It subsequently handles the CLA negotiation process that we describe in this section. Figure 4 depicts a scenario of the CLA negotiation process between a context-consumer, the Context Broker components, and a selected potential context-provider in order to reach an agreement for context provisioning. The negotiation steps are as follows:

Step 1: The context-consumer submits A CLA Request to the Context Broker to find out an appropriate context-provider that can meet its requirements in terms of context information and QoC.

Step 2: After processing the consumer's authentication, the Coordinator requests its profile from the Profile Manager. Then, it requests from the Context Request Dispatcher (CRD) to find out suitable context-providers that can deliver context information according to the consumer's requirements.

Step 3: The Coordinator requests policies of the selected context-providers from the Policy Manager.

Step 4: If the consumer's profile is available in the profile repository, because it had previously used some services of the Context Broker, the Coordinator may determine whether the context-providers, found by the CRD, can handle or not the consumer's request. This decision relies on the profile of the consumer and the policies of selected context-providers.

Step 5: If the consumer's profile is not available in the profile repository, then the Coordinator asks the consumer to provide information, such as context needs and required levels of QoC, in order to create a new profile for the consumer.

Step 6: If there is at least one context-provider that can meet the context-consumer requirements, the Coordinator requests from its CLA Manager to negotiate with the context-provider the terms and conditions of context delivery.

Step 7: The CLA Manager forwards the CLA Request to the CLA Manager of the context-provider requesting a CLA proposal. The context-provider parses the CLA Request and validates it against its CLA templates.

Step 8: If the CLA Request is acceptable to the context-provider, then its CLA Manager responds to the CLA Request by sending back a CLA proposal to the Context Broker. The Context Broker analyzes it to find out if it responds or not to all the consumer's context and QoC requirements.

Step 9: If all the consumer's expectations can be met, then the Context Broker accepts the context-provider offer and sends A CLA Confirmation to the CLA Manager of the context-provider. Otherwise, it rejects the proposal and makes a counter-proposal with different conditions, terms, costs, etc.

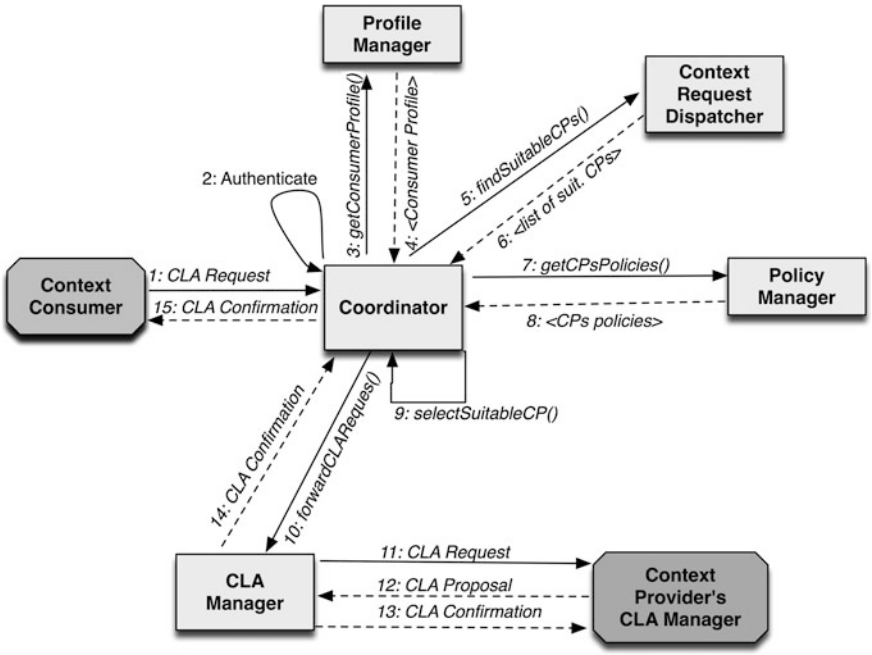


Fig. 4 CLA negotiation process

Step 10: In the case of the CLA Confirmation, the two parties approve the agreement. The context-provider can, then, start delivering context information to the context-consumer in accordance with the terms of the agreement.

At the steps 7, 8, and 9 of this process, a multi-attributes negotiation takes place between the CLA Managers of both parties. Context-consumers may have their own preferences in terms of QoC for each category of context information they would like to obtain. Similarly, context-providers may have different QoC offerings for each category of context information they are providing. In our earlier work [12], we specified an algorithm for the selection of suitable context-providers that are capable of providing context information needed by the context-consumer. The algorithm uses an auction-based approach and ranks potential context-providers using similarity ranking based on the Euclidian distance of each offer from the minimal requirements of the context-consumer. The selection algorithm does not guarantee that the offer of a selected context-provider is the best one. Therefore, to reach a CLA, the context-consumer and the selected context-provider need to negotiate CLOs for a given category of context information using a multi-attributes negotiation model.

Multi-attributes negotiation is a process in which multiple issues have to be negotiated concurrently by two parties. To reach an agreement, The CLA Managers of both parties have to go through several rounds of offers and counter-offers until they reach an agreement or reach a predefined maximum number of rounds. In each round, the Broker's CLA Manager evaluates an aggregate utility function, on behalf of the context-consumer, and determines if the offer of CP is acceptable or not. Likewise, the context-provider's CLA Manager evaluates the utility of the context-provider.

5 Scenarios of Interactions

Figure 5 shows the interactions among the components of the framework. The Notification Manager acts as an intermediary between publishers (context-providers) and subscribers (context-consumer) on a collection of context information (QoC offerings). Besides CLA negotiation, we distinguish three other kinds of interactions: registration with the Notification Manager, notification of QoC offering change, and request of current QoC offering.

Registration A context-consumer invokes the *registerSubscriber*() method of the Notification Manager to register its interest in receiving updates on QoC offerings of some types of context information. Similarly, a context-provider invokes *registerPublisher*() of the Notification Manager to register its interest to publish QoC offering of some types of context information through the Notification Manager.

Notification of QoC offering change The Notification Manager receives notifications on QoC offering change through its qocNotifyNM() method that a context-provider invokes. It, then, notifies concerned subscribers (context-consumers) about that change by invoking their qocNotifyCC() method.

Request of current QoC offering A context-consumer may request the current value for a given context information by invoking *getCurrentQocOfferingNM*() of the Notification Manager, which forwards the request to context-providers that are providing that context information requested by the context-consumer. The Notification Manager has also two private additional methods *findRegisteredSubscribers*() and *findRegisteredPublishers*(). The first method is invoked to get the list of context-consumers, which have subscribed to a given context information. The Notification Manager calls this method once it has received a notification of QoC offering change for that context information. The second method is invoked to get the list of context-providers that are publishing the context information requested by a context-consumer that has invoked *getCurrentQocOfferingNM*().

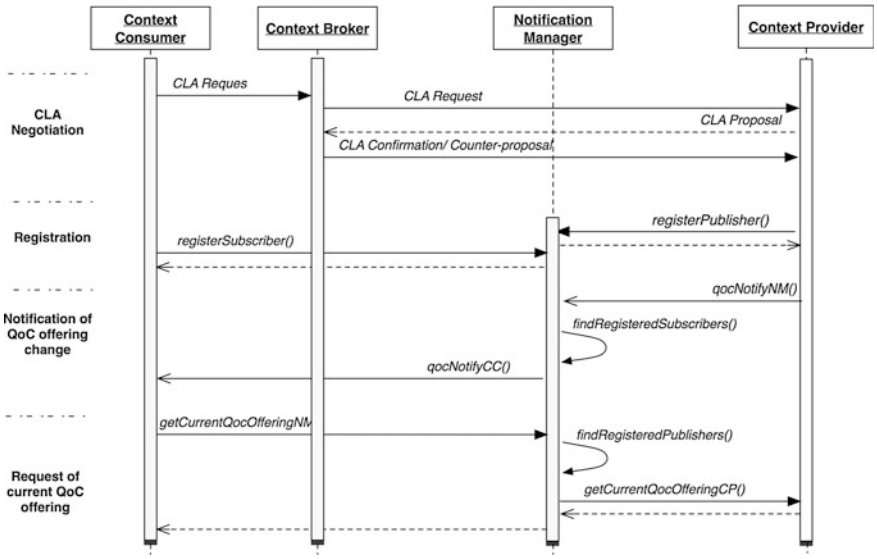


Fig. 5 Diagram of interactions among the framework components

6 Conclusion and Future Work

As a result of the emergent demand for context-aware services, context-providers are increasingly using SOA and the Web services technology to implement services that can ensure several QoC levels. In this paper, we have presented a framework for CLA-based context provisioning. The framework relies on Context Brokers, to mediate between context-consumers and context-providers, and a Notification Manager, to handle the notifications on the changes in the QoC offerings of context-providers, using a publish/subscribe model. We have described the model of interactions among the components of the framework and the CLA negotiation protocol. Selection of target context-providers by Context Brokers relies on our selection algorithm described in our previous work [22].

As a future work, we are planning to consider applying advanced algorithms for the multi-attribute negotiation process, investigate the issue of a common ontology-based model for QoC representation that all components of the framework can use; and then, describe the mappings from the various QoC representation models described in the literature to that common model. Moreover, we intend to build a prototype of the framework together with some real scenarios for QoC-aware context provisioning.

References

1. Dey, A.K.: Understanding and using context. J. Pervasive Ubiquit. Comput. **5**(1), 4–7 (2001)
2. Schilit, W.N.: A system architecture for context-aware mobile computing. PhD thesis, Columbia University (1995)
3. Sheikh, K., Wegdam, M., van Sinderen, M.: Middleware support for quality of context in pervasive context-aware systems. In: Fifth Annual IEEE International Conference on Pervasive Computing and Communications, Workshops, pp. 461–466 (2007)
4. Le Sommer, N., Guidec, F. and Roussain, H.: A context-aware middleware platform for autonomous application services in dynamic wireless networks. In: The First International Conference on Integrated Internet Ad-hoc and Sensor Networks (InterSense '06) (2006)
5. Santos, B.S., Van Wijnen, R.P., Vink, P.: A service-oriented middleware for context-aware applications. In: The 5th International Workshop on Middleware for Pervasive and Ad-hoc Computing (MPAC 2007), pp. 37–42, Newport Beach, CA, USA (2007)
6. Riva, O., di Flora, C.: Contory: a smart phone middleware supporting multiple context provisioning strategies. In: The 26th IEEE International Conference on Distributed Computing Systems (ICDCS 2006) workshops, p. 68 (2006)
7. Pinto, R.P., Cardozo, E., Guimaraes, E.G.: A component framework for context-awareness. In: The International Wireless Communications and Mobile Computing Conference, IWCMC '08, pp. 315–320 (2008)
8. Baldauf, M., Dustdar, S., Rosenberg, F.: A survey on context-aware systems. Int. J. Ad Hoc Ubiquit. Comput. **2**(4), 263–277 (2007)
9. Henricksen, K., Indulska, J., McFadden, T., Balasubramaniam, S.: Middleware for distributed context-aware systems. In: OTM Confederated International Conferences, pp. 846–863, Springer, Berlin (2005)
10. Truong, H.L., Dustdar, S.: A survey on context-aware web service systems. Int. J. Web Inf. Syst. **5**(1), 5–31 (2009). (Emerald)
11. Buchholz, T., Kpper, A., Schiffers, M.: Quality of context: what it is and why we need it? In: The 10th International Workshop of the HP OpenView University association (HPOVUA) (2003)
12. Sheikh, K., Wegdam, M., Van Sinderen, M.: Quality-of-context and its use for protecting privacy in context aware systems. J. Soft. **3**(3), 83–93 (2008)
13. Van Sinderen, M., et al.: Supporting context-aware mobile applications: an infrastructure approach. Commun. Mag., IEEE **44**(9), 96–104 (2006)
14. Toninelli, A., Corradi, A., Montanari, R.: A quality of context-aware approach to access control in pervasive environments. MobileWireless Middleware, Operating Systems, and Applications, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, Vol. 7, pp. 236–251. Springer, Berlin (2009)
15. Krause, M., Hochstatter, I.: Challenges in modelling and using quality of context (QoC), Mobility Aware Technologies and Applications. Lecture Notes in Computer Science, vol. 3744, pp. 324–333 (2005)
16. Filho, J.B., Miron, A.D., Satoh, I.: Modeling and measuring quality of context information in pervasive environments. In: The IEEE International Conference on Advanced Information Networking and Applications, pp. 690–697 (2010)
17. Manzoor, A., Truong, H., Dustdar, S.: On the evaluation of quality of context. In: The 3rd European Conference on Smart Sensing and Context (EuroSSC '08) (2008)
18. Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., Riboni, D.: A survey of context modelling and reasoning techniques. Pervasive Mob. Comput. **6**(2), 161–180 (2010)
19. Schmidt, H., Flerlage, F., Hauck, F.J.: A generic context service for ubiquitous environments. In: The IEEE International Conference on Pervasive Computing and Communications (PERCOM), pp. 1–6 (2009)

20. Coronato, A., De Pietro, G., Esposito, M.: A semantic context service for smart offices. In: The International Conference on Hybrid Information Technology, vol. 02, pp. 391–399 (2006)
21. Khedr, M., Karmouch, A.: Negotiating context information in context-aware systems. Intell. Syst., IEEE **19**(6), 21–29 (2004)
22. Badidi, E., Esmahi, L.: A scalable framework for provisioning context-aware application services with high-quality context information. Int. J. ACM Jordan **1**(3), 86–97 (2010)

Studying Informational Sensitivity of Computer Algorithms

Anastasia Kiktenko, Mikhail Lunkovskiy and Konstantin Nikiforov

Abstract This study is focused on informational sensitivity of an algorithm, defined as impact of different fixed-length inputs on the value of the algorithm's complexity function. In addition to classic worst-case complexity this characteristic provides a supplementary tool for more detailed and more "real world" approach to studying algorithms. Statistical measure of informational sensitivity is calculated based on statistical analysis of results obtained from multiple runs of the same program implementation of the algorithm in question with random inputs. This theory is illustrated by an example of algorithm that solves the travelling salesman problem by branch and bound method using the concorde package. For a sample of different input graphs with $1{,}000 \div 10{,}000$ vertices the statistical measurements of informational sensitivity were found and confidence ranges for complexity function were constructed. It was proven that this particular algorithm is highly sensitive to fixed-size inputs by complexity function.

Keywords Informational sensitivity · Resource effectiveness · Algorithm complexity · Travelling salesman problem · TSP

1 Introduction

Informational sensitivity of an algorithm by complexity is defined as impact of different fixed-length inputs on the value of the algorithm's complexity function. In general, in automatic control theory the impact of changes in input on the output characteristics of some object is traditionally referred to as informational sensitivity

A. Kiktenko (✉) · M. Lunkovskiy · K. Nikiforov
Faculty of Applied Mathematics and Control Processes, Saint Petersburg State University, Universitetskii prospekt 35, 198504 Saint-Petersburg, Russia
e-mail: knikiforov@apmath.spbu.ru
URL: http://www.apmath.spbu.ru/en/

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_35,

by some input parameter. This definition was originally introduced into computer algorithm studying in [1]. In given case the object of study is a computer algorithm, and its input is an informational array of data containing the input of this algorithm, i.e. we are talking about informational sensitivity of a computer algorithm.

The approach to studying of informational sensitivity of algorithms in this research is characterized by:

1. Use of random data that allows algorithm complexity to be regarded as a finite discrete random value;
2. Use of CPU time of execution for assessment of algorithm complexity function that defines number of base operations the algorithm needs to retire in given computational model.

In this study this random value was assessed via statistical methods by a numerical experiment employing parallel computations. As a result, we've obtained data on informational sensitivity of branch and bound method for the travelling salesman problem.

In general assessing the informational sensitivity gives as an additional tool for detailed algorithm analysis. For example it can be used for more well-founded solution of algorithm optimization problems based on complex criterion of resource efficiency [1]. If the computing system in question requires constant execution time on different fixed-length inputs, it is important to pick an algorithm with the least informational sensitivity.

Before formally stating the problem it is necessary to formally define the notions mentioned here.

2 Main Definitions

2.1 Complexity Function

Complexity of an algorithm A with input D [1] is defined as the number $f_A(D)$ of base operations in given computational model retired by algorithm with this input.

Exploring upper and lower bounds and average values of complexity with different fixed-length inputs it is possible to define complexity function that is only dependent on the input length:

$$f_A^{\wedge}(n)\ in\ worst{-}case\ scenario: f_A^{\wedge}(n) = \max_{D \in D_n}\{f_A(D)\}, \tag{1}$$

$$f_A^{\vee}(n)\ in\ best{-}case\ scenario: f_A^{\vee}(n) = \min_{D \in D_n}\{f_A(D)\}, \tag{2}$$

$$\overline{f_A}(n)\ in\ average\ scenario: \overline{f_A}(n) = \sum_{D \in D_n} P(D) f_A(D), \tag{3}$$

where $P(D)$ is the probability of input D for given usecase of the algorithm and D_n is the set of problems of size n.

We can also view the algorithm complexity with fixed-length input as a finite discrete random value that has some unknown distribution.

2.2 Informational Sensitivity

Quantitative value of informational sensitivity of an algorithm must be realistically computable with enough empirical data and be universal, i.e. independent of particular field where the algorithm might be applied. It must have an unified measurement and be comparable.

One value that has all the required features [2] is the statistical measure $\delta_{IS}(n)$ of informational sensitivity by complexity (with a fixed length input), that is a product of general variance coefficient of complexity function (as a finite discrete random value) and a normalized range of variance of complexity values:

$$\delta_{IS}(n) = V(n)R_N(n). \tag{4}$$

The variance coefficient V is a standard characteristic point of the sample dependent on size of n that is computed as

$$V(n) = \frac{\sigma_{f_A(n)}}{\overline{f_A}(n)}, \tag{5}$$

where $\sigma_{f_A}(n)$ is the standard deviation of complexity (as a finite discrete random value) with inputs of fixed length n and $\overline{f_A}(n)$ is a general mean value of complexity. The normalized range $R_N(n)$ for complexity values with input size n is calculated as half of variance interval divided by its median value:

$$R_N(n) = \frac{f_A^{\wedge}(n) - f_A^{\vee}(n)}{f_A^{\wedge}(n) + f_A^{\vee}(n)}. \tag{6}$$

The reasoning behind defining informational sensitivity this way is the following. With input of a fixed length n we see the random value of complexity as a random function of non-random value n. The classic point value for dispersion of a random value is the value of standard deviation. But measurement of informational sensitivity must also take into account length of the segment of all possible values of complexity function. This is necessary because, given equal dispersion values, the algorithm with larger segment of values will render greater informational sensitivity. To account for this the variance range value can be employed. This value is dependent on input length and is thus a function of n. It is defined as difference of values of complexity function in worst- and best-case scenarios: $f_A^{\wedge}(n) - f_A^{\vee}(n)$. Let's also note that maximum value of standard deviation of

complexity function as a finite discrete random value is half of variance range. We use the normalized (relative) range $R_N(n)$ of variance of complexity function for inputs with size n as half of variance interval divided by its median (6). As all the values of complexity function are positive and worst-case complexity is greater than or equal to the best-case: $f_A^{\wedge}(n) \geq f_A^{\vee}(n)$, the value of $R_N(n)$ is normalized:

$$0 \leq R_N(n) < 1.$$

Thus if we take the median of variance interval of statistical population as the measurement of mean value, we can expect that in worst-case scenario the complexity can be no more than $(1 + R_N(n))$ times larger. The value $R_N(n) = 0$ corresponds to situation $f_A^{\wedge}(n) - f_A^{\vee}(n) = 0$, i.e. the algorithm has zero informational sensitivity and its complexity depends only on the input size and not on any features of input data.

Let's also note that in worst-case scenario of complexity dispersion the general variance coefficient is equal to normalized variance range according to (6) and (5).

The measure $\delta_{IS}(n)$ of informational sensitivity of an algorithm with fixed-length input uses statistically precise values and takes into account both standard deviation of complexity values and length of the segment of possible values relative to a normalized unit. As such it is called a statistical value [2]. Let's note that a statistical measurement $\delta_{IS}(n)$ does not require any knowledge of particular distribution law of values of complexity and works with point measurements that are obtained by experiment. The problem of finding the normalized variance range can be solved by theoretical assessment of the algorithm in question and/or based on experimental data (i.e. minimum and maximum values of complexity in a given sample).

There are two principal approaches to finding $\delta_{IS}(n)$: theoretical and experimental. While using the theoretical approach it is necessary to obtain complexity functions for best-case, worst-case and average scenarios and a theoretical value of standard deviation as a function of input size. With this, $\delta_{IS}(n)$ is explicitly obtainable. We must note thus the obtained theoretical relations have to take into account the particular features of input in given usecase and circumstances.

The experimental approach employs methods of mathematical statistics and is based on forming a sample of complexity values that can be used to get necessary values for calculating $\delta_{IS}(n)$. Based on a series of tests with different fixed-size inputs it is necessary to find: sample median, sample dispersion, sample variance coefficient, minimum and maximum values in the sample. With this data, $\delta_{IS}(n)$ can be calculated as statistical measure of informational sensitivity by complexity.

It is also important to note that the sample must be representative relative to a set of base values corresponding to usecases of given algorithm in given computational system. It is this data that forms statistical population. Thus an individual experiment is finding a value of complexity function for given implementation of the algorithm with a particular input, i.e. in base operations of given computational model.

In this study we employ a simplified approach, measuring the complexity function in CPU time units of execution of a particular implementation of given algorithm. As, strictly speaking, units of complexity function are base operations of computational model where the algorithm is formally defined, computer-based assessment of the algorithm's program implementation will be closer to analysis of the algorithm itself the closer this chosen model is to a real-life computer (e.g. D. Knuth's MIX-machine [3]). We do, however, assume existence and use of such a computational model in which CPU time is the same as value of complexity function in units of this computational model. Existence of such model is not proven here, it is assumed as given. This approach can be further clarified by using CPU event profiling in addition to CPU time and by relating the profiling events to operations of a computational model.

The statistical measure of informational sensitivity $\delta_{IS}(n)$ is a combined one, so, in accordance with (4), algorithms with big variance range but small dispersion and algorithms with small variance range but big dispersion can end up producing the same values. This approach is rational, as in either case the probabilistic dispersion of expected relative changes in a complexity function would be roughly the same. Let's also note that the values $\delta_{IS}(n)$ of the same algorithm can vary significantly with increase of input size, both towards increase and decrease, as shown by a series of experiments [2].

In this study we've conducted a numerical experiment with an algorithm for solving the travelling salesman problem on a set of graphs with 1,000 ÷ 10,000 vertices (in modern applications graphs with 10,000 vertices or more are not uncommon) [4, 5].

The main goal of this example is assessment of informational sensitivity and confidence intervals of complexity for an algorithm solving the travelling salesman problem via branch and bound method.

As a result of the study, there were obtained point estimates for complexity function with confidence ranges, distribution histograms were plotted and the informational sensitivity of the algorithm was quantified. Practical steps of this application of the theory listed here is described in the next section.

3 Program Implementation

3.1 Concorde

Software implementations aimed on solving the travelling salesman problem during the last 30 years developed from problems of size 100–10,000 and more [4].

The concorde toolchain [6], developed in Princeton since 2001, is geared towards solving the travelling salesman problem by branch and bound method. As of 2013 concorde is arguably the best program implementation of branch and

bound method for symmetrical arrangements and can be used to solve symmetrical problems of all classes with size varying from 1 to dozens of thousands vertices.

Unlike, for example, local optimization methods, code for the branch and bound method is usually split into two blocks: linear programming library and the algorithm implementation. As a linear programming library in this study the QSopt package was used. It is developed by the authors of concorde and was chosen mostly because it is available for unrestricted educational and research use, although concorde can also use other libraries including commercial ones.

Same as concorde, QSopt is written in ANSI C and works on Windows and Linux/Unix platforms. It has two GUI frontends that allow the users to enter, edit and solve linear programming problems interactively. The problems can be saved to files in special format (LP or MPS) [6].

For this particular study concorde was installed onto several multicore computing systems including two computing clusters, running various Linux systems. The input files in TSPLIB [6] format were generated with portmgen commands [7].

3.2 Input Data Formats

The goal of this study requires building a significantly large sample from execution times of the program (implementing the algorithm) on varied random inputs.

Open experimental study of various algorithms for solving the travelling salesman problem are carried out for more than 10 years in DIMACS [7]. Random graph generator from their site was used for this study. Non-Euclidean (non-geometrical) graphs with weight matrices filled with random numbers were used for experiments. Number and sizes of random graphs used for study are shown in (Table 1).

3.3 Computation Methods

There are two possible ways of gathering statistics:

- Waiting for solution process of each random graph to stop and forming a sample of lengths of solution processes
- Stopping the solution after certain amount of time and counting the number of problems whose solutions managed to halt.

For this study we've used the first approach.

To gather necessary amount of data the program implementation of the algorithm had to be run on random data of given type. CPU times were recorded as one

Table 1 Sample inputs number

Size	1,000	2,500	4,000	5,500	7,000	10,000
Number	34,967	4,145	3,543	2,912	1,752	2,540

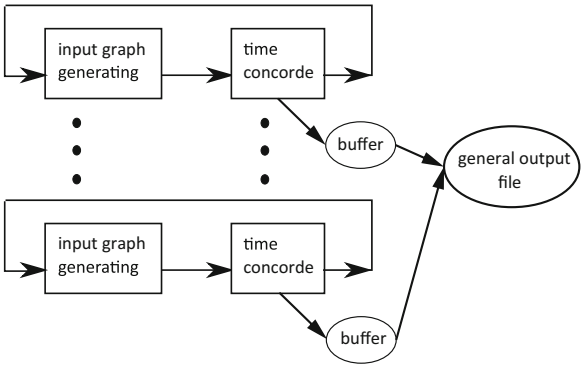


Fig. 1 Computational flow scheme

of characteristics of the algorithm's complexity. Use of parallel computation helped to speed up the process of data collection.

Parallelism of calculations is implicit: given number of computing cores we run the same number of background concorde processes and rely on the automatic balancing of the OS that would occupy ~ 100 % CPU time of a given core. The execution time is measured with GNU time command.

One of the most important features of GNU time in given context is that it can output the "pure" user time of a process that's independent of OS kernel that maintains the process.

In numerical calculations several instances of concorde were run in parallel. In each of the parallel processes the concorde code after the generation a graph of random size and type was run sequentially multiple times. IO streams of the processes were stored in buffers and unloaded into general output file as responding processes terminated (Fig. 1).

The computations were performed on the computing of Faculty of Applied Mathematics and Control Processes (AM-CP) of St. Petersburg State University (hpc.apmath.spbu.ru) running OS SuSe 11.

The same program was installed on an 8-core virtual machine (SuSe 11) of the SPbU Resource Center and on a T-platform cluster of the same Center running CentOS 5.

4 Numerical Results

For obtaining experimental data it is necessary to know the representative sample size [2]. For the size $n > 50$ and the confidence probability $\gamma = 0.95$ we can use the formula [2]

$$n^* = \frac{3,8416\, V^2}{\varepsilon^2}. \tag{7}$$

First, the value of V is assessed based on a preliminary experiment, i.e. with a sample of a priori defined size. Then we calculate $n^*_{(1)}$ and the algorithm is run a corresponding number of times. Then we can obtain a new value of V calculate $n^*_{(2)}$. If $n^*_{(2)} < n^*_{(1)}$, the experiment is halted and the sample size is defined as $n^*_{(2)}$, otherwise the iteration continues until halting condition is reached.

Then the following characteristics are defined:

- sample median $\overline{f_v} = \frac{1}{m}\sum_{i=1}^{m} f_i$;
- sample dispersion $s^2 = \frac{1}{m-1}\sum_{i=1}^{m} (f_i - \overline{f_v})^2$;
- standard deviation $\sigma = \sqrt{s^2}$;
- sample variance coefficient $V = \frac{\sigma}{\overline{f_v}}$;
- minimum and maximum element.

Sample size and its characteristic points are shown in Table 2 with non-geometric problems of sizes 1,000, 2,500, 4,000, 5,500, 7,000 and 10,000 vertices.

For obtaining probable values in the sample of complexity function random values one needs to construct confidence intervals. Half-range is calculated as:

$$\delta = t(0,05 : n^* - 2)\sqrt{\frac{s^2}{n^*}}, \tag{8}$$

where $t(0.05: n^* - 2)$ is a quantile of Student distribution with probability 0,05 and $n^* - 2$ degrees of freedom. Knowing the half-range of confidence range its bounds are calculated: left bound—$\overline{f_v} - \delta$; right bound—$\overline{f_v} + \delta$.

Plot 4 shows the relation of the sample median and confidence intervals to the input size.

For visualisation of obtained samples corresponding to fixed-size inputs the statistical frequency histograms were used. They were constructed based of same-frequency intervals. This approach was dictated by the fact that the complexity function in given case varies greatly with input data. Figure 2 shows the histograms for non-geometrical problems of size $1,000 \div 10,000$. The relation between informational sensitivity and input size is given below (Figs. 3 and 4).

Table 2 Representative sample inputs

Size	n^*	$\overline{f_v}$	σ	V_f	Max	Min
1,000	147	65.19	51.61	0.79	315.18	26.10
2,500	257	315.61	319.35	1.01	2833.41	68.9
4,000	440	1028.96	1315.60	1.28	12227.74	114.21
5,500	527	3055.37	4468.46	1.46	55530.2	161.89
7,000	652	5554.60	6665.58	1.2	44694.17	136.74
10,000	561	16238.91	21788.82	1.33	155065.47	344.27

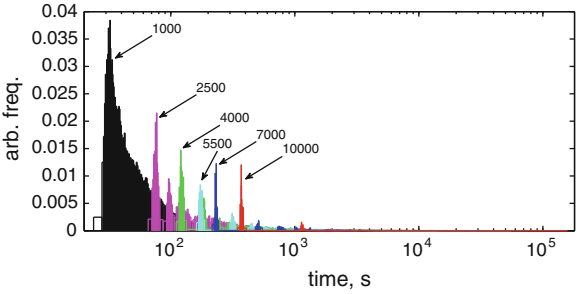


Fig. 2 Normalized frequency histograms for random input data of size $1{,}000 \div 10{,}000$

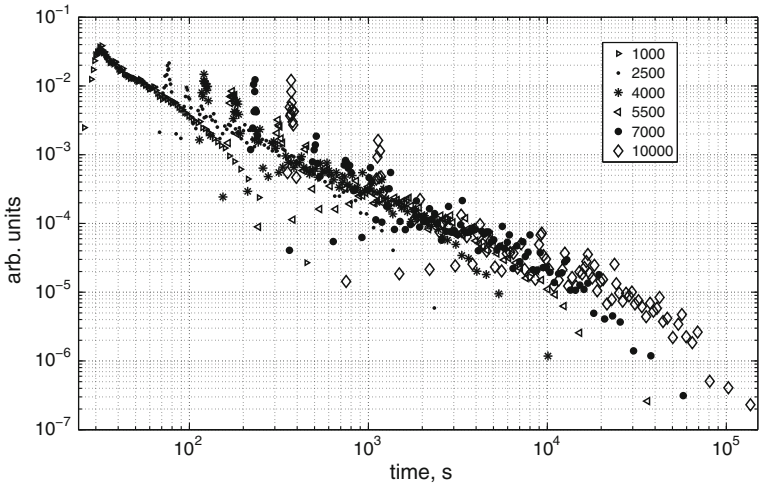


Fig. 3 Relation in logarithmic scale between fixed-size random input and time to algorithm halting. Markers are placed at mead-points of histogram bars from Fig. 2

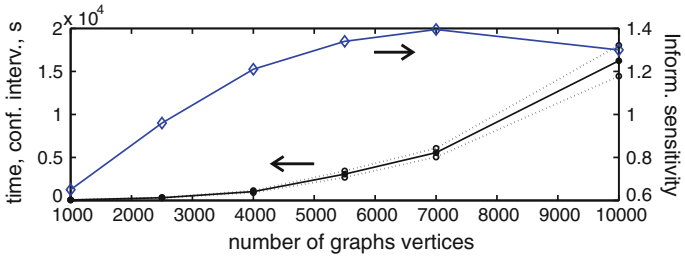


Fig. 4 Confidence intervals of complexity function and statistical measure of information sensitivity

5 Conclusion

This study managed to achieve the following results. Algorithmic complexity was assessed statistically as a random value, based on a numerical experiment employing parallel computation. For this the concorde package was installed on the AM-CP HPC complex and T-platform cluster of SpBU Resource Center. For the segment of inputs with 1,000 ÷ 10,000 vertices the characteristic points were obtained and analysed relative to input types, and frequency histograms were constructed for each of them. Statistical measure of informational sensitivity with fixed-length input was assessed.

The most important results are:

- Function of informational sensitivity was measured for program implementation of branch and bound method (concorde) given input graphs with number of vertices between 1,000 and 10,000 and uniformly filled random weight matrices.
- Median execution times for the same input type and their confidence ranges were obtained.
- The studied algorithm and its software implementation are found to be sensitive by complexity to the input data of fixed length.

Acknowledgments The authors acknowledge Saint-Petersburg State University for a research grant 9.38.673.2013. Research was carried out using computational resources provided by Resource Center "Computer Center of SPbU" (http://cc.spbu.ru).

References

1. Ul'anov, M.: Resource-effective computer algorithms. Development and analysis (in Russian). Fizmatlit, Moscow (2008)
2. Petrushin, V., Ul'anov, M.: Informational sensitivity of computer algorithms (in Russian). Fizmatlit, Moscow (2012)
3. Knuth, D.: The Art of Computer Programming, Fundamental Algorithms, vol. 1. Addison-Wesley, Massachusetts (1997)

4. Gutin, G., Punnen, A.: The traveling salesman problem and its variations. Kluwer Academic Publishers, Dordrecht (2004)
5. Appelgate, D., Bixby, R., Chvatal, V., Cook, W.: The Traveling Salesman Problem: A Computational Study. Princeton University Press, Princeton (2006)
6. Concorde TSP Solver, http://www.tsp.gatech.edu/concorde.html
7. 8th DIMACS Implementation Challenge: The Traveling Salesman Problem, http://dimacs.rutgers.edu/Challenges/TSP/

Binary Matchmaking for Inter-Grid Job Scheduling

Abdulrahman Azab

Abstract Inter-Grid is a composition of small interconnected Grid domains; each has its own local broker. The main question is how to implement cross-Grid job scheduling achieving stability and load balancing, together with maintaining the local policies of interconnected Grid. Existing Inter-Grid methodologies are based on either centralised meta-scheduling or decentralised scheduling which carried out by local brokers, but without proper coordination. The question is how to perform matchmaking between a particular local job and the workers of a remote domain. Performing matchmaking remotely would result in computational overhead in case of many domains asking for match from one domain. Performing matchmaking locally requires transmission of the resource information set of the remote domain, which would result in high data traffic. This position paper introduces a coordinated scheduling technique for broker based inter-Grid architectures. Resource information set of each Grid domain is stored in a binary form. Matchmaking is carried out in the local domain using fast logical operations. Our primary results show that the proposed technique achieves 26 speedup in the matchmaking process compared to Condor negotiator, and a reduction up to 99.92 % in the resource information size compared to Condor ClassAd.

Keywords Grid · Job scheduling · Efficiency evaluation · Condor negotiator

A. Azab (✉)
Department of Computer and Systems Engineering, Faculty of Engineering, Mansoura University, Mansoura, Egypt
e-mail: abdulrahman.azab@mans.edu.eg; abdulrahman.azab@ux.uis.no

A. Azab
Department of Electrical Engineering and Computer Science, Faculty of Science and Technology, University of Stavanger, Stavanger, Norway

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_36,

1 Introduction

Grid computing provides the infrastructure for aggregating different types of resources (e.g. desktops, mainframes, storage servers) for solving intensive problems in different scientific and industrial fields, e.g. DNA analysis, weather forecasting, modelling and simulation of geological phenomenon [1]. Computational grid Model is mainly composed of three components: (1) client/user, which consumes grid resources by submitting computational jobs, (2) resource broker/scheduler, which is responsible of allocating submitted jobs to matching workers, and (3) worker/executor, where jobs are executed [2]. Due to the rapid increase in the demand for compute resources as a result of building more and more compute intensive applications, grid had to scale to include more workers in order to fulfil those demands. One possible solution is to establish an interconnection between existing Grid domains, which is known as *Inter-Grid*. The concept of interconnecting Grid domains was first introduced by the condor project in 1992, as *flocking* [3]. The idea was to setup a *gateway* machine in each condor pool, i.e. Grid domain, for managing job migration between domains. The main drawback is that there was no coordination between those gateways. Other grid systems (e.g. gLite [4], Condor-G [5], Unicore [6], Nimrod-G [7]) used the concept of meta-scheduling [8], known also as super-scheduling. Meta-schedulers work in a layer above traditional brokers so that instead of submitting to their local brokers, users submit their demands to a meta-scheduler which transfers the submissions to a broker on a grid domain which can fulfil the demand. The problem with this approach, is the lack of coordination between meta-schedulers. Another approach, *broker overlay*, is to establish the interconnection through an overlay network between brokers [9–14]. This approach has proven to be scalable [9] but it doesn't achieve load balancing. The reason is that for each external job submission, the local broker sends a query to its neighbouring brokers looking for a match. The Process of matchmaking the job requirements with the resource information of an external domain can be carried out either by the local broker or remotely by the broker of the external domain. The problem with the first method is that the local broker will have to retrieve the resource information of each neighbour domain from its associated broker to perform the matchmaking. This would result in a considerable traffic due to the large size of the resource information set. The problem with the second method is that in case of many domains submitting jobs to one domain, this domain's broker will be computationally overloaded with many matchmaking procedures.

This position paper introduces a coordinated scheduling technique for interconnected Grid domains based on binary representation and matchmaking of resource information and job requirements. Resource information set of each Grid domain is stored as a set of binary operands, 26-Bytes operand to represent each worker. Job requirements are represented as operand of the same size. Matchmaking is carried out in the local domain using logical AND and XOR operations. Our primary results show that the binary representation makes a reduction up to

99.91 % in the resource information set size compared Condor machine ClassAd [15]. The binary matchmaker achieves 26 speedup in the matchmaking process compared to Condor negotiator.

2 Broker Architecture

The proposed scheduling technique is an update to our SLICK inter-Grid architecture [16, 17]. SLICK connects different Grid domains by connecting their local brokers in a broker overlay. The interconnection is based on structured-p2p [18] network of brokers. Each broker has a nodeID and a routing table which includes the neighbouring brokers within the overlay. SLICK broker is designed to work as a gateway on the top of the local broker of the Grid domain. Our current implementation is designed to work on Condor Grid negotiator [15]. The gateway broker has two roles: (1) pickup derelict jobs from the job queue and submit them to suitable Grid domain(s), and (2) receive job submissions from external brokers and insert them into the local broker queue. The broker architecture broker is presented in Fig. 1.

The grey circles in broker overlay represent the neighbouring brokers, i.e. those brokers which addresses are included in the routing table, while the wight circles represent the other brokers. In the architecture described in [16], there have been two job queues: the local broker queue and the gateway queue. In this design we use only the local broker job queue (JQ). Management of message exchanging between the broker and its neighbours is carried out by the communication controller. Local Resource information is periodically retrieved from the local workers, managed by the Information service (IS), and stored in the information service database (IS-db) as Condor machine ClassAds [15]. The binary information service BIS periodically converts the data stored in the IS-db to binary records and store them in the binary information service database (BIS-db). The binary representation of resource information is described in Sect. 3. The job allocation cycle is described as follows:

1. When a job j is submitted by one of the local clients, it is allocated in the job queue.
2. For each job $j \in$ JQ: The local scheduler performs matchmaking between the job requirements and the resource information in the IS-db. If match is found, then j is allocated to the matching worker(s). Otherwise, j is put at the end of the queue.
3. The *starvation detector* periodically reads the profile of each job in the queue to detect starving jobs, and label them. A job is labelled "starving" in either of two cases: (1) None of the local workers match the job requirements, or (2) There are matching workers but are fully or partially claimed. In case (1), the job is labelled as starving. In case (2), a job j is labelled as starving if [CurrentTime $-$ SubmissionTime(j)] $> \alpha$, where α is the timeout value.
4. For each starving job $j \in$ JQ: The gateway scheduler contacts the neighbouring brokers in the overlay to find a domain with matching worker(s) for j. If a

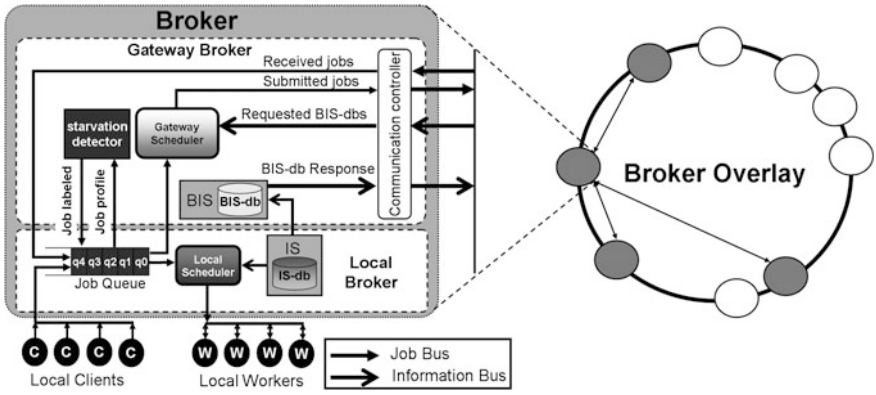


Fig. 1 Broker architecture

matching domain d is found, then j is submitted to the broker of d. Otherwise, j is put at the end of the queue. The matchmaking algorithm of the gateway scheduler is described in Sect. 4.

5. Upon receiving a job submission j from an external neighbouring broker, j is put at the end of the queue.

3 Binary Representation of Resource Information

There are different ways describing worker attributes for adopted by different Grid architectures [6, 19, 20]. One of the most popular is Condor Machine *ClassAd* [2], since it is the most descriptive and yet implemented by both Condor, gLite, and is an option in Globus [5]. The BIS converts the machine ClassAd of each worker as a 208-bit (i.e. 26-byte) worker attribute record, *WAR*, and store it in the BIS-db. The job requirements part of the job description, where the resource requirements of the target worker(s) are described, are converted to a record of the same structure, *JRAR*. The structure of WAR and JRAR is displayed in Fig. 2. Resource attributes are categorised into:

1. **Boolean** attributes, e.g. HasJava (supports java), and HasMPI (supports MPI [21] jobs), which have true/false values. The Boolean attribute part is represented as 16-bit binary value, $\boldsymbol{B}$ to store up to 16 attribute values. Condor ClassAd doesn't implement more than 10 Boolean attributes [19], but the administrator might add custom attributes, e.g. HasPython. The value, 0 or 1, of each bit in B reflects the value of the associated Boolean attribute. For example, the values of (b_5, b_6, b_7) represent (HasJava, HasMPI, HasVM) respectively.
2. **Multi-valued** attributes, e.g. Arch (Processor architecture), OpSys, and JavaVersion. The value of each of these attributes must be one of a fixed list of

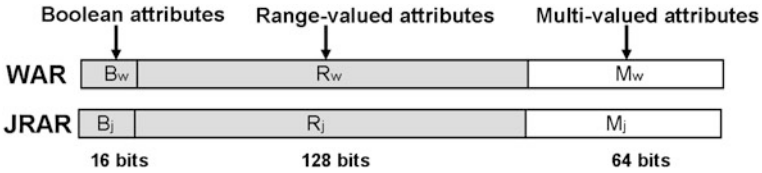


Fig. 2 Structure of the worker attribute record (WAR) and the job requirement attribute record (JRAR)

Table 1 Binary representation of three multi-valued ClassAd attributes

State		Arch		OpSys	
Unclaimed	0001b	INTEL	0001b	LINUX	0001b
Claimed	0010b	IA64	0010b	WINNT51	0010b
Owner	0011b	ALPHA	0011b	WINNT52	0011b
Matched	0100b	SGI	0100b	SOLARIS25	0100b
.....					

values supported by Condor for this particular parameter [19]. Multi-valued attribute part is represented as 64-bit, **M** to store up two 16 attribute values. Each attribute value $\{M[x] | x \in \{0, 1, \ldots, 15\}\}$ is stored in four bits to display a value from 1 to 15 (0001b to 1111b), so that there is a room for 15 options for each attribute. The zero value, 0000b, is not made available as an option in WAR. In JRAR, each attribute which has no value requirement by the job will have its associated value = 0000b. Table 1 presents examples of the binary representation of the multi-valued attributes.

3. **Range-valued** attributes, e.g. LoadAvg (load average of the CPU), Disk (available disk space), TotalCPUs, and Memory (available physical memory). Each attribute represents a numeric value which refers to a a specific machine performance metric. Range-valued part is represented as 128-bit value, ***R***, to store up to 32 attribute values. Each attribute value $\{v_x | x \in \{0, 1, \ldots, 31\}\}$ is stored in R as a 16-bit figure $R[x] = r_x r_{x+1 \ldots} r_{x+15}$. The value of each bit in $R_w[x]$ is calculated as in (1). The value of each bit in $R_j[x]$ is calculated as in (2).

$$R_w[x+i] = \begin{cases} 1 & v_x \in [I \times i, I \times (i+1)) \\ 1 & v_x > I \times (i+1) \\ 0 & v_x < I \times i \end{cases} \qquad \forall i \in \{0, 1, \ldots, 15\} \tag{1}$$

$$R_j[x+i] = \begin{cases} 1 & v_x \in [I \times (i-1), I \times (i+1)) \\ 1 & v_x > I \times (i+1) \\ 0 & v_x < I \times i \end{cases} \qquad \forall i \in \{0, 1, \ldots, 15\} \tag{2}$$

Table 2 Binary representation of multi-valued attribute using a range increment value I

Actual value	Binary representation
$v_x \in [I \times 0, I \times 1)$	$R_w[x] = 1000\ 0000\ 0000\ 0000b$
	$R_j[x] = 1100\ 0000\ 0000\ 0000b$
$v_x \in [I \times 1, I \times 2)$	$R_w[x] = 1100\ 0000\ 0000\ 0000b$
	$R_j[x] = 1110\ 0000\ 0000\ 0000b$
$v_x \in [I \times 2, I \times 3)$	$R_w[x] = 1110\ 0000\ 0000\ 0000b$
	$R_j[x] = 1111\ 0000\ 0000\ 0000b$
.....	
$v_x \geq I \times 15$	$R_w[x] = 1111\ 1111\ 1111\ 1111b$
	$R_j[x] = 1111\ 1111\ 1111\ 1111b$

where I is the range increment value. Table 2 describes the binary representation of a value v_x when it is located in different ranges. For example, the Memory attribute is represented in v_0. If the Memory value = 220 MB and the range increment value I = 64 MB, then $R[0] = 1111\ 0000\ 0000\ 0000b$ because $220 \in [64 \times 3, 64 \times 4)$. The value of I is set based on how much an increase/decrease in the value of the associated attribute would make a noticeable change in the performance.

The BIS-db, presented in Fig. 3 contains a collection of WARs, each is labelled by its associated worker id. Three values are also included: the broker id of the home broker, a time stamp which indicate the local read time of the stored worker records, and the total number of unclaimed workers.

4 Binary Scheduling Algorithm

The broker performs three types of scheduling: (1) Scheduling of locally submitted tasks to local workers, which is carried out by the *local scheduler*, e.g. condor negotiator. (2) Scheduling of externally submitted tasks to local workers by allocating them in the local job queue, and (3) Scheduling starving local tasks to external brokers, which is carried out by the *starvation detector* and the *gateway scheduler*. The gateway scheduler performs matchmaking between starving jobs in the queue and the BIS-db of each neighbouring broker. Algorithm. 1 describes the binary scheduling algorithm. The gateway scheduler collects the BIS-db from a neighbour broker and performs matchmaking. The rules for matchmaking a domain d and a job j are described as:

- If the number of available workers in the BIS-db$_d$ is less than the required number of workers by j, then j and d are unmatched.
- For each worker $w \in d$:
  - For binary attributes (B): B_w matches B_j if each *true* value condition in B_j, has the associated value in B_w *true*. This is logically represented as: $B_w \wedge B_j = B_j$. For example if has HasJava is *true* for both w and j, while HasMPI is *true* for w but is *false* for j, then B_w matches B_j.

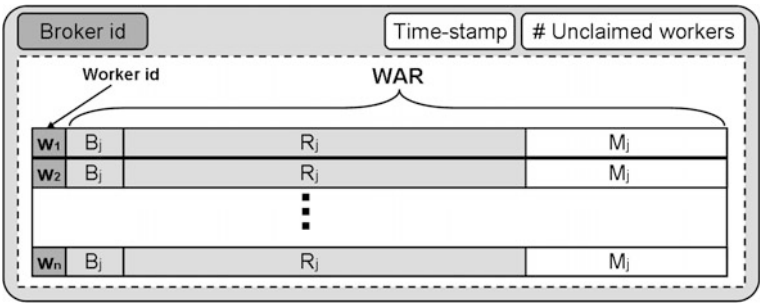


Fig. 3 The binary information set database (BIS-db)

– For multi-valued attributes (M): M_w matches M_j if each required attribute in M_j has an equal value in M_w. The equality condition between two binary figures is tested using the XOR operation. This cannot be implemented here, since the job requirements usually don't include a condition for each attribute. Attributes with no conditions are set to 0000b in M_j. The worker attribute figure M_w contains a value >0 for each attribute describing the worker specifications. To overcome this problem, we mask all non-required attributes in M_w to 0. This is carried out by generating a mask figure for M_j, MSK_j, presented in Algorithm. 2 and presented by example in Fig. 4. The logical representation for the case that M_w matches M_j is $(M_w \wedge MSK_j) \oplus M_j = 0$.

lined 1 Binary Matchmaking Scheduling Algorithm

Initialization:

`JQ` {The job queue of the local domain}

`|JQ|` {The job queue size}

N {Set of neighbouring brokers of the local broker}

Define:

`BIS-db`$_i$ {Binary information set of domain i}

B_w, R_w, M_w {Boolean, range, and multi-valued attributes of worker $w \in$ domain d}

B_j, R_j, M_j

Start:

for all $j \in$ `JQ` {Loop on each job in the queue}

  if j **is** *starving* **then** {j needs to run externally}

    $J \leftarrow j$

    EXIT Loop

for all $d \in N$ {Loop on each neighbour domain}

  $\gamma \leftarrow$ **get** `BIS-db` **from** d {get the `BIS` of domain d from its broker}

  if match(J, γ) **then** {Match exists}

    submit(J) **to** $Broker(d)$

$d \leftarrow random(N)$ {No matching neighbour domain. select a random neighbour domain}

submit(J) **to** $Broker(d)$

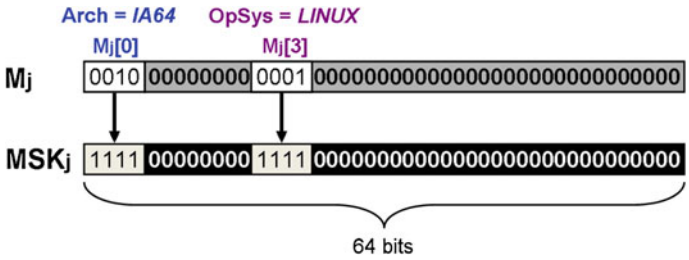


Fig. 4 Generating the mask value for a job j

lined 2 Binary Matchmaking Procedures

procedure match$(j, \texttt{BIS-db})$ {perform matching between a job j and BIS-db }
  if |BIS-db| $<$ $|j|$ **then** {available workers at BIS-db domain ¡ required workers by j}
    return false {No match}
  $MSK_J \leftarrow$ **mask**(J) {Calculate the mask for job J}
  $n \leftarrow 0$ {initiate the number of matching workers}
  for all $w \in$ BIS-db {Loop on each worker w in BIS-db }
    $BR \leftarrow B_w R_w \bigwedge B_j R_j$ {Matchmaking binary and range-valued attributes}
    $M \leftarrow (M_w \bigwedge MSK_j) \oplus M_j$ {Matchmaking multi-valued attributes}
    if $(BR \oplus B_w R_w) \vee M = 0$ **then** {attributes of w match the requirements of j}
      $n \leftarrow n + 1$ {increment the number of matching workers}
    NEXT w
  return $(\frac{n}{|j|} > 1)$ {return the number of j instances that can run on BIS-db domain workers}

procedure mask(j) {compute the 64-bits mask for the M part of the job requirements of j}
  $msk \leftarrow 0$ {initialise the mask value by zeros}
  for $i = 1$ **to** 16 {Loop on all 4-bits multi-valued attributes}
    if $M_j[i] \neq 0$ **then** {There is a required value of attribute i}
      $msk[i] \leftarrow 1111b$ {Set the associated value in the mask to 1s}
  return msk {Return the mask value of j}

– For range-valued attributes (R): R_w matches R_j if $R_w[x] \geq R_j[x] \quad \forall x \in \{0, 1, \ldots, 31\}$. This is logically represented as $R_w \wedge R_j = R_j$. For example, if the memory requirement of j is: memory $\geq$ 190 MB, the available physical memory of $w = 256$ MB, the index of the memory attribute in R is x $= 1$, and the memory increment value $I_1 = 64$ MB, then: $R_w[1] = 1111\ 1000\ 0000\ 0000b$ and $R_j[1] = 1111\ 0000\ 0000\ 0000b$. The matching condition is fulfilled for x $= 1$ since $R_w[1] \wedge R_j[1] = R_j[1]$.

- Return the percentage of matching workers to the number of required workers.

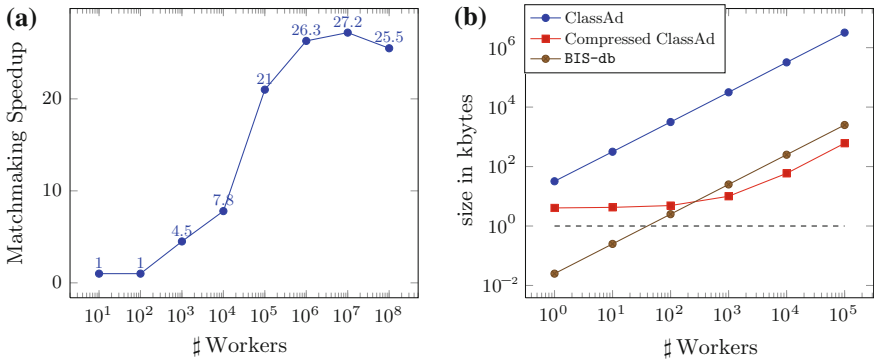


Fig. 5 Binary matchmaking speedup and resource information reduction. **a** Matchmaking speedup compared to Condor negotiator. **b** Resource information size reduction compared to Condor ClassAd

5 Primary Evaluation

We evaluate the efficiency of the proposed technique using two benchmarks: Speedup of matchmaking computational time, and reduction of the resource information size. information. Figure 5a displays the matchmaking speedup of our matchmaker compared to Condor negotiator. We run both matchmaker procedure codes for matchmaking one job to a number of workers $n \in \{10^1, 10^2, \ldots, 10^8\}$. We made 100 runs for each case and took the average. The run-time t is computed in milliseconds. For $n < 10^3 \Rightarrow t < 1$. The speedup grows for $n \in \{10^3, 10^4, 10^6\}$ and becomes nearly stable at 26 upwards. Figure 5b displays the size of the resource information for a number of workers $n \in \{10^0, 10^1, \ldots, 10^5\}$. We compare the size of BIS-db with: the actual text size of Condor ClassAd IS-db, and a compressed Condor IS-db using 7z LZMA2 [22] file compressor. The reduction grows with the number of workers to reach $\approx$99.91 % for $n = 10^5$. The reason is that the size of Condor machine ClassAd of one worker is $\approx$32 Kbytes while WAR is only 26 bytes. The 7z compressor achieves a better reduction for $n > 10^3$, but the compression time grows dramatically, +35 minutes for $n = 10^5$, which makes file compression improper for a large number of workers.

6 Conclusions

One important question to achieve efficient inter-Grid scheduling is how to perform matchmaking between a particular local job and the workers of a remote domain. Performing matchmaking remotely would result in computational overhead, while performing matchmaking locally would result in high data traffic. In this position paper, we introduced a coordinated scheduling technique which is

based on binary matchmaking and representation of the Grid resource information. Each worker information is stored in a 26-Byte sized record. Matchmaking is carried out in the local domain using logical AND and XOR operations. Our primary results show that the proposed technique achieves 26 speedup in the matchmaking process compared to Condor negotiator, and a reduction up to 99.92 % in the resource information size compared to Condor ClassAd.

References

1. Foster, I., Kesselman, C., Tuecke, S.: The anatomy of the grid: enabling scalable virtual organizations. Int. J. Supercomputer Appl. **15**(3), 200–222 (2001)
2. Raman, R., Livny, M., Solomon, M.: Matchmaking: Distributed resource management for high throughput computing. In: Proceedings of the Seventh IEEE International Symposium on High Performance Distributed Computing (HPDC7), Chicago, IL, July 1998
3. Evers, X., de Jongh, J.F.C.M., Boontje, R., Epema, D.H.J., van Dantzig, R.: Condor flocking: load sharing between pools of workstations. Department of Technical Mathematics and Informatics, Delft University of Technology, Delft, The Netherlands, Tech. Rep., (1993)
4. Laure E., et al.: Programming the Grid with gLite. CERN, Geneva, Tech. Rep., Mar 2006
5. Frey, J., Tannenbaum, T., Livny, M., Foster, I., Tuecke, S.: Condor-g: a computation management agent for multi-institutional grids. Cluster Comput. **5**(3), 237–246 (2002)
6. Schuller, B., et al.: Chemomentum-UNICORE 6 based infrastructure for complex applications in science and technology. In: Euro-Par Workshops, pp. 82–93, (2007)
7. Buyya, R., Abramson, D., Giddy, J.: Nimrod/g: an architecture for a resource management and scheduling system in a global computational grid. In: InProcee. HPC ASIA 2000, pp. 283–289 (2000)
8. Schopf, J.: Ten actions when superscheduling. In: Global Grid Forum. (2001)
9. Butt, A.R., Zhang, R., Hu, Y.C.: A self-organizing flock of condors. J. Parallel Distrib. Comput. **66**(1), 145–161 (2006)
10. Weissman, J.B., Grimshaw, A.S.: A federated model for scheduling in wide-area systems. In: Proceedings of the 5th IEEE International Symposium on High Performance Distributed Computing, ser. HPDC '96, p. 542. IEEE Computer Society, Washington, DC, USA (1996)
11. Shan, H., Oliker, L., Biswas, R.: Job superscheduler architecture and performance in computational grid environments. In: Proceedings of the 2003 ACM/IEEE Conference on Supercomputing (SC 2003), pp. 44–58 (2003)
12. Daval-Frerot, C., Lacroix, M., Guyennet, H.: Federation of resource traders in objects-oriented distributed systems. In: Proceedings of the International Conference on Parallel Computing in Electrical Engineering, ser. PARELEC '00, p. 84. IEEE Computer Society, Washington, DC, USA (2000)
13. Lai, K., Rasmusson, L., Adar, E., Zhang, L., Huberman, B.A.: Tycoon: an implementation of a distributed, market-based resource allocation system. Multiagent Grid Syst. **1**(3), 169–182 (2005)
14. Ranjan, R.: Coordinated resource provisioning in federated grids. Ph.D. dissertation, The University of Melbourne, Australia, July 2007
15. Litzkow, M., Livny, M., Mutka, M.: Condor: a hunter of idle workstations. In: Proceedings of the 8th International Conference of Distributed Computing Systems, June 1988
16. Azab, A., Meling, H.: Slick: a coordinated job allocation technique for inter-grid architectures. In: 7th European Modelling Symposium (EMS), Nov 2013

17. Azab, A., Meling, H.: Decentralized service allocation in a broker overlay based grid. In: Proceedings of the 1st International Conference on Cloud Computing, ser. CloudCom '09, pp. 200–211. Springer, Berlin (2009)
18. Rowstron, A., Druschel, P.: Pastry: scalable, decentralized object location, and routing for large-scale peer-to-peer systems. In: Middleware, pp. 329–350 (2001)
19. 2.5 Submitting a Job. http://research.cs.wisc.edu/htcondor/manual/v7.8/
20. Zhang, X., Freschl, J.L., Schopf, J.M.: A performance study of monitoring and information services for distributed systems. In: Proceedings of the 12th IEEE International Symposium on High Performance Distributed Computing, ser. HPDC '03, p. 270. IEEE Computer Society, Washington, DC, USA (2003)
21. Dongarra, J., Hempel, R., Hey, A., Walker, D.: Mpi: a message-passing interface standard. Int. J. Supercomputer Appl. **8**, 159–416 (1994)
22. 7z Format. http://www.7-zip.org/

Complex Objects Remote Sensing Forest Monitoring and Modeling

Boris V. Sokolov, Vyacheslav A. Zelentsov, Olga Brovkina, Victor F. Mochalov and Semyon A. Potryasaev

Abstract In this paper the concept of integrated modeling and simulation processes of the Complex Natural and Technological Object (CNTO) is presented. The main goal of the investigations consists in the practice of the predetermined modeling. The practice direction as the remote sensing forest monitoring is proposed by the authors. Here the methodical foundations of the integrated modeling and simulation, the process of CNTO operation, the technology of the remote sensing forest monitoring are considered. Principal concern is attended to the continuity of the model and object solving practical issues. More over results of CNTO remote sensing forest monitoring make it possible to adapt models of this system to changing environment conformably to the forest management.

Keywords Complex natural technological object · Control process · Simulation model · Processing of the space and airborne measurements · Forest monitoring

1 Introduction

In practice the processes of CNTO operation are non-stationary and nonlinear. The perturbation impacts initiate the CNTO structure-dynamics and predetermine a sequence of control inputs compensating the perturbation. In other words we

B. V. Sokolov (✉) · V. A. Zelentsov · V. F. Mochalov · S. A. Potryasaev
Russian Academy of Science, Saint Petersburg Institute of Informatics and Automation (SPIIRAS), St. Petersburg, Russia
e-mail: sokol@iias.spb.su

O. Brovkina
Global Change Research Centre, Academy of Science of the Czech Republic, Prague, Czech Republic

B. V. Sokolov
University ITMO, St. Petersburg, Russia

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_37,

always come across the CNTO structure dynamics in practice. For example, forest monitoring is considered. There are many possible variants of CNTO structure dynamics control [1].

In this paper we propose the practice of the predetermined modeling where CNTO is a Remote Sensing forest monitoring. Earlier various combinations of the analytical and simulation models were considered at several conferences with the similar theme.

Research issues addressed in the paper are conducted on the base of a comprehensive simulation theory of proactive monitoring and management of the structural dynamics of natural and technological objects (CNTO) and appropriate monitoring and management [1].

Estimation of required adequacy of modelling takes a special place in solving the problems of modelling complex objects $Ob^{op}_{<\ >}$ (actual or abstract). Obviously, it is necessary to evaluate every time how it is adequate in relation to the $Ob^{op}_{<\ >}$. An inaccurate initial assumptions in determining the type and model structure, error measurement in the test (experiment), computational error in the processing of measurement information could be the reasons for the inadequacy of the $Ob^{op}_{<\ >}$. Using an inadequate model can lead to significant economic losses, and to the default tasks of the actually existing system.

Following to the qualimetry of models consider two classes of simulated systems [1–3]. The first class consists of the systems with which an experiment (test) can be conducted and gotten by measuring the values of the system characteristics. Good example of the first class CNTO is Remote Sensing of Forestry.

Figure 1 presents the generalized technique for estimating and controlling the quality of models of objects of the first class.

In this figure, we take the following notation: 1, for forming the goals of functioning of $Ob^{op}_{<\ >}$; 2, for determination of input actions; 3, for setting goals of modeling; 4, for the modeled system (objects $Ob^{op}_{<\ >}$) of the first class; 5, for the model ($Ob^{m}_{<\ \theta\ >}$) of the investigated system $Ob^{op}_{<\ >}$; 6, for the estimation of the quality of a model (poly-model system); 7, for controlling the quality of models; 8, for controlling the parameters of models; 9, for controlling the structures of models; and 10, for changing the concept of model description.

All CNTO (including complex objects remote sensing processes and systems) working in an autonomous mode are examples of systems of the first class.

The second class of the simulated systems are systems that is impossible to carry out experiments. Among such systems are large-scale economic and social systems, complex organizational and technical systems operating in conditions of substantial uncertainty to the external environment, or virtual objects created as a result of mental activity of man.

Features of CNTO don't admit to achieve the degree of adequacy of the process description. Therefore, in practice the multiple-domain description of the study is used.

Proposed in earlier studies many variation of the formal description of objects and control subsystems included in monitoring and management of CNTO have

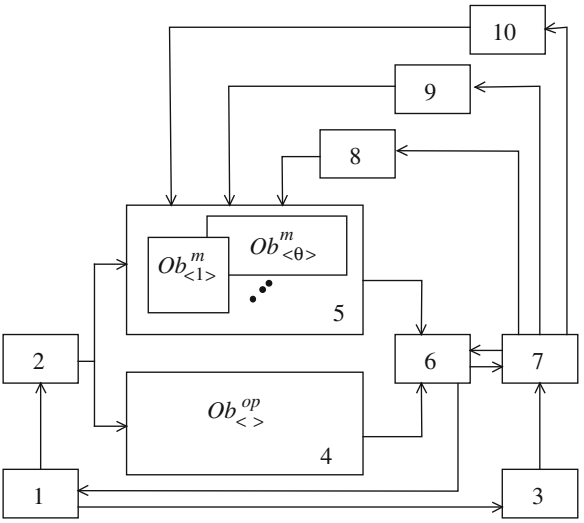


Fig. 1 The generalized technique of estimation and control of the quality of models of the first class

the broad prospects for the formulation and solution of problems as proactive (adaptive–proactive) synthesis of CNTO and proactive (adaptive and proactive) selection of complex models and algorithms of the software management and control objects in a constantly changing environment [2].

Analysis of earlier research on this subject demonstrates [1, 2, 4], that in this case the composition of tasks of the CNTO control should be supplemented by structural problems and parametric adaptation of the special mathematical software.

Consider in more detail the formal tasking of the structural adaptation of the CNTO models.

2 Problem Statement

Generalized formal statements of the structural adaptation CNTO model can be represented as:

$$\begin{aligned} &AD\left(M_{\Theta}^{(l)}, \bar{P}_{cs}\right) \rightarrow \min, \\ &t_{st}\left(\vec{w}^{(3)}, M_{\Theta}^{(l)}\right) \leq \bar{t}_{st}, \\ &M_{\Theta}^{(l)} \in \bar{\bar{M}},\ \vec{w}^{(3)} \in W^{(3)},\ M_{\Theta}^{(l)} = \bar{\bar{\Phi}}\left(M_{\Theta}^{(l-1)}, \vec{w}^{(3)}, \bar{P}_{cs}\right), l = 1, 2, \ldots, \end{aligned} \tag{1}$$

where $AD(M_{\Theta}^{(l)})$ is a functional characterizing the adequacy of the model $M_{\Theta}^{(l)}$ for CNTO. The latter is described, in its turn, with a set $\bar{P}_{cs}(t) = \{\bar{P}_{g}^{(cs)}, \bar{g} = 1, \ldots, \bar{G}\}$

of characteristics; t_{st} is a total time of CNTO models structure adaptation; $\overline{t_{st}}$ is a maximal allowable time of structural adaptation; $\overline{\overline{\Phi}}$ is an operator of iterative construction (selection) of the model $M_{\Theta}^{(l)}$, l is the current iteration number; $W^{(3)}$ is a set of allowable values for the vectors of structure-adaptation parameters.

$$T_{st}\left(\vec{w}^{(3)}, M_{\Theta}^{(l)}\right) \rightarrow \min, \tag{2}$$

$$AD\left(M_{\Theta}^{(\bar{l})},\ P\right) \leq \varepsilon_2, \tag{3}$$

$$M_{\Theta}^{(l)} \overline{\overline{\in}}\, M, \vec{w}_{(3)} \in W_{(3)}, \tag{4}$$

$$M_{\Theta}^{(l)} = \overline{\bar{\Phi}}\left(M_{\Theta}^{(l-1)}, \vec{w}^{(3)}, \bar{P}_{cs}\right) \tag{5}$$

where ε_2 is a given constant establishing an allowable level of the CNTO model $M_{\Theta}^{(l)}$ adequacy, $\Theta \in \hat{I}$, $\bar{\bar{M}}$ are a set of the CNTO models.

The analysis of expressions (1) shows that the structural adaptation starts and stops according to a criterion characterizing the similarity of a real object and an object described via models (a condition of models adequacy is applied) [5]. The adequacy of CNTO models does not mean description of all "details". It means that simulation results meet the changes and relations observed in reality.

Listed equations can be interpreted in relation to the forest monitoring models (for the forest structure, biomass estimation, forest growth models).

The main purpose of quantitative estimation of the model adequacy at time is to raise decision-maker's confidence in conclusions made on real situation. Therefore, the utility and correctness of CNTO simulation results can be measured via adequacy degree of models and objects.

Analysis of relations defining a common procedure of the proactive structural adaptation planning models and operational control of CNTO demonstrates that its implementation needs a set specific algorithms of iterative selection of the models (multiple-complexes). These models are denoted as $M_{\Theta}^{<k,l>}$, where k—the current number of management cycle ($k = 1,\ldots,K$); l—iteration current number, where the design (selection) image is performed ($l = 1,\ldots,L$); Θ—current model number from the model bank or model repository ($\Theta = 1,\ldots,\Theta$). Let's the model repository is named as the plurality:

$$\overline{\overline{M}}^{<k>} = \left\{ M_{\Theta}^{<k,l>};\quad k = 1,\ldots,K;\ l = 1,\ldots,L;\ \Theta = 1,\ldots,\Theta^{<k>} \right\} \tag{6}$$

A class of algorithms based on evolutionary modeling technology is one of the promising class of algorithms that can use a constructive way to solve the tasks of the structural adaptation.

3 Algorithm of Structure Adaptation of Models

The second group of algorithms for structural adaptation of CNTO models is based on the evolutionary (genetic) approach. As before in Problem statement, let us exemplify these algorithms in the structural adaptation of a model describing structure dynamics of one CNTO output characteristic of one element of the vector $[\vec{y}(t_{<k>})]$.

The residual of its estimation via the model M_θ, as compared with the observed value of the characteristic, can be expressed like this:

$$Q_{\langle k\rangle}^{\langle\theta\rangle} = \left[\psi_{\langle\theta,k\rangle}\left(\vec{x}\left(t_{\langle k\rangle}-1\right),\vec{u}\left(t_{\langle k\rangle}\right),\vec{\xi}\left(t_{\langle k\rangle}\right),\vec{\beta}_\theta,t_{\langle k\rangle}\right)-\tilde{y}\left(t_{\langle k\rangle}\right)\right] \tag{7}$$

The formula define a dynamic system describing CNTO structure-dynamics control processes. Here $\vec{x}(t)$ is a general state vector of the system, $\vec{y}(t)$ is a general vector of output characteristics. Then, $\vec{u}(t)$ is a control vector. Here $\vec{u}(t)$ represents CNTO control programs (plans of system functioning), $\vec{\xi}(t)$ is a vector of perturbation impacts. The vector $\vec{\beta}_\Theta$ is a general vector of CNTO parameters.

To simplify formula, we assume that the perturbation impacts $\vec{\xi}(t)$ are described via stochastic models. Thus, the following quality measure can be introduced for the model M_Θ:

$$\bar{Q}_{<K>}^{<\theta>} = \sum_{k=1}^{K} g(K-k)Q_{<k>}^{<\theta>}, \tag{8}$$

where $0 \le g \le 1$ is a "forgetting" coefficient that "depreciate" the information received at the previous steps (control cycles) [4]. If $g = 0$ then $\bar{Q}_{<K>}^{<\theta>} = Q_{<K>}^{<\theta>}$, i.e., the weighted residual is equal to one received at the last step, as the prehistory have been "forgotten". An extension of formula (9) was proposed in [5]. The coefficient $g^{(K-k)}$ was substituted for the function $f(K)$:

$$\bar{Q}_{\langle K\rangle}^{\langle\theta\rangle} = \sum_{k=1}^{K} f(K-k)Q_{\langle K\rangle}^{\langle\theta\rangle}, \quad \theta = 1,\ldots,\Theta \tag{9}$$

Here $f(\cdot)$ is a monotone decreasing function of "forgetting". It has the following properties:

$$\begin{aligned} &f(\alpha) > 0,\ f(0) = 1,\ \lim f(\alpha) = 0,\\ &f(\alpha) \ge f(\alpha+1),\ \alpha \overset{\alpha\to?}{=} 0,1,\ldots \end{aligned} \tag{10}$$

Now the structural-adaptation algorithm is reduced to a search for the structure $M_{\Theta\prime}$, such that

$$\bar{Q}^{<\theta'>}_{<K>} = \min_{\theta=1,\ldots,\Theta} \bar{Q}^{<\theta>}_{<K>}. \tag{11}$$

Thus, it is necessary to calculate the quality measures (8) for all competitive structures M_Θ,q = 1,…,Q of CNTO models at each control cycle $k = 1,\ldots,K$. All quality measures should be compared, and the structure M_Θ with the best measure (minimal residual) should be chosen.

Another way to choose model $M^{(l)}_\Theta$ is probabilistic approach. In this case the following formula is used

$$p_1(M^l_\Theta) = \frac{\sum_{\rho=k-d}^{k-1} J_i(M_i^{<\rho,L>})}{\sum_{\Theta=1}^{\Theta^{<k-1>}} \sum_{\rho=k-d}^{k-1} J_i(M_\Theta^{<\rho,L>})}, \tag{12}$$

where $J_i(M_i^{<\rho,L>})$–generalized quality measure value of model $M_i^{<\rho,L>}$ functioning on previous time intervals, d—is a "forgetting" coefficient. It should be emphasized that the calculation of the quality measure is expected to carry out each time on the basis of a solution of the problem of multi-type selection. Thus, despite the random choice of the next model (multiple-model complex) greater opportunity to be chosen gets the model, which had the best value of the generalized quality measure in the previous cycle control. Earlier various combinations of the analytical and simulation models were considered at the conferences with the similar theme.

The parametric adaptation of the model M_Θ [4, 6] should follow the structural one.

It is important to determine a proper "forgetting" function under the perturbation impacts $\vec{\xi}(t)$. The higher is the noise level in CNTO, the slower decrease of the function should be implemented. However, if CNTO highly changes its structure then the function $f(\alpha)$ should be rapidly decreasing in order to "forget" the results of the previous steps [6]. It can be demonstrated that the structural-adaptation algorithms based on model construction (synthesis) of atomic models (modules) are rather similar to the algorithms of the CNTO structure-functional synthesis [1]. These algorithms only differ in the interpretation of results.

4 Thematic Processing of the Remote Sensing Imagery

Thematic processing of the Remote Sensing data is the key link in the system of the forest monitoring. Generally the primary and secondary processing is applied. The operations are done based on the modeling and simulation in automatic mode supported by the expert's knowledge.

The experience of the thematic treatment of the many and hyperspectral data with the high spatial resolution defined some important factors. One of them is the

data presentation with the automatic identification of the test sites for algorithm training and adaptation. The next one is the complex processing of the source many (hyper)spectral and temporal Remote Sensing data and ground measurements. Third factor is the data results calibration and validation and optimal application of the spectral features data base of the landscape elements with reference to seasonal and daily variability. Lastly, the organization of the distributed access to the data is exchanged on the base of the special portals, geographic informational system capability and crowd sourcing.

The informational flow rises and the necessity of the integrated modeling is determined. At that the qualitative and quantitative requirements are increased.

Commonly the main steps of the thematic processing of the Remote Sensing data are designated for the qualitative solution of the integrated modeling task: input data, optimal survey parameters, change reflective and radiative settings of the landscape elements in seasonal and daily variability, data acquisition and treatment, imager radiometric correction and calibration, imagery geometric correction, maintain of the system of initial data relative to the reflective and radiative characteristics of the landscape elements, combination of methods and algorithms of the thematic treatment (cluster analysis, Fourier analysis, method of principal components, classification algorithms and others) (blocks 8 and 9, Fig. 1), CNTO modeling and simulation on the base of the expert's knowledge (blocks 1 and 3, Fig. 1), analysis of the situation dynamic based on the multi-temporal Remote Sensing data treatment (block 6, Fig. 1), predictive modeling of the step 5 results influence to ecological situation (block 5, Fig. 1), crowdsourcing through the geoinformational portal application (blocks 1, 3 and 4, Fig. 1), automatic environmental assessment in the space ecological monitoring network (blocks 6 and 7, Fig. 1), creation of the thematic layers and attributive information of the monitoring.

Analysis of the main trends for modern systems of the space forest monitoring indicates their peculiarities such as: multiple aspects and uncertainty of their behavior, hierarchy, structure similarity in the detection and recognition of the landscape elements, redundancy from the source data and variety of implementations for control functions. One of the main features of modern systems of the space forest monitoring is the variability of their parameters and structures due to objective and subjective causes at different phases of the system life cycle. In other words we always come across the system structure dynamics in practice.

5 Example

Example demonstrates determination of the optimal parameters for the integrated modeling and simulation of the CNTO described as the system of the space forest monitoring (block 3, Fig. 1).

This task includes the determination of the optimal year period for the airborne data acquisition under condition that the obtained airborne data is planned to use for the forest species map creation at the Moravian-Silesian Beskydy region.

An initial data for the model was defined by the archival airborne hyperspectral data of the study Beskids Mountain forest area. Hyperspectral images have been obtained in August 2009, August 2010, September 2010, May 2011 and September 2011 at about midday. In additional, parameters of the airborne equipment (AISA and HyMap hyperspectral sensors), such as data spectral and spatial resolutions were the initial data also.

Particularities of the territory consist of the mountain relief with the varied altitude from 500 to 900 m. As a consequence, the forest vegetation grows on the mountain slopes with the various solar radiations. The dominant forest type is even aged monoculture Norway spruce (Picea abies L.). However, it should be noted, that mixed forest occupies a substantial part of the territory and differs by the percentage ratio of the coniferous [Norway spruce, Scotch pine (Pinus sylvestris L.)] and broadleaves species [European beech (Fagus sylvatica L.) and ash (Fraxinus excelsior L.)] (Michalko, J. 1986: Geobotanicka mapa CSSR. Veda, Bratislava: 186 p.).

Analysis of the spectral features of spruce, beech, mixed forest, grass and young forest sites at the sun and shadow mountain slopes have been carried at various vegetation periods on the study forest area in the block 3 of the model.

Hyperspectral vegetation indices (narrow-band NDVI and CARI) were calculated (block 4, Fig. 1). Chlorophyll Absorption Ratio Index (CARI) measures the depth of chlorophyll absorption at 670 nm relative to the green reflectance peak at 550 nm and the reflectance 700 nm. Narrow-band Normalized Difference Vegetation Index uses the highest absorption and reflectance regions of chlorophyll 910 and 682 nm. Values of indices for sites with spruce, beech, mixed forest, grass and young forest at the sun and shadow mountain slopes were compared by the dispersion measures. Time periods with maximum difference in the index values for spruce, beech, mixed forest, grass and young forest at the sun and shadow mountain slopes were determined at the last week of May, the last week of August and the third week of September.

Thereby the optimal year periods for the acquisition of airborne hyperspectral data for the forest species map creating were signed considering the particularity of the Beskids Mountain forest area.

The perturbation influences were presented by the control model parameters, that could be evaluated on the real data available in CNTO and parameters that could be evaluated via simulation models for different scenarios of future events.

6 Conclusion

In the study of monitoring and management of ecological and technological objects found that the necessary degree of adequacy of these complex objects can be achieved in the case when the modeling process is manageable by itself.

Possible reasons for the inadequacy of the models may be inaccurate initial assumptions in determining the type and model structure, measurement error in the tests (experiments), computational errors in the processing of measurement information. Using an inadequate model can lead to significant economic losses, emergencies, to the default tasks of the actual current system.

Thus, the study proposes a multi-level multi-stage procedure for parametric and structural adaptation as traditionally used in such cases, mathematical models (analytical and simulation), and the models used in modern engineering knowledge. As the latest models, the report discusses models. On the base of these models on a constructive level structural adaptation of the model process of the monitoring of CNTO is considered to the current, a posteriori and priori data about their condition on the basis of aerospace landscape monitoring.

The main difference and advantage of the proposed approach in the study are that the procedure of random search of purposeful selection of models most adequately describes the dynamics of changes in the state of ecological (forest) objects not on the basis of heuristics approach but on the basis of results of exact calculations and optimization of the structural dynamics of these control objects [1, 4].

Acknowledgments The research described in this paper is supported by the Russian Foundation for Basic Research (grants 12-07-00302, 13-07-00279, 13-08-00702, 13-08-01250, 13-07-12120-ofi-m, 12-07-13119-ofi-m-RGD), Department of Nanotechnologies and Information Technologies of the RAS (project 2.11), by Postdoc project in technical and economic disciplines at the Mendel University in Brno (reg. number CZ.1.07/2.3.00/30.0031), by ESTLATRUS projects 1.2./ELRI-121/2011/13 «Baltic ICT Platform» and 2.1/ELRI-184/2011/14 «Integrated Intelligent Platform for Monitoring the Cross-Border Natural-Technological Systems» as a part of the Estonia–Latvia–Russia cross border cooperation Program within European Neighborhood and Partnership instrument 2007–2013. This work was partially financially supported by Government of Russian Federation, Grant 074-U01.

References

1. Ohtilev, M.Y., Sokolov, B.V., Yusupov, R.M.: Intellectual technologies for monitoring and control of structure-dynamics of complex technical objects. Nauka, Moscow (2006)
2. Skurihin, V.I., Zabrodsky, V.A., Kopeychenko, Y.V.: Adaptive control systems in machine-building industry. Mashinostroenie, Moscow (1989)
3. Rastrigin, L.A.: Adaptation of Complex Systems. Zinatne, Riga (1981)
4. Ivanov, D., Sokolov, B., Kaeschel, J.: A multi-structural framework for adaptive supply chain planning and operations with structure dynamics considerations. Eur. J. Oper. Res. **200**(2), 409–420 (2010)
5. Sokolov, B., Zelentsov, V., Yusupov, R., Merkuryev, Y.: Information fusion multiple-models quality definition and estimation. In: Proceedings of the International Conference on Harbor Maritime and Multimodal Logistics M&S, pp. 102–111. Vienna, Austria, 19–21 September 2012
6. Ivanov, D., Sokolov, B.: Control and system-theoretic identification of the supply chain dynamics domain for planning, analysis and adaptation of performance under uncertainty. Eur. J. Oper. Res. **224**(2), 313–323 (2012) (Elsevier, London)

Building a Non-monotonic Default Theory in GCFL Graph-Version of RDF

Alena Lukasová, Martin Žáček and Marek Vajgl

Abstract The aim describes the idea of graph-based representation of clauses. This approach follows the Richards idea of graph-based clausal form knowledge representation. Moreover, it enabled to build up the graph-based formal system GCFL (Graph-based Clausal Form Logic) that cannot only illustrate knowledge bases graphically, but also allows us to obtain consequents of a knowledge base in a special graph-based way. The article continues the idea by creation of a graph-based formal system of generating revisable theories following the known Reiter's default principle of building non-monotonic theories.

Keywords Graph · RDF · Default theory · GCFL · Clausal form logic

1 Introduction: Clausal form Logic and Its Graph-Based form GCFL with Quantifiers

The area of formal logic is a very interesting part of information science. There have been several areas of research fields aimed at finding the way how to represent knowledge formally aiming at handling information and using it to inference in a similar way to human thinking. Most of the nowadays existing approaches are built on the first order predicate logic (FOPL) (as presented in e.g. [1]), extended by different expression behaviour (fuzzy predicate logic, description

A. Lukasová · M. Žáček (✉) · M. Vajgl
Department of Computers and Informatics, University of Ostrava, Ostrava, Czech Republic
e-mail: martin.zacek@osu.cz

A. Lukasová
e-mail: alena.lukasova@osu.cz

M. Vajgl
e-mail: marek.vajgl@osu.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_38,

logic [2]). One of the approaches is clausal logic (see e.g. [3]), based on clauses representing knowledge. However, for some examples, labelling inference mechanism for clausal logic may be confusing, especially when the relation between clauses in not clearly defined. Using a more human-intuitive mechanism, such as graphical representation, may improve readability of the knowledge. Graphical representation has already been used in more areas, e.g. semantic networks [4], conceptual graphs [5, 6], or combined with RDF [7].

Usability of graph-based representation with clausal form logic has already been presented in [7], too. However, these approaches are naturally based on a closed world assumption, where all knowledge is expected to be already captured and stored in a knowledge base. There are also systems with an open world assumption, where the knowledge base can be reviewed according to a new knowledge. Once again, there is multiple different formal logic approaches used to work with incomplete information—e.g. description logic [8], modal logic [9]. One of the most used approaches is Reiter's default logic [10]. The following contribution presents how the default logic can be extended using graph clausal form logic to provide a formal system working with incomplete information.

Our idea of graph-based representation of clauses connects to that of Richard's CFL and proposes a complete graph-based inference system GCFL. The knowledge base of the system consists of clauses represented by graphs.

To distinguish statements in the antecedent part of a graph-based clause from statements in its consequent part, we introduce a convention

- to draw the arcs of antecedent vectors by dashed lines and
- to draw the arcs of consequent vectors by solid lines.

All the vectors (with solid or dashed lines) represent atomic statements and generally have the following structure (Fig. 1):

From the point of view of first order predicate logic, our formal system GCFL is a special form of Richard's clausal form logic that uses only binary predicates to express relationships between concepts as well as properties of the concepts.

General statements, such as "It is not possible that X is a man and simultaneously X is a woman", expressing relationships between concepts (here "man" and "woman"), could have the following (Fig. 2) graph-based form of representation (a clause without a consequent) in GCFL.

2 Default Logic

Default logic is an important method of knowledge representation and reasoning, because it supports reasoning with incomplete information and allows revising theories that have been previously built up. Default logic can be naturally used in many application domains, such as information retrieval, specifications of systems, diagnostic of problems, etc.

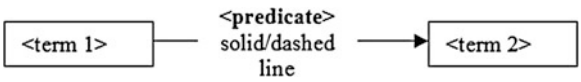


Fig. 1 Vector

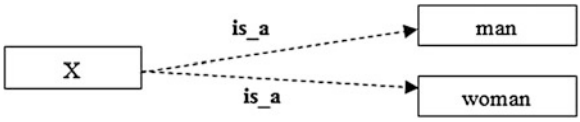


Fig. 2 A clause without the consequent

Default theories can be used either to model reasoning with incomplete information, which was the original motivation, or as a formalism which enables compact representation of knowledge that varies in time and ought to be revised.

To define logic for default reasoning, we must solve a two-fold task:

1. To provide a formal definition of the extensions to an underlying first order theory induced by a set of defaults.
2. To provide a proof theory in the form of a procedure which, given formula *w*, determines whether *w* can be believed, i.e. whether there is an extension induced by the defaults which contain w.

In this paper we refer to the original Reiter's article [10] for the first task and propose a GCFL graph-based [11] method of reasoning in the RDF for the second task.

3 Non-monotonicity of Default Theories

The default theory is an attempt to formally capture the idea that a knowledge base (a set of beliefs) may enable to make certain conclusions even if these conclusions are not logically implied by the knowledge base. We assume a knowledge base as a set of beliefs completed by its logical consequents.

A conclusion made on the basis of a default logic principle only applies if there is a default rule that supports its derivation and if the conclusion cannot be deductively derived from the existing knowledge base.

A fundamental feature of the first order logic is that it is monotonic, i.e. if A and B are sets of first order formulae and A⊢w then A, B⊢w.. It means: *What was valid in the presence of information A remains valid when new information B has been added*.

In contrast, any logic which presumes to formalize default reasoning must be revisable and so it must be non-monotonic.

4 Default Rules

The meaning of a default rule is built up on notions of provability and consistency with respect to a given knowledge base. Together with conclusions sanctioned by a set of default rules, the knowledge base is called its default extension. Default rules state how to extend a knowledge base with respect to a certain set of statements we believe.

A default theory is a pair (D, W). W is a set of logical formulae, called the background theory, that formalize the facts that are known for sure, D represents a defeasible information. D is a set of default rules, each one being of the form (1):

$$\frac{\alpha = \alpha_1, \ldots, \alpha_m : \beta_1, \ldots, \beta_n}{\gamma} \tag{1}$$

$\alpha = \alpha_1, \ldots, \alpha_m$ stands for the *prerequisite* (*knowledge base*), $\beta_1, \ldots, \beta_n$ are *justifications*, γ is the *conclusion*.

Such a rule is informally interpreted as "if α is true, and $\beta_1, \ldots, \beta_n$ are consistent with what is known, then conclude γ by default".

The premises of the default rule consist of two components. The first means the current knowledge base (a set of first order logic expressions) and is referred to as the prerequisite. These expressions must be proven to be true (in a standard deductive sense) or believed in a common sense. The second set is referred to as the consistency set. These expressions must be consistent with the current knowledge base. Thus from the formal point of view, it must be proven that the negation of the expressions does not take source from the current knowledge base. A default rule can be applied to the theory if its precondition is entailed by the theory and its justifications are all consistent with the theory. If the rule is proven to be applicable, then the expression referred to as the consequent is added into an extension of the theory. Application of a default rule results in adding its consequence to the extended theory. Other default rules may then be applied to the resulting theory. The default rules may be applied in a different order, and this may lead to different extensions.

A default rule is categorical or prerequisite-free if it has no prerequisite (or, equivalently, its prerequisite is tautological).

A default rule is normal if it has a single justification that is equivalent to its conclusion.

The categorical normal default logic is known and is formalized by negation as failure (2):

$$\frac{: \neg F}{\neg F} \tag{2}$$

for every fact F.

Reiter defines the "default negation" as a rule that says something like (3): "if we believe that the opposite statement does not hold, we can accept the statement".

$$\frac{\alpha = \alpha_1, \ldots, \alpha_m \ : \ \neg\beta_1, \ldots, \neg\beta_n}{\gamma} \tag{3}$$

A common default logic assumption is the known Closed World Assumption (CWA): "what is not known to be true is believed to be false".

Reasoning about the word under the closed Word assumption considerably simplifies representation of the world: only positive information is explicitly represented in the knowledge base, negative be inferred by default. When it concerns such a theory that no other default can be applied, the theory is called *extension of the default theory*.

5 Example of Logic Puzzle

Imagine two couples that do not known each other spend their holidays in a B&B. They are: John and his girlfriend Anett, and Philip with Renata. These couples are accommodated in rooms of the B&B of a family who offers to rent the first floor of their house. Because the owners had bad experience with some guests, they have installed a camera at the entrance to their apartment, where guests are not usually allowed. One day, while the four people are staying here, a valuable picture (icon) disappears from the wall of the owner house.

For example, we can assume to resolve this theft as follows:

Nobody got in but the four people, because the house is locked. All four guests could get to the apartment with their keys. The mentioned icon that disappeared is very valuable, which an expert on paintings might appraise. John studies painting, Renata studies art history, Philip studies engineering and Anet studies medicine. After a few years of studying art, John and maybe Renate have sufficient knowledge to appreciate precious antiques. Their possible motive of the theft might arise.

The initial knowledge base {T} = {t1, t2, t3, t4, t5, a1, a2, a3, a4, a5, a6, a7, a8} consists of clausal form rules t1, …, t5 and facts a1, …, a8.

In our example of graph-based clausal form representation of the knowledge base above, we draw the antecedent part of the graph representing a clause as vectors with dashed-line arcs and as the consequent part of the graph a vector with solid-line arcs.

Initial knowledge base {**T**}:
t1, …, t5 (Fig. 3) are universal clauses that represent general statements. The following a1,…, a4 (Fig. 4) clauses without antecedent represent positive facts (Table 1).

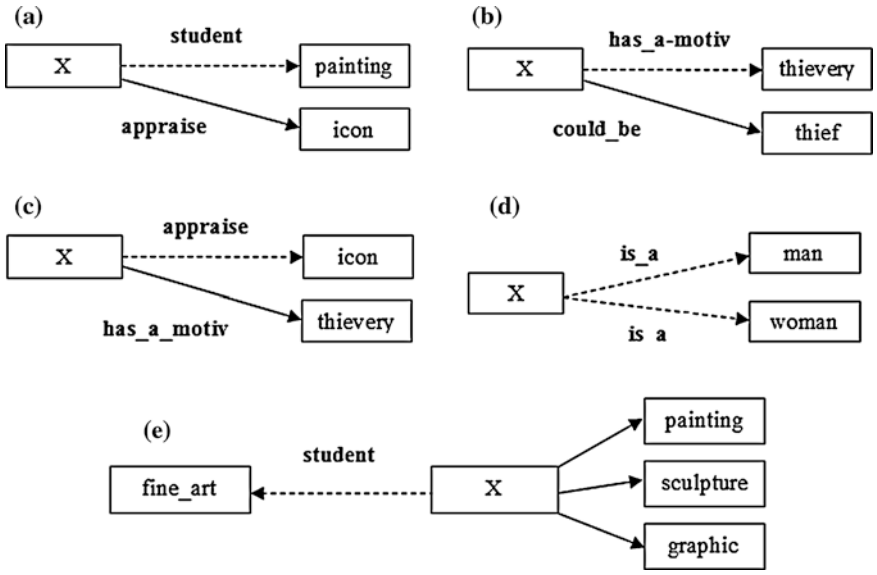


Fig. 3 Rules **a** t1, **b** t2, **c** t3, **d** t5, **e** t4

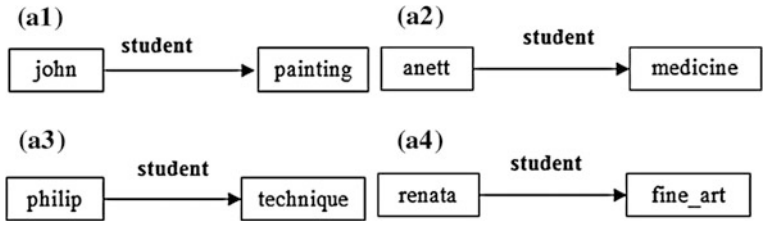


Fig. 4 Clauses without antecedent

Proof 1 (of the statement "John could be the thief" in the GCFL representation)
(1) t1
(2) Substitution {john/X} into t1 (Fig. 5a)
(3) a1
(4) cut (2), (3) - (t1,a1) (Fig. 5b; Table 2)

The conclusion c1: **could_be**(john, thief) of the step (10) now could be added to the initial knowledge base {T} because it represents a proper logical consequent in the monotonic formal system GCFL.

{T′} = {T, c1} now be the current knowledge base.

We suppose that it is possible to obtain a conclusion **could_be**(renata, thief) in a similar way.

Table 1 Initial knowledge base {**T**}—clauses corresponding to RDF vectors

t1: **student**(X, painting) → **appraise**(X, icon)
t2: **appraise**(X, icon) → **has_a_motiv**(X, thievery)
t3: **has_a_motiv**(X, thievery) → **could_be**(X, thief)\
t4: **student**(X, fine_art) → **student**(X, painting), **student**(X, sculpture), **student**(X, graphics)
t5: **is_a**(X, man), **is_a**(X, woman) →
a1: **student**(john, painting)
a2: **student**(philip, engeneering)
a3: **student**(anet, medicine)
a4: **student**(renata, fine_art),
a5: **is_a**(john, man)
a6: **is_a**(philip, man)
a7: **is_a**(anet, woman)
a8: **is_a**(renata, woman)

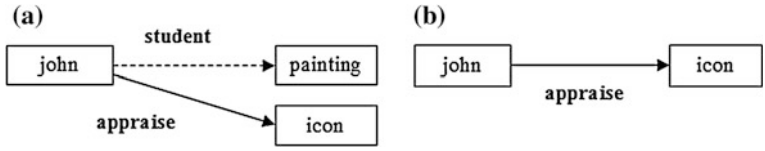


Fig. 5 GCFL representation

Table 2 Corresponding proof in the CFL representation

1. student(X, painting) → **appraise**(X, icon)	
2. → **student**(john, painting)	
3. appraise(X, icon) → **has_a_motiv**(X, thievery)	
4. has_a_motiv(X, thievery) → **could_be**(X, thief)	
5. student(john, painting) → **appraise**(john, icon)	subst. {john/X}
6. → **appraise**(john, icon)	cut 2., 5.
7. appraise(john, icon) → **has_a_motiv**(john, thievery)	subst. {john/X}
8. → **has_a_motiv**(john, thievery)	cut 6., 7.
9. has_a_motiv(john, thievery) → **could_be**(john, thief)	subst. {john/X}
10. could_be(john, thief)	cut 8., 9.

Proof 2
(1) Figure 6
(2) Substitution {renate/X} into 1) (Fig. 7)
(3) a4
(4) cut (2), (3) (Fig. 8).

It is obvious that the proof cannot lead to the conclusion d: **could_be**(renata, thief) neither in a direct nor in an indirect way of the proof.

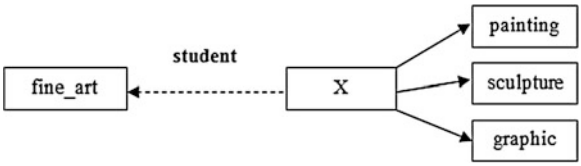


Fig. 6 Statement

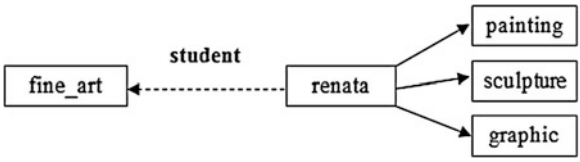


Fig. 7 Substitution

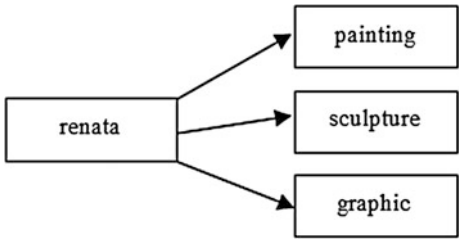


Fig. 8 Rule of cut

Adding an instance of the normal default rule δ offers a possibility to obtain a rather softer conclusion from the current knowledge base {T, c1} in default logic.

Normal default rule δ (4):

$$\frac{\{T'\} : \text{could\_be}(\text{X}, \text{thief})}{\text{could\_be}(\text{X}, \text{thief})} \tag{4}$$

and its instance (5)

$$\frac{\{T'\} : \text{could\_be}(\text{renata}, \text{thief})}{\text{could\_be}(\text{renata}, \text{thief})} \tag{5}$$

The clause $\rightarrow$ **could_be**(renata, thief) is not provable from the knowledge base {T, c1} and the same holds in the case of its negation—the clause **could_be**(renata, thief) $\rightarrow$.

This fact leads only to the conclusion of consistency d with the knowledge base {T, c1}, not to a logical consequent. It is because of neither d nor $\neg$d is provable from {T, c1}.

It means that with the help of the normal default rule we have obtained a default extension E of {T, c1}, it means E = {T, c1, d}.

But neither the proof in a direct way nor that one in an indirect way do not lead to the supposed conclusion. It is the case if it holds a rather "weaker" default conclusion (6)

$$\text{d: could\_be(renata, thief)} \tag{6}$$

consistent with the knowledge base $\{T'\}$.

For the validity of the preconditions contained in the knowledge base, John and Renata could take the mentioned icon. The problem seems to be unsolvable. Moreover, none of these two persons does not confess because they do not want to be ashamed in the eyes of the others. However, the owners of the house remember the newly installed camera and, seeing a bit unclear record of the time when the icon was stolen, they find out that it was taken by a woman.

Default rule δ' has to contain, as justification, a general statement is_a (X, woman), which represents the fact that the somewhat unclear record of a video camera is a woman.

Default rule δ' (7):

$$\frac{\{T, c1, d\} : \text{could\_be(X, thief)}, \text{ is\_a(X, women)}}{\text{could\_be(X, thief)}} \tag{7}$$

It is obvious, that only the case of substitution {renata/X} fulfils the default rule δ', so the thief must be Renata.

6 Example of RDF

Nowadays the accepted semantics for RDF (Resource Description Framework) language is a strictly non-monotonic one, even if it is in a way clear that the Web itself is not a monotonic object. At the same time, it is easy to understand that the compactness of using defaults rules cannot be exploited by the current idea of the Semantic Web. The two strictly and related concepts of non-monotonic and default reasoning are in a way solved by introducing a new explicit semantics for rdf:type and rdf:subClassOf. Please, notice that type, strictly speaking, is defined in the RDF with the following definition, found on the official Web page

```
<rdf:Property rdf:ID="type">
  <samePropertyAs
          rdf:resource="http://www.w3.org/
               1999/02/22-rdf-syntax-ns\#type"/>
</rdf:Property>
```

We show a simple example on the implementation of default inheritance. Consider a case where there is an rdf:Class of penguins that is a rdf:subClassOf of an

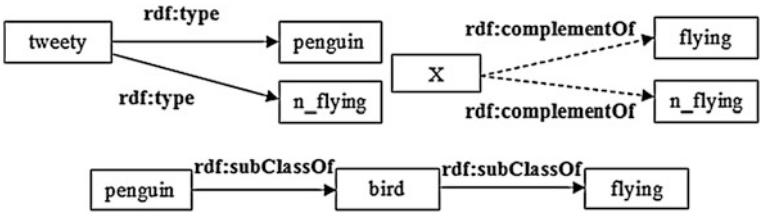
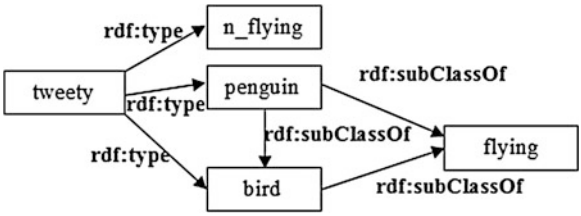


Fig. 9 Knowledge base (KB)

Fig. 10 The result



rdf:Class of birds that is a rdf:sublClassOf of an rdf:Class of flying things. There is also an rdf:Class of things that do not fly, which is defined as rdf:complementOf the class of the things that fly and, of course, Tweety, a penguin.

It is well-known in the RDF model that such rules could be displayed in a sequence of logical facts in Fig. 9.

Tweety is of rdf:type penguin, therefore exploiting the monotonic semantics of the property rdf:subClassOf Tweety is an instance of the class of things that fly. Since that is not true, we have made an RDF assertion which says explicitly that Tweety is an instance of the class of things that do not fly. Due to the monotonic semantics of rdf:complementOf, there will be a statement into the KB saying that Tweety cannot be an instance of two disjoint classes.

If we apply the default rules, the solution will result in Fig. 10.

A possible existence of a penguin that flies can be captured by the non-monotonic semantics because there is a direct way to infer that it flies, either by an explicit directed arch that says that it flies or by inference through the semantics of rdf:subClassOf. In this semantics, by default, any other instance of classes that are subclasses of the class of the birds will be considered an object that flies, avoiding the necessity of writing any statement relating to object's ability to fly.

7 Conclusion

The main aim of this article was to illustrate that representation of knowledge in RDF in a graphic version is very transparent, illustrative and demonstrative. Clear visualisation of using graphic representation demonstrates how the rules are

applied and how the inference is done. This can be achieved without any deep knowledge about the formal logic and its representation, including e.g. clausal logic and its mechanisms. On the other hand, as the article presents, the presented graph-based mechanism has the same inference capabilities and expressivity as the clausal logic, or as the traditional first-order logic, which is expressive, but not easy to understand and handle.

Moreover, the presented example shows its usage in the fields of non-monotonic formal systems and how the part of the previous knowledge can be revised by new information added into the knowledge base using the default logic. This is very important because current common systems based on knowledge representation need to be built on the open world assumption, which does not take for granted that "what is not known is not true". This approach is common in the semantic web and therefore it is an advantage for the formal system to be able to support this approach.

Acknowledgments The research described here has been financially supported by University of Ostrava grant SGS/PřF/2014. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not reflect the views of the sponsors.

References

1. Hamilton, A.G.: Logic for Mathematicians. Cambridge University Press, Cambridge (1978). ISBN 0-521-21838-1
2. Baader, F., et al.: The Description Logic Handbook—Theory, Implementation and Applications. Cambridge University Press, Cambridge (2003). ISBN 0-521-78176-0
3. Richards, T., et al.: Clausal form Logic: An Introduction to the Logic of Computer Programming. Addison Wesley, Reading (1989). ISBN 978-0201129205
4. Quillian, M.R.: Semantic memory. In: Minsky, M. (ed.) Semantic Information Processing, pp. 27–70. MIT Press, MA (1968)
5. Sowa, J. F.: Conceptual graph (Online), 2005 (Citation: 20.12.2013). http://www.jfsowa.com/cg/index.htm
6. Harmelen, F., Lifschitz, V., Porter, B. (eds.): Conceptual Graphs, Handbook of Knowledge Representation, pp. 213–237. Elsevier, Amsterdam (2008)
7. Lukasová, A., Vajgl, M., Žáček, M.: Reasoning in RDF graphic formal system with quantifiers. In: Proceedings of the International Multiconference on Computer Science and Information Technology, pp. 67–72 (2010)
8. Hustadt, U.: Do we need the closed world assumption in knowledge representation? In: Proceedings of 1st Work-shop KRDB'94, Reasoning about Structured Objects: Knowledge Representation Meets Databases. Saarbrücken, Germany (1994)
9. Emerson, E.A.: Temporal and modal logic. In: Handbook of Theoretical Computer Science. pp. 995–1072. Elsevier, Amsterdam (1995)
10. Reiter, R.: A logic for default reasoning. Artif. Intell. **13**, 81–132 (1980)
11. Lukasová, A., Žáček, M., Vajgl, M.: Reasoning in graph-based clausal form logic. IJCSI Int. J. Comput. Sci. Issues **9**(Issue 1, No 3), 37–43 (2012). ISSN (Online) 1694-0814

An Intranet Grid Computing Tool for Optimizing Server Loads

Petr Lukasik and Martin Sysel

Abstract The article describes the principles of the developed Intranet grid computing used in the corporate sector as a tool for optimizing computing loads on the server that is deployed for production planning and scheduling. ICT development allows companies to install higher computing performance with lower costs. This trend is particularly evident for investments in personal computers, laptops or smartphones. Investments in the backbone infrastructure (servers, networks) are controlled by a different philosophy. For this area, it is important (in many cases due to software licensing policy) which is a very rigorous consideration of the system performance parameters to be used. This result is well-known as a problem with high-loads on servers, as against almost negligible computing loads on the user-side.

Keywords Intranet · Grid · Optimizing the computing load · Production planning

1 Introduction

Effort of the optimization is distribution of some tasks to the grid computing and transfer the load from the server side to the client computers. Substitution of some standard batch jobs by the grid, greatly improves computational load on the server

P. Lukasik (✉) · M. Sysel
Department of Computer and Communication Systems, Faculty of Applied Informatics, Tomas Bata University in Zlin, Nam. T. G. Masaryka 5555 760 01 Zlin, Czech Republic
e-mail: plukasik@tajmac-zps.cz

M. Sysel
e-mail: sysel@fai.utb.cz
URL: http://www.fai.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_39,

side and significantly increase the consistency of planning data. Main advantage of the grid is solving the problem at the moment of creation. The result is to reduce the load on the server against the batch tasks.

2 Motivation for the Grid in Production Planning

Main goal to use the grid is better distribution of the computing load on corporate intranet and reduction of necessary investments on the server infrastructure. Comparing the load device on the network is a significant difference in the average computing load on the server and workstations.

The original idea is based on practical experience in managing ERP systems in engineering company. Claims for the planning process have a negative effect on the response of the system, especially when running batch jobs (Production Planning, Product Cost and Low-Level Code). Goal was to divide tasks into smaller fragments that will be addressed in the intranet grid.

Batch jobs are executed in certain planned cycles. In the course of a working day is very difficult to find a suitable time for their work. The basic and performance-demanding tasks, represent in particular the following algorithms.

- *Production Planning (ERP).* Production Planning algorithm, is a computer-based inventory and production management system designed for scheduling of the production.
- *Product Cost (PCC).* Calculation of the production costs, based on knowledge of the product structure (BOM) and costs of the components entering into the product.
- *Low-level Code (LLC).* The low-level code defines the lowest level of usage of a material in all tree product structures. Low-level code determines planning direction.

The concept of the grid must also be prepared for a different type of tasks. Main features of the proposal are described below:

- Independent of the hardware platforms and operating system.
- Independent of the database platform.
- Easy implementation of the client.
- Easy definition of another type of job.
- The possibility of solving a number of different types of tasks at once.

The grid is intended to be used only in the Corporate Intranet. This means that the security policy of the application may be reduced. It is assumed, that all attacks outside of the firewall will be detected.

Based on the requirements described above has been chosen Java RMI technology. The Java RMI [1–3] for this type of application is well customized and is primarily intended to provide Java objects to the use on the remote client. RMI is designed so that the RMI client (master) uses the services provided by RMI server that is in the role of "slave". For the grid solution is necessary to reverse the roles. In this case, the client is in the role of "slave" and executes tasks that distributes the server services. Original consideration was the possibility to use the CORBA/IIOP technology. CORBA allows you to run objects remotely, that are not primarily written in the Java. Due to the high level of support of Java on the server side and the client was CORBA/IIOP technology rejected.

The second reason for the Java RMI using, is a communication interface that is easier than in the CORBA technology [4, 5]. The disadvantage of Java RMI is that the remote objects can be written only in Java language.

3 The Basic Principle of the Grid

Above the data system planning and management are defined SQL triggers [6, 7], which capture the events with an impact to change the data consistency (Fig. 1). For example, such a change is a process of opening the production charge. This will disturb the balance in the planning database. Batch job returns balance between customer demands, product inventory, and production levels.

This cannot be done after every change in the database, because amount of the production orders, which can be open daily, is a large number. Planning tasks can be run in a selective (reschedules only those items where changes have occurred), or regenerative mode, where the entire database rescheduled. However, both modes are challenging for a time computing.

However, both modes are challenging for computer time and power processing.

Intranet grid (as a substitute for batch jobs), reschedules the specific item immediately at the time when the change occurred. This means that the database is permanently in a consistent state.

The event is captured by a trigger and recorded in the grid scheduler. Scheduler sends the task to the client, which is available as the first. The client task calculates and returns the result to the planning database (Fig. 2).

Disadvantage of the tasks that are solved in the grid, is a higher load of network traffic. Client grid is similar in this case as the "fat client". This is a consequence of the remote objects calling. Due to the fact that part of the computing load is transferred to the client grid has a positive effect to optimize performance on the main server.

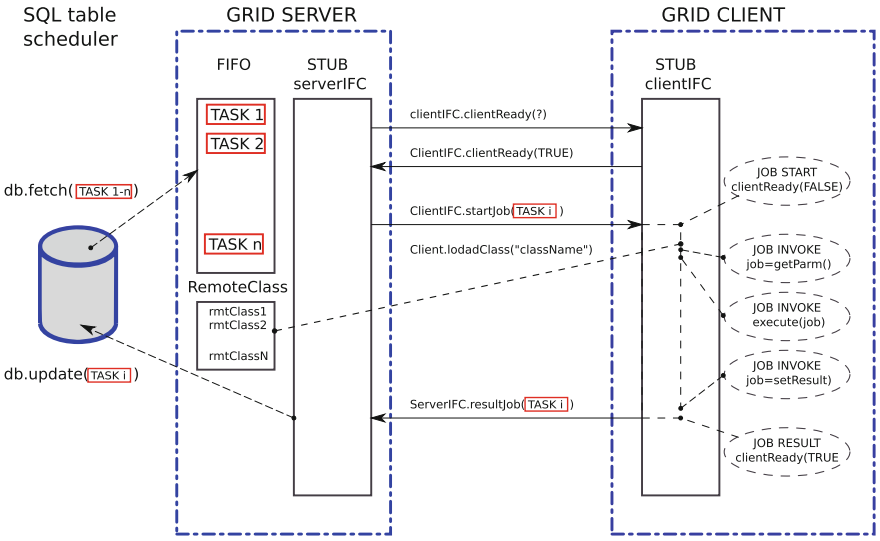


Fig. 1 Logical diagram of the grid for optimizing the server load in the production planning area

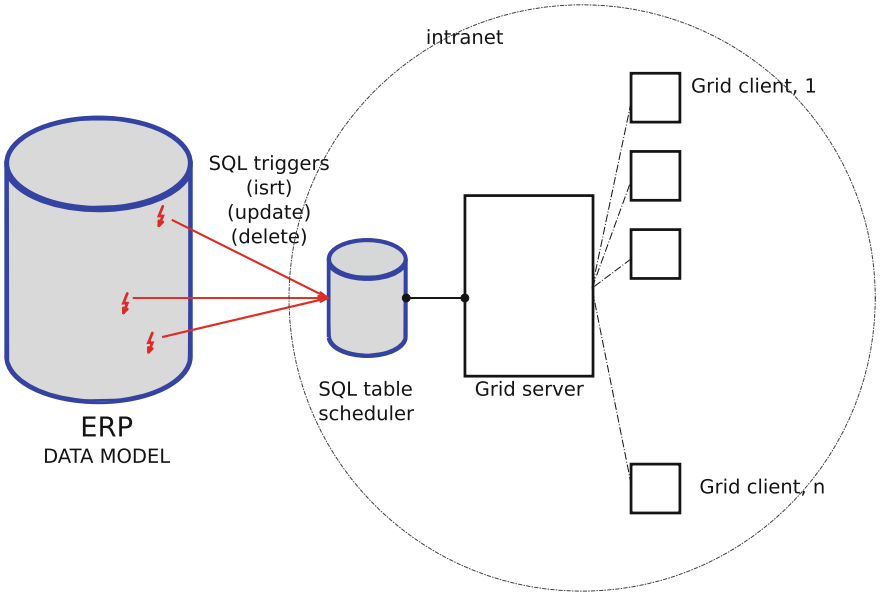


Fig. 2 Schematic diagram for the grid infrastructure components

4 Architecture for Distributed Objects

Grid Server. Maintains a current list of clients connected to the network and ensures distribution of tasks using the Task Scheduler (Fig. 2). Apache Commons DBCP project is used to the database connection [5]. This project optimizes and increases permeability of operations *ConnectionDB.open()* and *ConnectionDB.close()*. Grid server application is primarily designed to consume as little as possible processing load and system resources of the main server [8]

Job Scheduler. Is designed as a FIFO queue to determine the tasks for clients. After emptying the queues are loaded other tasks. Task Scheduler try to optimize system resource at the lowest level, while trying to maintain the optimal computing load across the grid system [9, 10]

Strategy of the task schedulers is follows:

- Optimum performance is controlled by the speed loop. Speed loop has a feedback control depending on CPU load. If a hardware load is growing, the loop schedulers slows down.
- A strategy for assigning tasks is managed based on their priorities. Priority is dependent on the type of the job. For example, LLC task must be solved sooner than the MRP. Therefore, calculations of low-level codes have priority over the other tasks.
- Each client has the same priority. The concept of client grid is designed so that you can process multiple jobs at the same time. Number of parallel tasks depends on the performance of the hardware client.

Grid Client. Is ready for the multi-threaded support. The motivation for this approach is the fact, that in the case of a small number of clients currently connected to the grid should be able to enter multiple tasks to one client at a time.

Allocation strategy of threads is depending on the performance of the client side hardware. If a client has a low power, scheduler provide a maximum of two tasks at once to him. The maximum number of threads is limited to 4 above.

The client uses a project Apache Commons DBCP for connectivity to the database. Communicates with the server using the interface serverIFC. ServerIFC is responsible for the definition of a common interface for the server and the clients. This interface includes functions for client management, task allocation and saving the results.

5 Hardware Comparsion

For comparsion the load of the master server for the production planning and the end workstations, three were selected categories of personal computers from different areas of applications (office applications, programmer and design department).

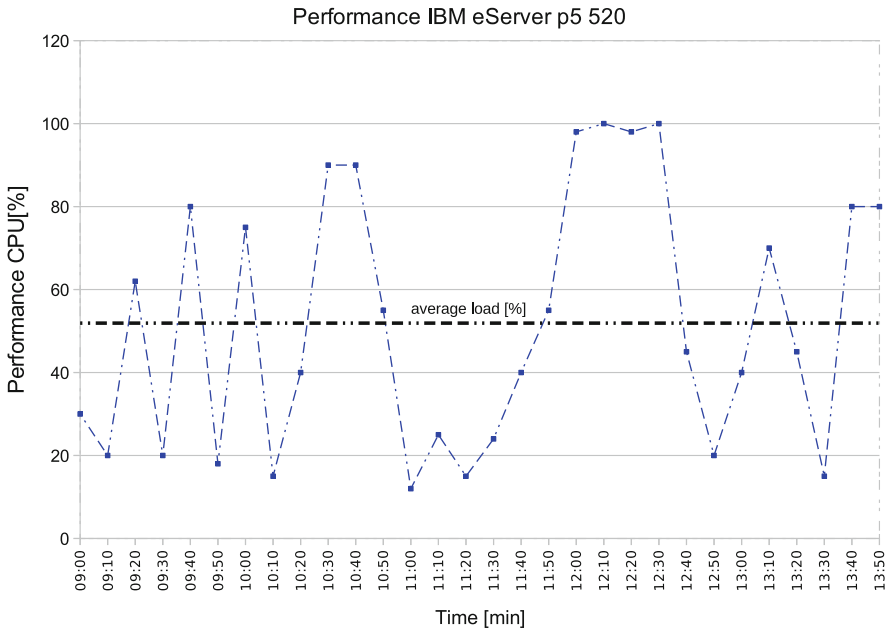


Fig. 3 The IBM eServer 520 load in the time window 6 h. Maximum load at startup batch jobs

From the results is evident disproportion in load of the master server and personal computers.

Server		
		IBM eServer p5 520
		64-bit POWER5 technology
		Concurrent users 300–500
Workstation		
M001		Fujitsu Siemens CELSIUS M460
	Memory	8 GB
	CPU	Intel(R) Pentium(R) 4 CPU 2.80 GHz
Personal computer		
M002		Fujitsu Siemens CELSIUS M420
	Memory	1.96 GB
	CPU	Intel(R) Pentium(R) 4 CPU 2.66 GHz
Personal computer		
M003		Celsius W360 Fujitsu Siemens 32 bits
	Memory	2 GB RAM
	CPU	Intel(R) CPU E 6750 @ 2.66 GHz 32 bits

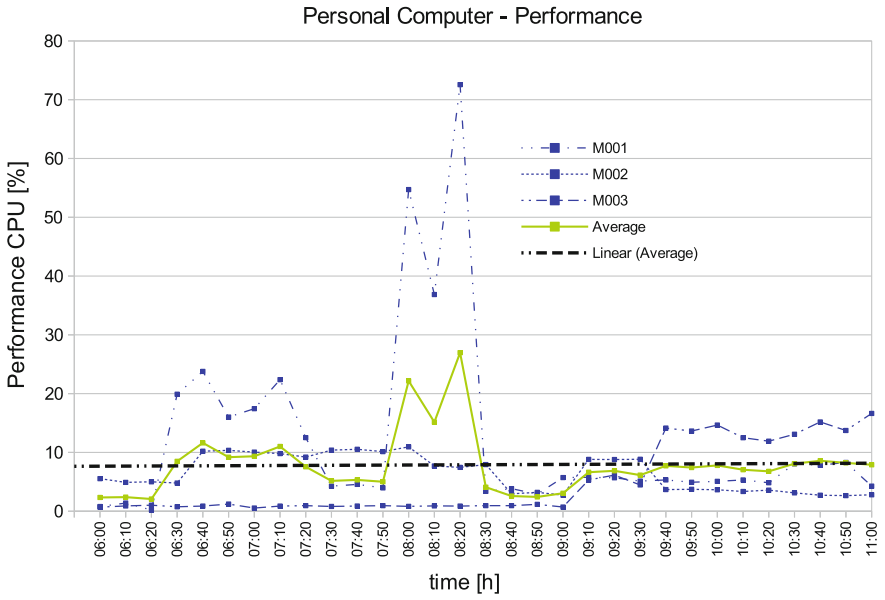


Fig. 4 The desktop load in the time window 6 h

The graph shows power reserve on the personal computers. On the server side is load much higher (Fig. 4). When we consider into account the fact that the server load to 100 % (Fig. 3) extend a poor response for all users. Part of the load who is translated to the grid has a positive impact on the whole system.

It means that transfer of the load on the server computer on the satellite will significantly increase the performance of the entire system despite the fact higher load the network infrastructure above-mentioned. Figure 4 confirms the well-known rule that not more than 10 % average of personal computers total load is used.

6 Conclusion

The aim was to obtain the optimal distribution of the computing performance of that part of the tasks to be sent to the grid. In practice, it is a batch job which requires processing power. The advantage of this concept is a better layout of computational performance and the on-line consistency of data structures in the ERP system. The disadvantage is of course, higher network communication loads.

The grid is primarily designed for handling the tasks mentioned above, which do not require high demands on managing input/output parameters. The input is usually information about the item (Item Number) or production batch (Order Number), and the output is the result, stored on the database in the ERP system.

For solving other tasks (such as extensive scientific and technical calculations), the design of the grid must be complemented by the following functionality.

- The input/output parameters are transmitted in XML or binary structures.
- Remote objects forwarded in the binary form using a database BLOB object.

The advantage of this approach is the easy definition of the task and the return of the result using SQL data structures. Current database systems have good support for managing large data structures [7]. Unfortunately these extensions (especially XML) are not fully portable. In practice, this problem has been resolved so that the XML structures are stored in gzip format in binary data types. Independence from the database is advantageous for use in the corporate sector, which uses a wide portfolio of database systems.

Another part of the research deals with the principles of the Grid scheduler. The aim is to design a scheduler such as a removable module optimized for certain types of the tasks.

The future plan for Intranet Grid development is the use of graphics cards for parallel computing on the client side. In this case, we assume the use of projects such as jocl.org (Java bindings for OpenCL) [11] and jcuda.org (Java binding for CUDA) [12]. The goal is to deploy this technology for Hardware-in-the-loop simulation in development on CNC work centres.

References

1. ORACLE, Java RMI. http://docs.oracle.com/javase/6/docs/technotes/guides/rmi/index.html
2. ORACLE, Java Remote Method Invocation: Distributed Computing for Java. http://www.oracle.com/technetwork/java/javase/tech/index-jsp-138781.html
3. Baclawski, K.: Java RMI Tutorial Northeastern University. http://www.eg.bucknell.edu/%20cs379/DistributedSystems/rmi_tut.html
4. Information technology: Object Management Group Common Object Request Broker Architecture (CORBA), Interfaces, ISO/IEC 19500-1:2012(E) Date: April 2012
5. Chappell, D.: The Trouble with CORBA. http://www.davidchappell.com/articles/article_Trouble_CORBA.html (1998)
6. Bedoya, H., Cruz, F., Lema, D., Singkorapoon, S.: Stored Procedures, Triggers, and User-Defined Functions on DB2 Universal Database for iSeries. IBM RedBook (2006)
7. PostgreSQL 9.2.2 Documentation, Copyright 1996–2012. The PostgreSQL Global Development Group
8. Apache Commons, Database connection pooling services. http://commons.apache.org/
9. Dong, F., Akl, S. G.: Scheduling Algorithms for Grid Computing: State of the Art and Open Problems. School of Computing, Queen's University Kingston, Ontario (2006)
10. Jacob, B., Brown, M., Fukui, K., Trivedi N.: Introduction to Grid Computing. IBM RedBook (2005)
11. jocl.org, Java bindings for OpenCL. http://www.jocl.org/
12. jcuda.org, Java bindings for CUDA. http://www.jcuda.org/

Discovering Cheating in Moodle Formative Quizzes

Jan Genci

Abstract Introduction of modern information technologies in educational process provides new opportunities for students and teachers. However, apart from indisputable contributions, modern information technologies and its broad usage bring forth some new challenges. This paper presents an approach which we use to detect cheating during formative assessment in Moodle environment. We describe process of obtaining data from Moodle backup archive, its transformation and, consequently, its evaluation. Later we show what possibilities the evaluated data give us to identify potential cheaters. Some ideas about enhancement of cheating detection process are also discussed.

Keywords e-Learning · Assessment · Formative assessment · Moodle · Moodle quiz · Quiz statistics

1 Introduction

Central European higher education area underwent enormous changes during last 15–20 years. We shifted from elite to mass or, may be, even universal higher education system [1, 2]. According [3]: "The number of students in public… and private colleges… increased from 62,103 in 1990 to 196,886 in 2006, which is about three-fold increase". Taking in account the fact, that number of young people in the age 18–24 (higher education age) is about 400 thousands, it is clear that Slovak higher education system could by classified (according Trow [1] and Prudký [2]—15 % of relevant population) as elite in the 1990. During approx. 15 years, till 2006, it shifted to mass (50 %) system and if we take in account

J. Genci (✉)
Department of Computers and Informatics, Technical University of Košice, Košice, Slovakia
e-mail: genci@tuke.sk

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_40,

Slovak students, who study in Czech Republic (some sources claim it is about 20 thousands students)—we have the courage to say so—it has shifted even to a universal higher education system.

Trow and Prudký claim, that transition period is specific by attempts to run the new system using old methods. It seems to us, the claim is true not only regarding management methods of universities, faculties and departments, but is valid even when dealing with methods used in the education process. We have discussed this topic in more details in [4]. Just shortly to summarize content of [4]—average intellectual potential of students is reduced from around IQ 125 to IQ 105; correlated internal motivation to study and to reveal new dimensions of cognition is lowered as well. Students' self-motivation seems to be very low. Moreover, cheating frequency in our region was historically quite high. Advent of mobile devices, cameras and any other technological achievement together with the wide spread of social networks cause enormous increase of cheating. That was the reason we decided to try and find some new approaches—supported by information technologies—to reduce problems which arose in front of us, higher education teachers.

The new approach we implemented was formative assessment combined with strict application of credit transfer system—we oblige students to earn their credits step-by-step during the term by set of activities strictly defined at the beginning of the term (which was not usual in the past and is still not usual in all of our courses). The goal of the formative assessment was to make students to prepare for seminars or laboratory works by filling open items tests relevant to related topics. Students were required to read textbooks and lab manuals to find answers relevant to questions specified in the quiz item. However, because of limited resources and some other reasons, we decided not to vary quiz items. Of course, students practically immediately published correct answers for all quiz items at their social network. At the first stage, because of open access social network site, we were able to identify students who published questions (based on open items approach) and ask them not to do it in the future. Later, students closed their network for public and we decided to run statistics related to time characteristics tied with each student's quiz time profile.

2 Related Works

As already said, cheating is wide spread and even discussed problem [5]. Plenty of algorithms [6] and even more or less complex systems [7] were proposed to discover cheaters. However, the most attention is devoted to detecting cheating in text based documents (essays, programming languages source code files). Quite high amount of papers can be found dealing with cheating during knowledge tests, but there is not so many papers which deal with approaches, how to discover it. We were able to find the only one [8].

Moodle Learning Management System is widely used in an academic and even non-academic environment. Its Quiz Module seems to us to be one of the best designed testing systems in the open software community. Besides the functionality required to prepare quiz items and quizzes itself the module incorporates very well designed and implemented statistics (see Moodle documentation: Quiz statistics calculations), together with explanation how it can be used for quiz evaluation (Moodle documentation, sections: Quiz report statistics, Quiz statistics report). Together with various blogs, it seems to us to be the only source of relevant information regarding detection of cheating in knowledge tests.

3 Formative Assessment

According to [9], formative assessment is "a range of formal and informal assessment procedures employed by teachers during the learning process in order to modify teaching and learning activities to improve student attainment". According [10]: "Assessment should be designed to teach, not just to measure …"

In our case we tried to address very low motivation of our students regarding preparation for seminars in the Operating Systems subject. Seminars of the course are devoted to the topics oriented to System Programming in the UNIX/Linux environment—starting from file manipulation and device handling, continued by process management and inter-process communication (pipes, signals, shared memory and semaphores) and ended with network communication using TCP/IP and TCP/UDP protocols (sockets). Our students have at their disposal a detailed lab manual for each seminar, which should guide them through the particular topic as they revise. For each topic we designed a set of open answer quiz items based on study materials (lab manual, LINUX man pages, books) which cover relevant parts of the topic we work on during seminar. Students are required to complete quiz 'at home' during the week before seminar. They are not limited nor as to the way how to complete it or the time when to start and when to proceed; the only limitation is the deadline of quiz, because quiz has to be filled in until midnight of the day before the seminar. It means that students can proceed with quiz in accordance with their preparation for the topic. Every quiz item is provided in the adaptive mode without or with very small penalty, so student can try to find the correct answer typing the answer several times. Because of logical sequence of terms, concepts and principles of quiz items are not shuffled. We require 75 % for passing the quiz.

Because the items in the Moodle LMS are processed by string matching, open answer items are 'sensitive' to exact answer—which requires that all relevant synonyms are marked as correct. In Slovak language the whole process is complicated by diacritics (answer can be provided with or without diacritics). We overcome this by monitoring students' answers in the quiz items statistics and adding relevant new answers as correct answers (however, this was an issue mainly at the first stage of the project).

4 Description of a Problem

When we started with formative tests, students realized very quickly that the test is common for all students and some of them published all answers at their social network webpage. Just to be honest, we have to say that the discussion regarding particular items was, then, welcomed by us and we did not limit it.

Because of the open nature of the quiz items, to find a person who published answers was quite easy, based on the set of answers. Moreover, students who were cheating, did not realize, that they can be identified based on time spend to complete test (some of them completed test in several minutes). Of course, students very quickly realized it and started to interrupt the quiz for several hours, even days (what was fully legal, according to rules defined by us) in between quiz items.

At that time we realized/noticed time information, which is associated with every student's attempt. Every answer, send to Moodle, is saved and timestamped. That means that in the systems there is a detailed history of activities carried out by the student during the quiz completion. We were limited only by fact, that in some quizzes items were grouped by 5 or 10 per page and students were able to send them to the system at once—all 5/10 items were timestamped with the same value (which was also a bit suspicious, because more convenient way seems to be to deal with each item independently).

5 Getting Data

Moodle supports several relation database management systems. The first idea was to explore the relation scheme to identify relevant relations and data elements, which allowed the gathering of required data. However, this approach required authenticated access to the database, which was usually granted to administrator only and is beyond the rights granted to ordinary Moodle user. After short investigation we discovered, that Backup functionality, provided by Moodle to each course administrator is completely sufficient for our purposes. The backup data is provided as an XML file with all data specified by the course administrator during the backup procedure. That means that we can get all data required for detailed analysis of the quiz—data about users (Fig. 1), data about quizzes (Fig. 2), data about items (Fig. 3) and data about answers provided by students (Fig. 4).

The data we obtained in the XML file was transformed to ordinary text file into CSV (comma separated values) format using XSL transformation (Fig. 5), which was used as input to SQLite database. That gives us possibility to manipulate the data using standard SQL statements.

Last step in the process of obtaining data was to generate CSV files which serve as input to EXCEL, where we made data evaluation. This process was

Fig. 1 XML file—user data

```
<USER>
  <ID>3594</ID>
  <AUTH>ldap</AUTH>
  <CONFIRMED>1</CONFIRMED>
  <USERNAME>xxx@hotmail.com</USERNAME>
  <FIRSTNAME>Tomas</FIRSTNAME>
  <LASTNAME>Mihal</LASTNAME>
  ...
</USER>
```

Fig. 2 XML file—quiz data

```
<MODULES>
  <MOD>
  <ID>625</ID>
  <MODTYPE>quiz</MODTYPE>
  <NAME>2013-Test 3rd week</NAME>
  <TIMEOPEN>1361175000</TIMEOPEN>
...
  <ATTEMPTS_NUMBER> 1 </ATTEMPTS_NUMBER>
...
```

Fig. 3 XML file—item data

```
  <QUESTION>
    <ID>3111</ID>
    <PARENT>0</PARENT>
    <NAME>Superblock in OS Unix
    </NAME>
    <QUESTIONTEXT>What data  contains super-
block in OS Unix? </QUESTIONTEXT>
   <QUESTIONTEXTFORMAT>1</QUESTIONTEXTFORMAT>
   <IMAGE></IMAGE>
   <QTYPE>multichoice</QTYPE>
  ...
  </QUESTION>
```

accomplished by simple BASH scripts in the Linux environment, with an access to SQLite database and some additional processing using program 'awk'.

A definitive data model we used in a process of data evaluation is presented at Fig. 6.

6 Data Evaluation

We prepared several outputs, from which the most interesting seems to be the data ordered by students and timestamp (Fig. 7). Then we calculated "offset" for each

Fig. 4 XML file—answer data

```
<ATTEMPTS>
   <ATTEMPT>
   <ID>72431</ID>
   <UNIQUEID>72431</UNIQUEID>
   <USERID>9516</USERID>
   <ATTEMPTNUM>1</ATTEMPTNUM>
   <SUMGRADES>65.1</SUMGRADES>
   <TIMESTART>1361182737</TIMESTART>
   <TIMEFINISH>1361623146</TIMEFINISH>
...
  <STATES>
    <STATE>
      <ID>2825275</ID>
      <QUESTION>15518
        </QUESTION>
      <SEQ_NUMBER>0
        </SEQ_NUMBER>
      <ANSWER>duplication</ANSWER>
      <TIMESTAMP>1361182737</TIMESTAMP>
      <EVENT>0</EVENT>
      <GRADE>0</GRADE>
      <RAW_GRADE>0</RAW_GRADE>
      <PENALTY>0</PENALTY>
    </STATE>
```

```
<?xml version="1.0"
encoding="UTF-8"?>
<xsl:stylesheet xmlns:xsl=
"http://www.w3.org/1999/XSL/Transform" version="1.0">
 <xsl:template match="/">
  <xsl:for-each select="MOODLE_BACKUP/…">
    <xsl:value-of select="ID"/>,
    <xsl:value-of select="FIRSTNAME"/>,
    <xsl:value-of select="LASTNAME"/>
    <xsl:text>&#xa;</xsl:text>
  </xsl:for-each>
 </xsl:template>
</xsl:stylesheet>
```

Fig. 5 XSL transformation—data about students

timestamp from the beginning of the test (column K) and the difference in the time between consequent activities (column L)—in this case, just for first attempt on every item (col. I).

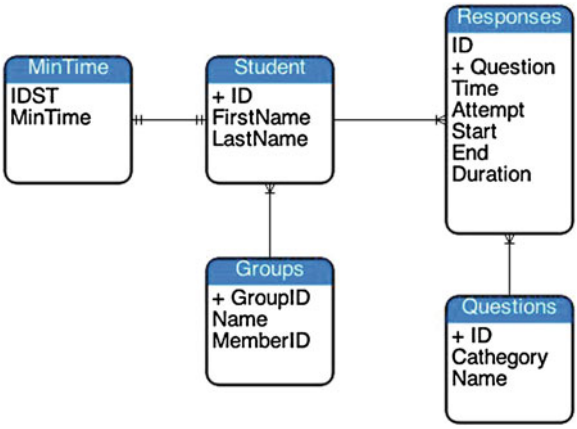


Fig. 6 Data model used for further data exploration

E	F	G	H	I	J	K	L
IDQ	CategQ	NameQ	TMPSTM	Attem	Duration	TM delta	TM diff.
15516	01-Sluzby-All	UND-01-02	1361813562	1	6435	167	167
15517	01-Sluzby-All	UND-01-03	1361813583	1	6435	188	21
15518	01-Sluzby-All	UND-01-04	1361813600	1	6435	205	17
15519	01-Sluzby-All	UND-01-05	1361813619	1	6435	224	19
15520	01-Sluzby-All	UND-01-06	1361813884	1	6435	489	259
15521	01-Sluzby-All	UND-01-07	1361814062	1	6435	667	178
15522	01-Sluzby-All	UND-01-08	1361814316	1	6435	921	248
15524	01-Sluzby-All	UND-01-10	1361814402	1	6435	1007	86
15525	01-Sluzby-All	UND-01-11	1361814432	1	6435	1037	30
15526	01-Sluzby-All	UND-01-12	1361814571	1	6435	1176	81

Fig. 7 Timestamps data of particular student and consequent derived data

Figure 7 provides random values in column L for the particular student. However, in the Fig. 8 (data for another student) we can see interesting values. The fourth value in column L—39468—means more than 10 h break in the test (which is absolutely OK). Subsequent values—block of zeroes—means, that student filled in 9 answers during 59 s, what means 6 s per answer in average—for read the question, find the answer in the supported materials and fill it in the Moodle. Students with such data pattern were put on the list of suspicious students.

Another interesting statistics seems to be the number of attempts on each item in the quiz (Figs. 9 and 10). Some students' data shows randomness (Fig. 9), on the other side, we identified student(s), who entered (almost) each open answer question on the first attempt correctly (Fig. 10, extreme case; value 2 in the graph is a result of implicit Moodle additional write operation per item to database for quiz completion).

F	G	H	I	J	K	L
CategQ	NameQ	TMPSTM	Attem	Duration	TM delta	TM diff.
)2-Subory-porozume	Post-Q01-001	1361911487	1	42594	728	32
)2-Subory-porozume	Post-Q01-002	1361911893	1	42594	1134	406
)2-Subory-porozume	Post-Q01-002	1361912791	1	42594	2032	898
)2-Subory-porozume	Post-Q01-002	1361952259	1	42594	41500	39468
)2-Subory-porozume	ost-Q01-003-F	1361952259	1	42594	41500	0
)2-Subory-porozume	ost-Q01-004-F	1361952259	1	42594	41500	0
)2-Subory-porozume	Post-Q03_001	1361952259	1	42594	41500	0
)2-Subory-porozume	ost-Q04-1_1(	1361952259	1	42594	41500	0
)2-Subory-porozume	ost-Q09-1_0!	1361952259	1	42594	41500	0
)2-Subory-porozume	ost-Q10-V2-0(	1361952259	1	42594	41500	0
Week-02-Subory-II	FL2-01	1361952259	1	42594	41500	0
Week-02-Subory-II	FL2-02	1361952259	1	42594	41500	0
Week-02-Subory-II	FL2-03	1361952318	1	42594	41559	59
Week-02-Subory-II	FL2-04	1361952318	1	42594	41559	0
Week-02-Subory-II	FL2-05	1361952318	1	42594	41559	0
Week-02-Subory-II	FL2-06	1361952399	1	42594	41640	81
Week-02-Subory-II	FL2-07	1361952399	1	42594	41640	0
Week-02-Subory-II	FL2-08	1361952399	1	42594	41640	0
Week-02-Subory-II	FL2-09	1361952399	1	42594	41640	0
Week-02-Subory-II	FL2-10	1361952399	1	42594	41640	0
Week-02-Subory-II	FL2-11	1361952399	1	42594	41640	0
Week-02-Subory-II	FL2-12	1361952399	1	42594	41640	0
Week-02-Subory-II	FL2-13	1361952470	1	42594	41711	71

Fig. 8 Suspicious data (column L)

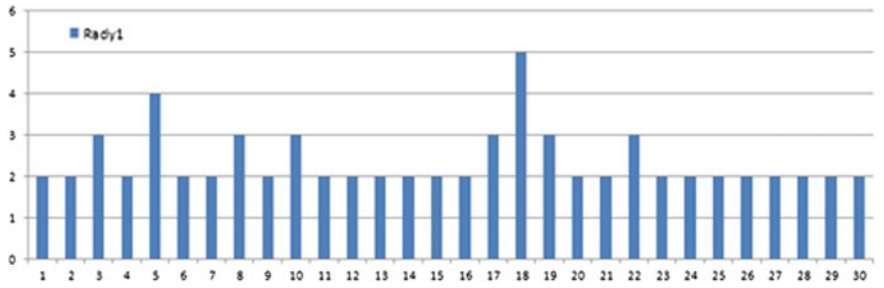


Fig. 9 Number of attempts per quiz item (question) (*x* axis—questions, *y* axis—number of attempts)

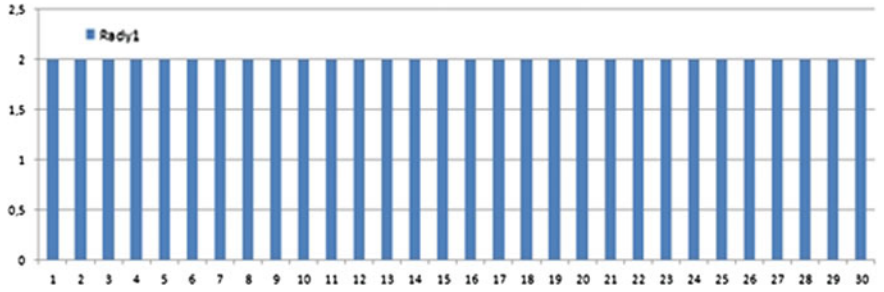


Fig. 10 Number of attempts per quiz item—suspicious data (*x* axis—questions, *y* axis—number of attempts)

7 Conclusion and Future Work

Presented approach was applied for all formative tests in the Course of Operating Systems. Suspicious students were notified about suspicion and they were given chance to defend themselves. At the end, nobody of them used the presented chance, although initially some of them claimed that he/she did not cheat.

We are aware of some drawbacks of presented approach. First of all its manual evaluation—instructor has to browse data manually (which is not so complicated for second approach—number of attempts for item, but quite laborious for first one—typically, there is tens of thousands records in the database for each quiz). Moreover, we revealed, that some students, identified as suspicious in one approach, did not fall in the suspicious group in the second approach.

In the future, we plan to introduce a more advanced processing of data, which should provide us with automatic categorization of students—cheater/non-cheater—what completely removes necessity of manual investigation of data.

Acknowledgments This work has been supported by the KEGA grant 062TUKE-4/2013, granted by Cultural and Education Grant Agency (KEGA), Ministry of Education of Slovak Republic.

References

1. Trow, M.: Problems in the transition from elite to mass higher education. Carnegie Commission on Higher Education, Berkeley (1973)
2. Prudký, L., Pabian, P., Šima, K.: České vysoké školství: na cestě od elitního k univerzálnímu vzdělávání 1989–2009, p. 162. Grada Publishing a.s, Prague (2010). (in Czech)
3. National Report.: Social and economic conditions of students at colleges and universities. Slovak Republic, National report, Bratislava (2007) (In Slovak)
4. Genči, J.: Possibilities to solve some of the Slovak higher education problems using information technologies. In: ICETA 2012—10th IEEE International Conference on Emerging eLearning Technologies and Applications, Stará Lesná, SR, pp. 117–120 (2012)
5. Maurer, H., Kappe, F., Zaka, B.: Plagiarism: a survey. J. Univers. Comput. Sci. **12**(8), 1050–1084 (2006)
6. Mozgovoy, M.: Enhancing computer-aided plagiarism detection. Dissertation, Department of Computer Science and Statistics, University of Joensuu Joensuu, Finland (2007)
7. Lancaster, T., Culwin, F.: Classifications of plagiarism detection engines. In: e-J. Italics **4**(2) (2005)
8. Matos, R., Torrão, S., Vieira, T.: Moodlewatcher: detection and prevention of fraud when using Moodle quizzes. In: Proceedings of INTED2012 (2012)
9. Crooks, T.: The validity of formative assessments. In: British Educational Research Association Annual Conference, University of Leeds, Sept 13–15, 2001 (2001)
10. Fox-Turnbull, W.: The influences of teacher knowledge and authentic formative assessment on student learning in technology education. Int. J. Technol. Des. Educ. **16**, 53–77 (2006)

Mobile Video Quality Assessment: A Current Challenge for Combined Metrics

Krzysztof Okarma

Abstract Rapid development of mobile devices such as smartphones and tablets causes the growing interest in video transmission and display dedicated for mobile devices. Considering the typical distortions introduced mainly by video compression and transmission errors, their influence on the perceived video quality is not necessarily very similar to subjective evaluation of still images or videos presented using typical computers equipped with monitors. Therefore, there is a need of verification of usefulness of known image and video quality metrics for this purpose together with recently proposed combined metrics leading to highly linear correlation with subjective quality evaluations. In this paper some results of such verifications conducted using LIVE Mobile Video Quality Database as well as results of optimisation of proposed combined metric are presented. Obtained results are superior in comparison to other known metrics applied using frame-by-frame approach.

Keywords Video quality assessment · Combined metrics · Mobile video

1 Introduction

Image and video quality assessment has become one of the most relevant fields of research related to computer vision and image analysis in recent years. Such rapid growth of interest is caused by several reasons related to the availability of digital cameras and mobile equipment as well as the development of many new image and video processing methods including lossy compression and transmission

K. Okarma (✉)
Faculty of Electrical Engineering, Department of Signal Processing and Multimedia Engineering, West Pomeranian University of Technology, Szczecin, 26 Kwietnia 10, 71-126 Szczecin, Poland
e-mail: okarma@zut.edu.pl

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_41,

methods, especially using wireless networks. Since a comparison of each new image or video compression or transmission method with existing solutions requires a reliable verification of their impact on the image or video quality, the necessity of development new objective image quality metrics has become obvious and quite urgent.

One of the most desirable features of objective metrics is their universality considered as independence on the image contents leading to the same results for various images subjected to the same type and amount of distortions as well as the sensitivity to various types of distortions introduced by noise, lossy compression, transmission errors, blur etc. Another feature is the time efficiency conditioning the possibilities of real-time implementation, especially important for video files. Nevertheless, the most relevant issue is the correlation of results of automatic assessment with subjective quality evaluations.

For the validation of such correspondence several image and video quality assessment databases have been delivered by various research groups during recent years. They contain various number of images or video sequences subjected to different types of contamination together with results of subjective experiments expressed as Mean Opinion Scores (MOS) or Differential MOS (DMOS) values. Some of those datasets, commonly accepted by the image processing community, have become an unofficial standard for verification of objective metrics' performance, typically using Pearson Linear Correlation Coefficient (PCC) and two rank-order correlation coefficients: Spearman (SROCC) and Kendall (KROCC). As additional measures the Outlier Ratio and Root Mean Square Error (RMSE) can also be used.

Unfortunately, most of the metrics described later do not utilise any colour information as well as motion vectors, so the most typical approach for the video quality assessment purposes is the use of frame-by-frame approach considering only the luminance information. Although most of the image quality assessment databases contain colour images, they do not contain any files contaminated by colour specific distortions. From this point of view colour image and video quality assessment still remains an open field of research [9, 10].

Recently, a growing interest in mobile devices such as smartphones and tablets can be observed and therefore image and video quality plays an important role also in mobile visual communication solutions. However, for such devices, both observation conditions as well as distortion types are specific, so the metrics developed and verified using typical datasets may be not well correlated with subjective perception of images and videos presented using mobile devices. In order to allow the verification of existing metrics and the development of some new ones, the LIVE Mobile Video Quality Database has been made available by the researchers from Laboratory for Image and Video Engineering being a part of Texas University at Austin [7, 8]. The dataset contains 200 video files with modelled video distortions in wireless networks with heavy traffic obtained from 10 High-Definition reference files. The database incorporates both well-known distortion types such as compression and wireless packet-loss, and dynamically varying distortions changing in time—frame-freezes and temporally varying

compression ratios. In the subjective experiments over 50 persons have been involved and a half of video files have been additionally assessed for tablet screen. The detailed description of the dataset can be found in the article [6].

2 Development of Objective Image Quality Metrics

As for many years a typical approach to image quality assessment has been the use of pixel-based Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR) or similar metrics, the idea of sliding window approach, first proposed in 2002 together with Universal Image Quality Index (UIQI) [17], can be considered as a milestone in this field of research. Further modifications of this approach have led to probably the most popular Structural Similarity (SSIM) metric [18] and its numerous modifications, including its multi-scale version [19]. Some of the other metrics, considered in this paper, which are based on the similar idea are Gradient SSIM [2] and Quality Index based on Local Variance (QILV) [1].

An interesting approach, quite similar to SSIM, but based on the similarity of features has been proposed in the papers [21, 22]. The metric proposed as the first one, called Riesz-transform based Feature Similarity metric (RFSIM), is based on the assumption that the most important regions by means of perceived image quality are edges and their neighbourhood. They can be easily extracted e.g. using well-known Canny filter leading to the map of key locations for which human observers are more sensitive to low level features (edges, corners, lines or zero-crossings). The authors of the paper [22] have proposed the application of the Riesz transform to the nearest neighbourhood of the detected edges in order to use the first and second order Riesz transform coefficients as five masked image features. Then, the local comparison of two feature maps for the reference and distorted images (or video frames) can be made using the following formula (quite similar to the SSIM):

$$d_i(x,y) = \frac{2 \cdot f_i(x,y) \cdot g_i(x,y) + C}{f_i^2(x,y) + g_i^2(x,y) + C} \tag{1}$$

where f and g denote the two compared images, $i = 1 \ldots 5$ is the feature number and C is a small stabilizing constant value.

Assuming that M is the binary mask obtained as the result of the edge filtering, the overall value of the RFSIM index is expressed as

$$\mathrm{RFSIM} = \prod_{i=1}^{5} \frac{\sum_x \sum_y d_i(x,y) \cdot M(x,y)}{\sum_x \sum_y M(x,y)}. \tag{2}$$

The metric proposed next year by the same group of researchers [21], known as Feature Similarity (FSIM), is based on the combination of the phase congruency

(PC) and gradient magnitude (G) information. Both these factors are also calculated locally leading to a local similarity index expressed as

$$S(x,y) = \left(\frac{2 \cdot PC_1(x,y) \cdot PC_2(x,y) + T_{PC}}{PC_1^2(x,y) + PC_2^2(x,y) + T_{PC}}\right)^{\alpha} \cdot \left(\frac{2 \cdot G_1(x,y) \cdot G_2(x,y) + T_G}{G_1^2(x,y) + G_2^2(x,y) + T_G}\right)^{\beta} \tag{3}$$

where T_{PC} and T_G are small stabilizing constants.

For simplicity the importance exponents for both elements are set to $\alpha = \beta = 1$ but they can also be optimized leading to Weighted Feature Similarity with slightly better rank-order correlations for most datasets as shown in the paper [14]. The phase congruency can be determined using the method described by Liu and Laganière [4] and the values of the gradient magnitude can be obtained using the Scharr convolution filter (but for simplicity Prewitt or Sobel masks can also be applied).

The overall FSIM index is obtained by averaging the local similarity values according to the following formula:

$$\text{FSIM} = \frac{\sum_x \sum_y S(x,y) \cdot PC_m(x,y)}{\sum_x \sum_y PC_m(x,y)} \tag{4}$$

where $PC_m(x, y) = \max(PC_1(x, y), PC_2(x, y))$ denotes the higher of the two local phase congruency values calculated for two images being compared.

Some other approaches to full-reference image quality assessment which can lead to high correlation with human perception of various image distortions are based on the Singular Value Decomposition as well as the information theory. The examples of the first approach are the MSVD [16] and R-SVD [5] metrics as well as some other attempts, also for colour images [20]. A representative example of the second mentioned group of metrics can be the Visual Information Fidelity (VIF) metric presented in the article [15].

Apart from full-reference metrics, there are also some no-reference (also known as "blind") and reduced-reference ones. The "blind" metrics are of great interest due to their ability to predict the image distortions without the knowledge of the reference (undistorted) image. However, in some applications where one has the access to the original image, such approach is rather not applied because of the worse performance and lower universality in comparison to full-reference metrics. Most of the known no-reference metrics are sensitive only to one or two common types of distortions such as blur or JPEG compression artifacts. Similar situation takes place for reduced-reference metrics which require only the partial knowledge of some chosen reference image features. Nevertheless, the development of no-reference metrics dedicated for mobile video quality assessment is one of the future challenges, not considered in this paper.

3 Combined Metrics

Considering the usefulness of the objective metrics for practical applications, it is obvious that an ideal metric should be linearly correlated with subjective scores without the necessity of any nonlinear mapping as suggested in some of the publications. From this point of view, the most relevant performance measure for all objective image quality assessment methods is Pearson Linear Correlation Coefficient calculated for raw quality scores and MOS or DMOS values. As the PCC is used for measuring the prediction accuracy and rank-order correlations (SROCC and KROCC) are utilised only for measuring the prediction monotonicity, the main goal of research in this direction is related to the optimization of PCC value for raw scores using available datasets.

Unfortunately, due to some nonlinearities of the Human Visual System, none of the single metrics allow obtaining high linear correlations of the raw scores with MOS/DMOS values. For this reason a nonlinear combinations of some metrics of different types can be used in order to achieve high values of the PCC. The first such attempt has been presented in the paper [11]—proposed Combined Quality Metric (CQM) has been defined as the weighted product of MS-SSIM, VIF and R-SVD metrics with exponent values optimized for the largest image quality assessment database (namely Tampere Image Database) leading to PCC = 0.86. Further modification by replacing the R-SVD by the colour version of the FSIM metric (FSIMc) leading to the Combined Image Similarity Index (CISI) presented in [13] has increased this value to 0.8752 for the same dataset. Due to a different character of distortions present in the video sequences, a similar approach has also been applied for video files [12] using the video sequences from the LIVE Wireless Video Quality Assessment Database for the optimization purposes. Nevertheless, due to the limited number of distortion types, this dataset has been replaced by LIVE Mobile Video Quality Assessment Database and therefore it is currently unavailable at the LIVE website.

The usefulness of the approach based on the combination of some metrics has been validated also by some other researchers as evidenced by the results presented in one of the recently published articles [3] related to the application of multi-method fusion. Nevertheless, it is worth noticing that the authors of this paper still apply the nonlinear mapping using the logistic function before calculation of the PCC values in order to increase the performance.

4 Proposed Metric and Discussion of Experimental Results

As the main goal of this paper is related to the optimization of the combined quality metric dedicated for mobile devices, the first experiments have been related to the calculation of some known full-reference metrics for the videos included in the LIVE Mobile dataset. Unfortunately, assuming the use of the frame-by-frame

Table 1 Obtained results of the correlation coefficients for various metrics

Subset	156 Files mobile		71 Files tablet		71 Files mobile	
Metric	PCC	SROCC	PCC	SROCC	PCC	SROCC
SSIM	0.6990	0.6990	0.6563	0.6663	0.5717	0.5510
MS-SSIM	0.6156	0.7962	0.5642	0.8163	0.5896	0.7823
GSSIM	0.6491	0.6450	0.6700	0.6298	0.5572	0.4941
QILV	0.6342	0.8380	0.5535	0.7944	0.6150	0.8371
R-SVD	0.2982	0.4006	0.1833	0.2444	0.3029	0.3944
VIF	0.8707	0.8416	0.8688	0.8250	0.8912	0.8700
FSIM	0.7183	0.8552	0.7044	0.8240	0.7642	0.8973
RFSIM	**0.8756**	**0.8625**	**0.9399**	**0.9059**	**0.9429**	**0.9332**
WFSIM	0.5415	0.7892	0.4701	0.6831	0.5490	0.7855
CQM	0.8250	0.8338	0.8424	0.8546	0.8514	0.8574
CISI	0.8410	**0.8641**	0.8410	0.8580	0.8820	0.9155
CVQM	0.8441	0.8569	0.8432	0.8612	0.8905	0.9273
CM1	0.8992	0.8688	0.9318	0.8716	0.9401	0.9064
CM2	0.9026	0.8713	**0.9443**	0.8843	**0.9494**	0.9167
CM3	0.9090	0.8776	0.9288	0.8768	0.9454	0.9101
CM4	**0.9127**	**0.8844**	0.9130	0.8620	0.9363	0.9026

approach, totally 30 files from 200 should be rejected from all experiments. As these files were distorted by frame freezes in stored video which did not result in the loss of a video segment, the relation between the frame numbers was lost and longer video sequences were created. Since the videos maintain temporal continuity after the freeze, the application of frame-by-frame approach for such type of distortions does not make sense. It is worth noticing that frame freeze in live videos are also available in the LIVE Mobile dataset and those sequences have been used in all calculations.

As the DMOS values for mobile study obtained from 36 subjects have been provided for all video files and only 100 files have been used for the tablet study with 17 subjects, the optimization has been conducted using DMOS values from the mobile study while subjective scores obtained for tablet study have been used for the verification. The distortion types considered in the database and in the calculations are: lossy compression (4 layers), wireless channel packet-loss (for 4 compressed sequences), frame-freezes (only for real time live video delivery), rate adaptation (3 scenarios) and temporal dynamics (5 scenarios). Such 17 distortions (without 3 frame-freezes for stored videos) for 10 reference video gives 170 files. Nevertheless, for some files due to the presence of many dark areas (zeros), some singularities can be obtained during the calculations of e.g. VIF metric, leading to improper results influencing also the optimization results. After rejection of those files, the 156 remaining video sequences have been using for calculations (with 71 of them with available additional DMOS values from tablet study).

After the calculations of single objective metrics and their correlations with subjective scores, similar calculations have been conducted for combined metrics

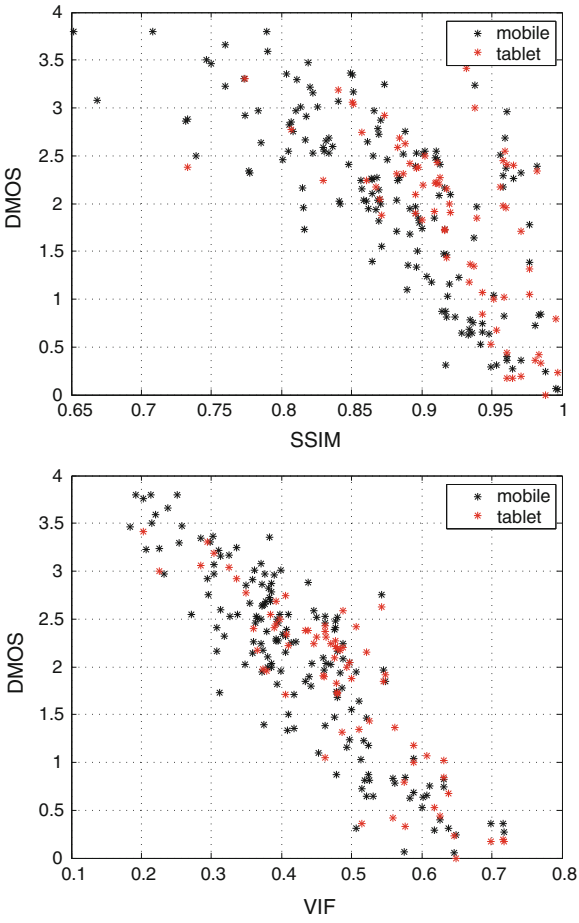


Fig. 1 Scatter plot of the SSIM and VIF metrics for LIVE Mobile Video Quality Assessment Database

(CQM, CISI and CVQM). Finally, the optimization procedure have been applied using MATLAB environment with Image Processing Toolbox using *fminsearch* and *fminunc* functions. The best results for two metrics have been obtained for the combination of RFSIM and VIF metrics. Further increase of correlation coefficients have been obtained after adding the QILV and MS-SSIM metrics. The details of the obtained PCC and SROCC values are presented in Table 1 and the obtained scatter plots illustrating the linearity of the correspondence between various metrics and DMOS values are shown in Figs. 1 and 2.

The proposed variants of the combined metrics with PCC and SROCC values presented in Table 1 are defined as: CM1—unweighted product of RFSIM and VIF, CM2—weighted product of RFSIM and VIF (exponents: 1.325 and 0.6), CM3—weighted product of RFSIM, VIF and QILV (exponents: 1.0291, 0.6826 and 1.2618), CM4—weighted product of RFSIM, VIF, QILV and MS-SSIM (exponents: 1.1406, 0.6728, 2.4683 and −2.3569).

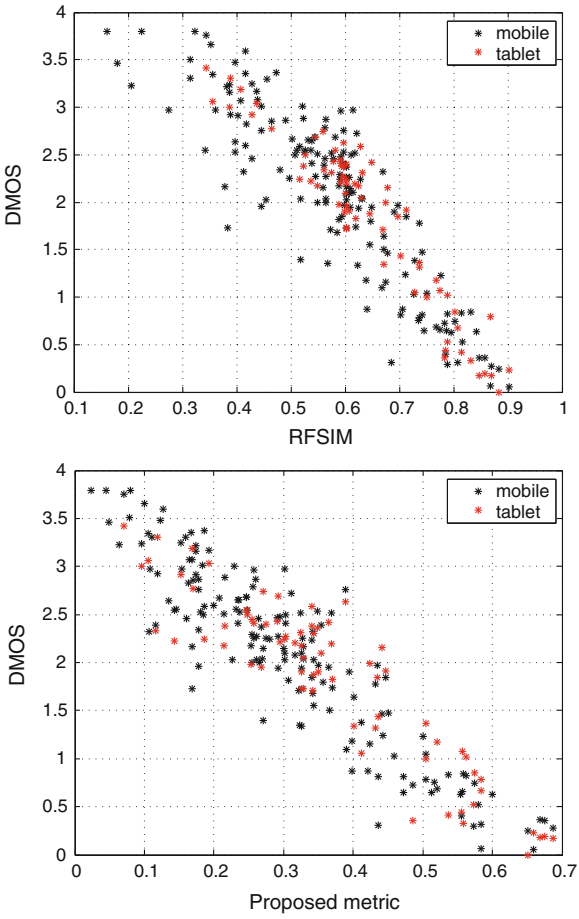


Fig. 2 Scatter plot of the RFSIM and combined metric CM4 for LIVE Mobile Video Quality Assessment Database

5 Conclusions

Analysing the results presented in Table 1 and the scatter plots presented in Figs. 1 and 2 the advantages of using the combined metrics proposed in this paper can be easily noticed. Looking at the scatter plot obtained for the CM4 metric strongly linear relationship between the metric and DMOS values can be observed which corresponds to highest PCC values obtained for 156 files used in the calculations. Obtained results prove that the application of the combined metrics is an interesting solution leading to high correlation of objective metrics with subjective evaluations of video quality, also for mobile devices.

Nevertheless, some issues e.g. related to using the colour information or application of some other methods than simple frame-by-frame approach are still an open field of research. Another limitation of development of some new metrics is related to the availability of video databases containing the subjective scores

(currently only LIVE Mobile Video Quality Assessment Database is available for mobile videos) as well as the computational effort necessary for verification and optimization of metrics. However, presented results may be a good starting point for further research related to mobile video quality assessment, also for some other researchers.

References

1. Aja-Fernandez, S., Estepar, R.S.J., Alberola-Lopez, C., Westiniu, C.F.: Image quality assessment based on local variance. In: 28th IEEE Annual International Conference on Engineering in Medicine and Biology Society (EMBS), pp 4815–4818. New York City (2006)
2. Chen, G.H., Yang, C.L., Xie, S.L.: Gradient-based structural similarity for image quality assessment. In: Proceedings of 13th IEEE International Conference on Image Processing (ICIP), pp. 2929–2932. Atlanta, Georgia (2006)
3. Liu, T.J., Lin, W., Kuo, C.C.J.: Image quality assessment using multi-method fusion. IEEE Trans. Image Process. **22**(5), 1793–1807 (2013)
4. Liu, Z., Laganière, R.: Phase congruence measurement for image similarity assessment. Pattern Recogn. Lett. **28**(1), 166–172 (2007)
5. Mansouri, A., Mahmoudi-Aznaveh, A., Torkamani-Azar, F., Jahanshahi, J.: Image quality assessment using the Singular Value Decomposition theorem. Opt. Rev. **16**(2), 49–53 (2009)
6. Moorthy, A.K., Choi, L.K., Bovik, A.C., de Veciana, G.: Video quality assessment on mobile devices: subjective, behavioral and objective studies. IEEE J. Sel. Top. Sign. Proces. **6**(6), 652–671 (2012)
7. Moorthy, A.K., Choi, L.K., de Veciana, G., Bovik, A.C.: Mobile Video Quality Assessment Database. In: IEEE ICC Workshop on Realizing Advanced Video Optimized Wireless Networks, pp. 7055–7059. Ottawa, Canada (2012)
8. Moorthy, A.K., Choi, L.K., de Veciana, G., Bovik, A.C.: Subjective analysis of video quality on mobile devices. In: 6th International Workshop on Video Processing and Quality Metrics for Consumer Electronics (VPQM), pp. 63–68. Scottsdale, Arizona (2012)
9. Okarma, K.: Colour image quality assessment using structural similarity index and Singular Value Decomposition. In: Bolc, L., Kulikowski, J., Wojciechowski, K. (eds.) ICCVG 2008. LNCS, vol. 5337, pp. 55–65. Springer, Heidelberg (2009)
10. Okarma, K.: Two-dimensional windowing in the structural similarity index for the colour image quality assessment. In: Jiang, X., Petkov, N. (eds.) CAIP 2009. LNCS, vol. 5702, pp. 501–508. Springer, Heidelberg (2009)
11. Okarma, K.: Combined full-reference image quality metric linearly correlated with subjective assessment. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L., Zurada, J. (eds.) ICAISC 2010. LNCS, vol. 6113, pp. 539–546. Springer, Heidelberg (2010)
12. Okarma, K.: Video quality assessment using the combined full-reference approach. In: Choraś, R.S. (ed.) IP&C 2010. AISC, vol. 84, pp. 51–58. Springer, Heidelberg (2010)
13. Okarma, K.: Combined image similarity index. Opt. Rev. **19**(5), 249–254 (2012)
14. Okarma, K.: Weighted feature similarity—a nonlinear combination of gradient and phase congruency for full-reference image quality assessment. In: Choraś, R.S. (ed.) IP&C 2012. AISC, vol. 184, pp. 187–194. Springer, Heidelberg (2013)
15. Sheikh, H., Bovik, A.C.: Image information and visual quality. IEEE Trans. Image Process. **15**(2), 430–444 (2006)
16. Shnayderman, A., Gusev, A., Eskicioglu, A.: An SVD-based gray-scale image quality measure for local and global assessment. IEEE Trans. Image Process. **15**(2), 422–429 (2006)

17. Wang, Z., Bovik, A.C.: A universal image quality index. IEEE Signal Process. Lett. **9**(3), 81–84 (2002)
18. Wang, Z., Bovik, A.C., Sheikh, H., Simoncelli, E.: Image quality assessment: from error measurement to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
19. Wang, Z., Simoncelli, E., Bovik, A.C.: Multi-scale structural similarity for image quality assessment. In: 37th IEEE Asilomar Conference on Signals, Systems and Computers. Pacific Grove, California (2003)
20. Zhang, F., Li, J., Chen, G., Man, J.: Assessment of color video quality with Singular Value Decomposition of complex matrix. In: 5th International Conference on Information Assurance and Security, pp. 103–106. Xi'an, China (2009)
21. Zhang, L., Zhang, L., Mou, X., Zhang, D.: FSIM: a feature similarity index for image quality assessment. IEEE Trans. Image Process. **20**(8), 2378–2386 (2011)
22. Zhang, L., Zhang, L., Mou, X.: RFSIM: a feature based image quality assessment metric using Riesz transforms. In: 17th IEEE International Conference on Image Processing, pp. 321–324. Hong Kong, China (2010)

Face Extraction from Image with Weak Cascade Classifier

Václav Žáček, Jaroslav Žáček and Eva Volná

Abstract The aim of this paper is to propose an artificial vision-based face detection approach, which could be primarily used in robotics. Three main problems arise from this expectation. The first one is the computation time of the whole process. The second one is the quality of the input information due to a camera with low resolution. The third one is the robustness of the involved techniques regarding the implementation. The paper discusses all three problems in the first part and introduces the Haar Cascade theory. The second part of the paper proposes a new noise reduction approach to improve detection result mostly in eyes and mouth area. Next part of the paper shows experimental results and finds the best threshold parameter to minimize overlapping areas. The last part explains advantages of the proposed technique.

Keywords Weak cascade classifiers · Face extraction · Face recognition system · Open CV

1 Introduction

Despite the fact that the face detection technology has significantly improved and can be successfully used in real-time image processing under appropriate (constrained) situations, face detection still requires a lot of improvement to increase

V. Žáček (✉) · J. Žáček · E. Volná
Department of Informatics and Computers, University of Ostrava,
30 dubna 22, 70103 Ostrava, Czech Republic
e-mail: vaclav.zacek@gmail.com

J. Žáček
e-mail: jaroslav.zacek@osu.cz

E. Volná
e-mail: eva.volna@osu.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_42,

robustness. There are some problems in unconstrained tasks, e.g. multiple viewpoints, uneven illumination, multiple facial expressions, and facial details. New technologies, such as nVidia CUDA, accelerate computations in all state-of-the-art algorithms. Thus all new proposed algorithms should be parallel ready.

Face recognition applications are divided into two broad categories in terms of user's cooperation: cooperative user scenarios and non-cooperative user scenarios. Cooperative user scenarios are mostly used in applications, such as computer login, physical access control or e-passport, where the user is willing to be cooperative by presenting his/her face in a proper way (for example, in a frontal position with neutral expression and eyes open) in order to be granted access or a privilege. The distance between the face and the camera is usually less than 1 m. Non-cooperative user scenarios are used in surveillance applications, where the user is unaware of being identified, such as computer access control (e.g. watch list identification). The distance between the face and the camera is more than 1 m. By this definition, face detection in the case of robotic vision would be classified as a cooperative scenario. If we simplify it, we can expect that a person will be looking directly into the camera, or at least towards the place where they will expect the camera lens. An example of this behavior can be observed by using Samsung phones. If you look at the display, the phone identifies the face and does not dim the screen brightness.

Face detection is a visual pattern recognition problem, where a face is represented as a three-dimensional object. This object represents a subject in varying illumination, position, expression, and other factors need to be identified based on the acquired images.

2 Processing Time

To ensure useful real time face recognition, we have to define and analyze processing time. Processing time depends on two main factors.

The first factor is the algorithm which is used for face detection. The speed of the algorithm depends on the type of the information which it processes. The images can be processed in two main ways. The first one depends on the color of the image. This means that a sort of a color filter is used to segment the image to get areas with a specific color. Having obtained the segments, geometric approach is needed to obtain approximate location of the face. The second one works with wave representation, thus you need to create a sort of a kernel to process the pixels surrounding the area which is selected at the time. Both of these approaches consist of a large number of repetitive calculations on the image.

The second factor is implementation of the algorithm. If we want to drastically reduce the time needed for processing of the image, we have to use massive parallelization of the computation process. We are able to meet this condition because we do not modify the image itself in the computation process. Therefore, we can create multiple threads working on the same instance of the image.

3 Face Detection Methods

In the early development of face detection [1–3], *geometric facial features* such as eyes, nose, mouth, and chin were explicitly used. Properties of the features and relations among them (e.g. areas, distances, angles) were used as descriptors for face recognition. Big advantages of these approaches are economy and efficiency. The main cause of the economy and efficiency of these methods is the data reduction and insensitivity to variations in illuminations and viewpoints. Therefore, the method saves computation time. On the other hand, if we use only geometric properties, the recognition becomes inaccurate in the relevance to specific face data like facial texture [4]. This is the main reason why early feature-based techniques were not effective.

Statistical learning methods are the second approach used to build face detection systems these days. Effective features are learned from a training data set and that features involve prior knowledge about faces. The appearance-based approach, such as Principal Component Analysis (PCA) [5] or Linear Discriminant Analysis (LDA) [6], has significantly moved the face recognition technology forward. This approach operates directly on raster representation (e.g. fields of pixel intensities). It extracts features in a subspace that are derived from training images.

The most significant advantage of this method is avoiding instability of manual selection and tuning in the early phase of geometric shape learning process. Statistic methods encode prior knowledge contained in the training data. Cardinality of the training set is crucial.

However, they are not able to capture fineness of face subspaces: protrusions may be blurred out and concavities may be smoothed. Therefore, useful information can be lost. Note that appearance-based methods require proper justification of the face images. These methods are typically based on eye location.

The most successful approach so far to handle the nonconvex face segmentation works with *local appearance-based features.* These features are extracted using appropriate image filters. An advantage lies in distribution of face images through local feature space, which is less affected by changes in facial appearance. Early work in this area includes local features analysis (LFA) [7] and Gabor wavelet-based features [8]. Current methods are based on a local binary pattern (LBP) [9]. There are many variants on the basic approaches like: Ordinal Features [10], Scale-Invariant Feature Transform (SIFT) [11], and Histogram of Oriented Gradients (HOG) [12]. These features are general purpose; face specific local filters are learnt from images [13]. An advantage is that a large number of local features can be generated with respect to varying parameters associated with positions, scales, and orientations of the filters. For example, more than 400,000 local appearance features can be generated if an image of size 100×100 is filtered with Gabor filters with five different scales and eight different orientations for all pixel positions.

4 Haar-Cascade Detection with OpenCV

The last part introduces state-of-the-art face detection mechanisms. However, this paper is aimed at robust detection of human faces technique, which is highly effective with regards to colored and gray scale pictures, even when the lighting conditions are not optimal. This goal has been fulfilled with the OpenCV framework thanks to GPU-accelerated computing with Haar feature-based cascade classifiers [13]. The time needed to locate a face in the picture depends on two factors: scale of the presented picture and hardware strength of the computation power (in our case a graphic card). There is a little problem with using the OpenCV framework. If NVIDIA graphic cards with CUDA are used, we do not have a problem with it. However, if we try to calculate Cascade of Classifiers with processor power, there is a problem with parallelization of the processes. OpenCV uses only one core (single thread) on the multi-core processor, whereas GPU-accelerated computing uses the graphic processing unit (GPU) together with the CPU to accelerate calculations. For recognition of a face, it is crucial to be able to locate regions with specific face features like eyes, mouth, brows and cheeks. These areas hold necessary information for robust recognition of the given face. Haar Cascades works just like a convolutional kernel mask. To explain the Haar Cascades method, we have to introduce integral images. It simplifies calculation of the sum of pixels. All possible sizes and locations of each kernel are used to calculate plenty of features. Each feature is a single value obtained by subtracting the sum of pixels under a white rectangle from the sum of pixels under a black rectangle (Fig. 1). These integral images can be computed in a very fast way (1):

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \tag{1}$$

where $ii(x, y)$ is the integral image and $i(x', y')$ is the original image [13]. Using the following pair of recurrent equations (2):

$$\begin{aligned} s(x, y) &= s(x, y-1) + i(x, y) \\ ii(x, y) &= ii(x-1, y) + s(x, y) \end{aligned} \tag{2}$$

where $s(x, y)$ is the cumulative row sum, $s(x, -1) = 0$, and $ii(-1, y) = 0$. Thanks to these equations, the integral image can be computed in one pass over the original image.

This means that the integral image is a subarea of the original image. We use these areas to calculate the differences inside the integral images with respect to the learnt cascades before. Basically, it means that we take the sum of the black areas and compare it with the sum of white areas in the classifier (Fig. 2). The sum of pixels within rectangle B can be computed with three array references. The value of the integral image at location 1 is the sum of the pixels in rectangle A. The sum within B can be then obtained as $3 + 1 - 2$.

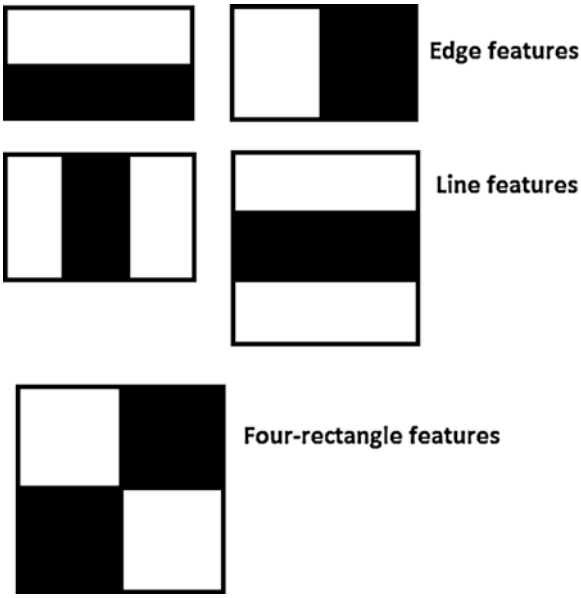


Fig. 1 Haar cascades features

The best threshold is found for each feature, which will classify the faces to positive and negative. But obviously, there will be errors or misclassifications. We select the features with the minimum error rate, which means the features that best classify the face and non-face images. The final classifier is a weighted sum of weak classifiers. It is called weak because they cannot classify the image alone, but together they form a strong classifier.

5 Noise Reduction

The principle of our face detection and feature extraction using Haar Cascades consists of three phases. The goal of the *first* phase is conversion of the image to the grayscale. Detection does not depend on the color in the image. This means that detection is possible even with black and white pictures. In the *second* phase, the system tries to localize all faces in the image with the Haar Cascades, which has been taught to recognize faces from another set of faces before. In the *third* phase, the system tries to find eyes and mouth for each detected face. The proposed sample code could be interpreted as follows:

1. Convert the picture to the gray scale image.
2. Apply the appropriate classifier which has been trained before to detect faces.

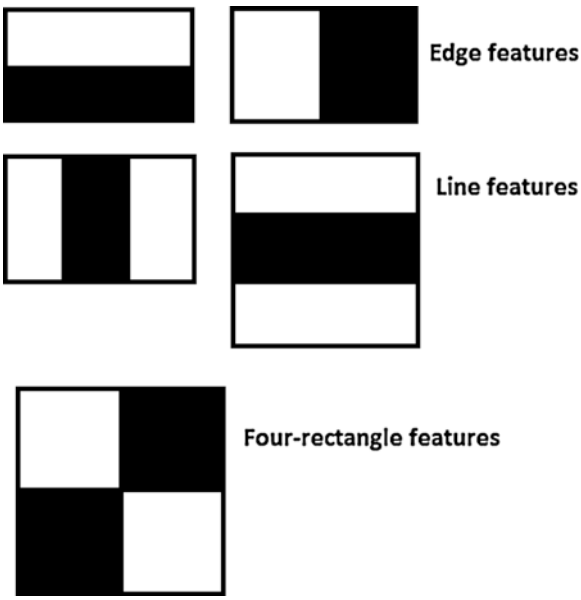


Fig. 2 Classificatory computation

3. Collect the areas with the highest probability of containing faces.
4. Try to search for eyes in each area.
5. Limit the number of eye areas with the numeral threshold for overlapping classifiers and select only ones that are located in expected areas.
6. Try to search for mouth in each area that could be a face.
7. Limit the number of mouth areas with the numeral threshold for overlapping classifiers and select only those objects that are located in expected areas.

There are free Haar Cascades for eyes and mouth in [14]. These cascades are highly effective to find these areas in most pictures. One of many problems that need to be taken into consideration is the existence of face accessories. For example, we have to be sure that the cascade will be able to detect eyes even behind the glasses. When the lighting conditions are not ideal, there are a number of false positive detections of eyes and mouth. That will be problematic in face processing later on. In Fig. 3a both nostrils were falsely classified as eyes. This happened due to the following: stronger lighting; angle from which the photo was taken; shape of the nostrils; shadow of the nostrils and their shape similar to eyes (darker areas). A similar situation happened with the mouth. Next problem was the presence of glasses because of the line of the frame and shadow in the left ear, which is similar to the line of the mouth (in a smaller scale). This problem could be

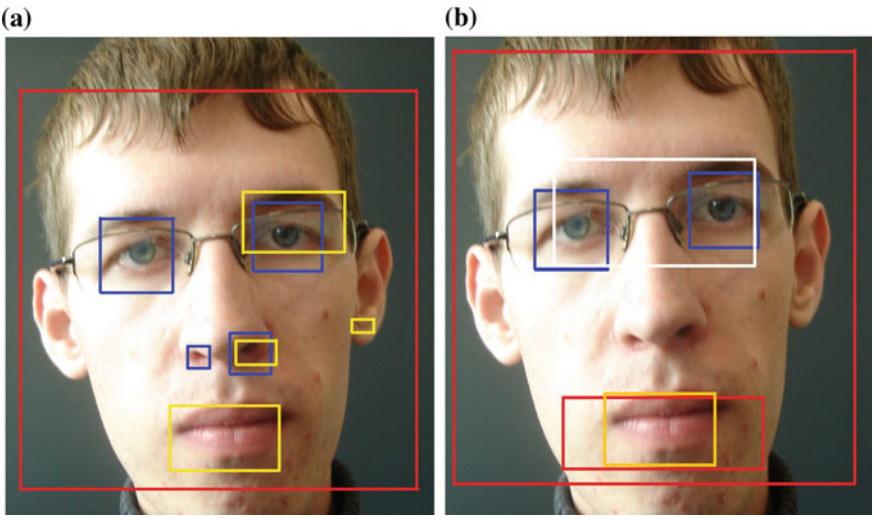


Fig. 3 Detected areas for face (*red rectangles*), eyes (*blue rectangles*), and for mouth (*yellow rectangles*). **a** Before reduction. **b** AFter reduction

avoided with selecting features that are situated in specific areas of the face, as shown in Fig. 3b.

As it can be seen in Fig. 3b, all false positive readings disappeared from selection of the detected areas. After that, we can work only with areas where eyes and mouth are identified with the most probability. False positive readings can also be reduced by specification of the threshold which represents a number of possible overlapping areas before they are marked as a possible feature. These changes lead to further reduction of false readings.

6 Experimental Results

We have tested 10 overlapping areas for faces, 15 overlapping areas for eyes, and 20 overlapping areas for mouth (further on referred to as threshold) in our experimental study. These values also reduce the number of detected features with the same center but different scale. An example of the behavior is shown in Fig. 4.

Figure 4a shows how the detection algorithm works if the number of areas is set to a lower number for eyes. Figure 4c shows reduction of the overlapping areas with a higher threshold number (number in area is set to 29). When the threshold was too high, the application wasn't able to detect eyes in most of the cases. A similar case was observed with the mouth detection as shown in Fig. 4b respectively d, where the number of overlapping areas for mouth is set to 10, respectively 20.

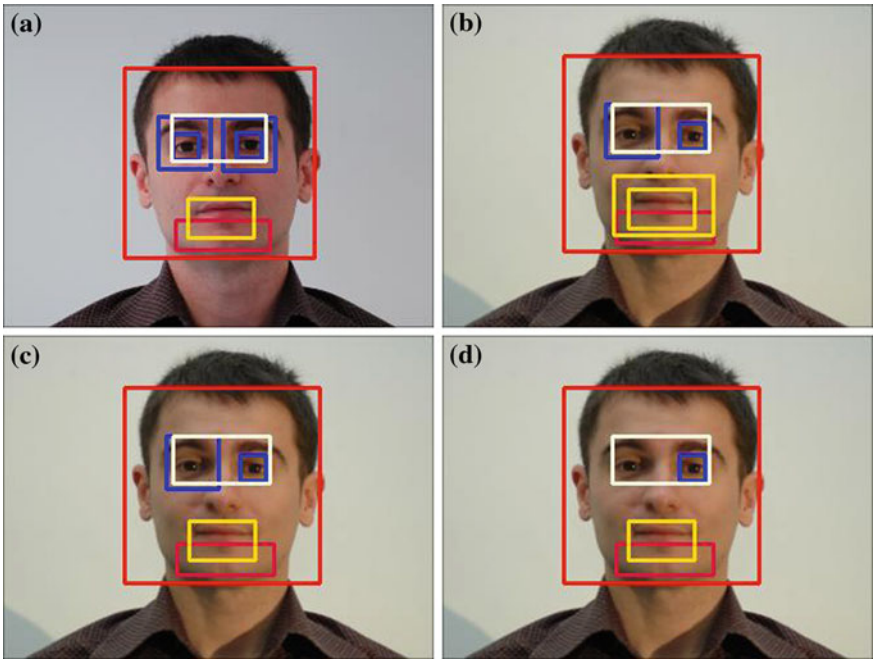


Fig. 4 Overlapping areas example: **a** overlapping of eyes **b** reduction of overlapping eyes also example of overlapping mouth **c** reduction of overlapping mouth **d** too high threshold for eyes

This test has been designed to find the weaknesses of the selected classifiers. Fifty images of the same face were used for the testing. These pictures had differences in quality. Some of them were blurry, too dark, overlighted, or contained compression errors. The experimental study was aimed at learning possible settings for Haar Cascades to obtain good performance with bad camera conditions. Continuous results are shown in Table 1. The results referred in the table as "Ability to find" is the number which we can get if we take all testing faces together and say if the classifier was able to detect at least something. This means that if the classifier has been able to detect eyes or mouth in the right area, it is considered as good detection. The percentage for eyes may seem rather low. The main cause of these low numbers is bad lighting conditions of the tested images. The weaknesses and the strong points have been highlighted.

A number of false detection was caused by bad light conditions. Therefore, the used classifiers for *eyes detection* did not find them in some cases. A summary of the acquired knowledge for the used classifiers is as follows:

- They are not able to detect closed eyes.
- They need to see the white part of the eyes (sclera).
- There is a possibility to detect eyes even when the image is blurry.
- They perform better in larger images (higher resolution).

Table 1 Experimental results of eyes detection and mouth detection

Threshold on overlapping	Eyes detection		Mouth detection	
	Number of false detections	Ability to find (%)	Number of false detections	Ability to find (%)
5	1	54	0	82
6	1	50	0	74
7	1	42	0	74
8	0	42	0	72
9	0	40	0	68
10	0	40	0	64
11	0	40	0	64
12	0	40	0	58
13	0	40	0	56
14	0	32	0	56
15	0	32	0	54
16	0	32	0	48
17	0	30	0	46
18	0	30	0	42
19	0	28	0	38
20	0	26	0	38
21	0	26	0	36
22	0	26	0	34
23	0	26	0	30
24	0	26	0	28
25	0	26	0	26
26	0	22	0	26
27	0	22	0	26
28	0	22	0	24
29	0	22	0	24
30	0	22	0	20

- They are more robust for rotation of the head.
- They have no problems when detecting eyes behind glasses. They work well even if some reflections are present in the glasses.

The performed experiments show that eye detection is possible even if we have a bad camera or lower image resolution, but we have two conditions to ensure positive detection. The first one is that eyes are open. The second one is that the sclera needs to be visible.

The test result for the *mouth detection* shows that mouth detection is more successful than eye detection in bad light conditions. However, the used classifiers have problems with rotation of the face. For accurate detection of mouth, when the face is rotated in some way, a sort of normalization of the face should be used to

compensate this disadvantage. A summary of the acquired knowledge for used cascade classifiers is as follows:

- Their performance is the same with respect to all sizes of images.
- They achieve poor results for rotated faces. They sometimes detect only a part of mouth.
- They are able to detect mouth in blurry images.

7 Conclusion

This article is focused on face detection, which could be primarily used in robotics. The proposed solution was tested on real images where two main problems have been discovered. The first one is the time of the whole processing phase. If the time of computation is too long, the application cannot be used in robotics. The second one is the robustness of the combined detector. We have proposed a new detector based on a combination to improve the detection rate on images with bad quality. Robustness of the detection is the reason why the experimental study is aimed at analyzing weak points of the proposed solution in detail.

Most robots use cameras with low resolution. Therefore, the output image has bad lightness of the captured scene and general noise.

The threshold settings for eyes and mouth are not the same. It can be observed from the experimental test results that some images needed very low threshold setting for eyes but other images needed higher because of the amount of false positive readings. The same situation can be observed with the mouth. These thresholds should be mainly tested and set on the device which will provide image capturing (for example a robot). Accuracy of eyes detection mostly depends on the lightness of the input image.

The mouth threshold value mostly depends on rotation of the face. That means we need a little higher threshold in classical frontal face detection, therefore, the number of false detections will be reduced.

Acknowledgments The research described here has been financially supported by University of Ostrava grant SGS23/PřF/2013. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

References

1. Brunelli, R., Poggio, T.: Face recognition: features versus templates. IEEE Trans. Pattern Anal. Mach. Intell. **15**(10), 1042–1052 (1993)
2. Kanade, T.: Picture processing system by computer complex and recognition of human faces. PhD thesis, Kyoto University (1973)

3. Samal, A., Iyengar, P.A.: Automatic recognition and analysis of human faces and facial expressions: a survey. Pattern Recogn. **25**, 65–77 (1992)
4. Cox, I.J., Ghosn, J., Yianilos, P.: Feature-based face recognition using mixture-distance. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 209–216 (1996)
5. Turk, M.A., Pentland, A.P.: Eigenfaces for recognition. J. Cogn. Neurosci. **3**(1), 71–86 (1991)
6. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans. Pattern Anal. Mach. Intell. **19**(7), 711–720 (1997)
7. Penev, P., Atick, J.: Local feature analysis: a general statistical theory for object representation. Neural Syst. **7**(3), 477–500 (1996)
8. Wiskott, L., Fellous, J., Kruger, N., von der Malsburg, C.: Face recognition by elastic bunch graph matching. IEEE Trans. Pattern Anal. Mach. Intell. **19**(7), 775–779 (1997)
9. Ahonen, T., Hadid, A., Pietikainen, M.: Face recognition with local binary patterns. In: Proceedings of the European Conference on Computer Vision, pp. 469–481. Prague (2004)
10. Liao, S., Lei, Z., Zhu, X., Sun, Z., Li, S.Z., Tan, T.: Face recognition using ordinal features. In: Proceedings of IAPR International Conference on Biometrics, pp. 40–46 (2006)
11. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of IEEE International Conference on Computer Vision, p. 1150. Los Alamitos, CA (1999)
12. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 886–893 (2005)
13. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, p. 511 (2001)
14. Reimondo, A., Haar Cascades.: OpenCV Swiki. Haar Cascades, OpenCV Swiki (online) 2006 (cit. 2013-09-19). Available from http://alereimondo.no-ip.org/OpenCV/34

Computer Aided Analysis of Direct Punch Force Using the Tensometric Sensor

Dora Lapkova, Michal Pluhacek and Milan Adamek

Abstract This research was focused on measuring and analyzing of the direct punch force of young adults. The main focus was on the differences between genders and among groups of participants with different level of training. In this long-term study more than 200 participants took part. The collected data were analyzed and stored for future use in research. This paper presents the results of first analysis focused on the difference in the mean maximum of direct punch force of participants in different categories.

Keywords Punch force · Tensometric sensor · Gender differences · Combat training · Data analysis

1 Introduction

The force of a hand-to-hand attack of person is usually very unpredictable. Presumably there are many affecting factors such as gender, body weight, and height. Also not in the last place the training of the attacker seems to be an important factor. Even though such knowledge would be very helpful not only for the safety forces or coroners, very little research has been done on this field. Even through intensive literature review we could not find any similar research, thus our research

D. Lapkova (✉) · M. Pluhacek · M. Adamek
Faculty of Applied Informatics, Tomas Bata University in Zlin,
Nam. T. G. Masaryka 5555, 760 01 Zlin, Czech Republic
e-mail: dlapkova@fai.utb.cz

M. Pluhacek
e-mail: pluhacek@fai.utb.cz

M. Adamek
e-mail: adamek@fai.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_43,

is aiming to fill that gap with analysis of maximum force of different striking techniques for different groups of people with the aid of microcomputer with tensometric sensor and computer data analysis.

The striking techniques are one of the basic elements of the majority of combat sports [1], martial arts [2] or combat systems [3]. In these techniques the striking energy (or impulse force) [4] is transferred thru arms, legs or head. In this paper the direct punch force is closely analyzed. The direct punch is delivered by the arm following direct line. The hitting area is a closed fist [5]. In the following experiment the punch was delivered by the back hand (see Fig. 1).

2 Experiment Goal and Participants

The main goal of the experiment was to measure the profile of direct punch force in time. The main focus was on the differences between genders and among groups of participants with different level of training.

Two research questions were defined:

1. Is the maximal direct punch force dependent on gender?
2. Is the maximal direct punch force dependent on the level of training?

The participants of the experiment were divided into several categories based on their gender and previous training in combat sports, self-defense or martial arts. Following categories were analyzed in this research: Men with no training, mid-trained, trained and self-trained; Women with no training, mid-trained and self-trained.

3 Measuring Devices

The experiment was conducted by means of tensometric sensor [6] that was placed into a leather target (punching bag). The punching bag was subsequently attached to the measuring station created from oriented strand boards (Fig. 2).

The strain gauge sensor of the pressure force, type SRK-3/V (Fig. 3) is a passive electromechanical converter which converts force to a proportional electrical signal [4].

As a mechanical-electrical converter it uses silicon resistive strain gauges because their deformation sensitivity is sixty times higher than that of the film or wire resistive strain gauges. The sensor is sized and calibrated for constant loading of 3 Kn force exerted in the axis of the sensor; nevertheless, it also endures a long-term repeated overload up to 200 % (6 Kn) in the axis of the sensor [4].

The sensor consists of a base in the shape of a short cylinder which verges into a truncated cone in its upper part. The upper base of this truncated cone is formed by a membrane with four silicon resistive strain gauges AP120-3-12 affixed on its

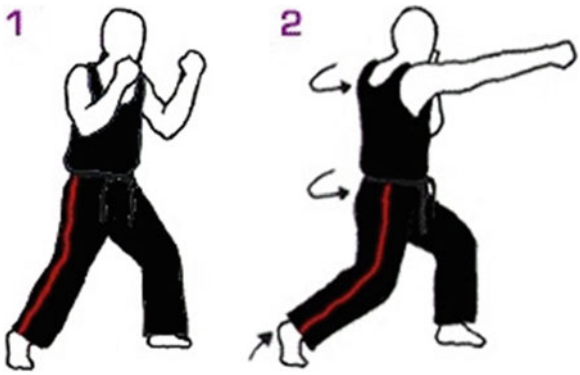


Fig. 1 Direct punch [5]

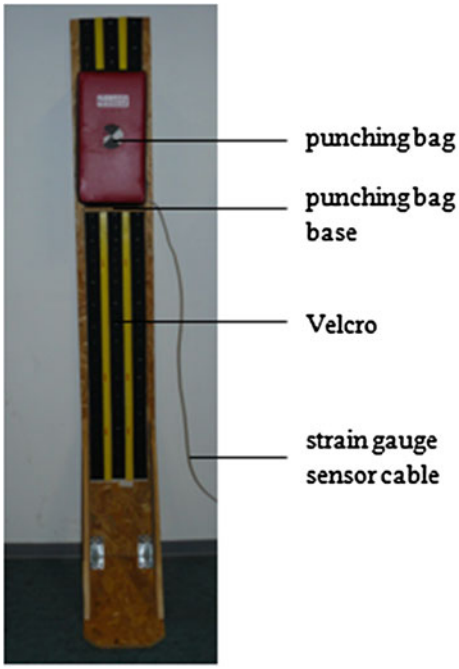


Fig. 2 Measuring station [4]

inner surface. In the middle of the inner surface of the membrane there is a junctor that connects the membrane to a measuring area in the shape of a spherical cap. All of the described parts of the sensor are made of one piece of dimensionally stable alloy treated steel [4].

The pressure force exerted on the measuring area is being transmitted to the membrane by means of the junctor and deforms it proportionally to any exerted force. At the same time the force is being transferred to four silicon resistive strain gauges fixed to the membrane by a special tensometric adhesive which converts it to an electrical resistance proportional to the deformation. The connection of the

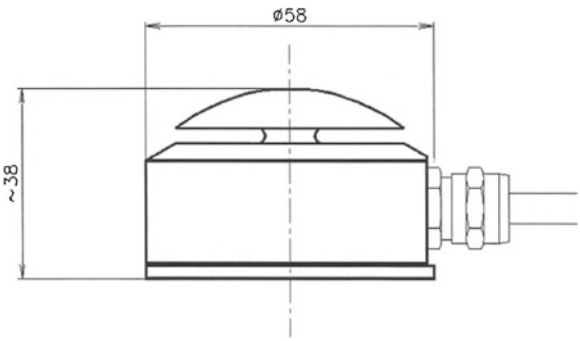


Fig. 3 Pressure force sensor (SRK-3/V) [4]

strain gauges to the Wheatstone bridge provides an effective primary compensation of the influence of temperature on the measuring system [4].

The sensor is connected to the computer, which is used for data storage, through the strain gauge. The strain gauge type TENZ2334 is an electronic appliance that converts the signals to data that is stored in memory. The core of the appliance is a single-chip microcomputer that controls all of the activities. The strain gauge sensor is connected to this appliance via four-pole connector XLR by four conductors. The number of values measured by the sensor averages around 600 measurements per second while the data is immediately stored in the memory of a device with a capacity of 512 Kb [4].

4 Experiment Setup

The total of 219 participants took part in the experiment; 198 men and 21 women. All participants were in the age from 19 to 25. Based on previous training and experience the participants were divided into following categories:

- No training—These persons have never done any combat sport, martial art or combat system. They have no theoretical knowledge of the striking technique. The technique was presented to these persons before the experiment for safety reasons.
- Mid-trained—These persons have the theoretical knowledge of striking techniques and do attend the Special physical training course for at least six months. The course is focused on self-defense and professional defense.
- Trained—These persons do attend the Special physical training course for two or more years or practice a combat sport or martial art for the same time period.
- Self-trained—These persons did practice or still do practice (for less than 2 years) some combat sport, martial art or combat system. As there is no guaranty on the quality of the training they are separated into separate category.

During the experiment each person made two strikes. During the measurement the target was positioned in such manner that the center of the tensometric sensor

was in line with the striking persoń s shoulder. That way the direct punch has the maximum velocity and force (as there is no decomposition of force or velocity into other axes). The person was made to stay at the same place for the whole experiment. Any unnecessary movement (e. g. lunge etc.) would lead to data distortion.

5 Results

The collected data were processed and analyzed in the Wolfram Mathematica environment [7]. In this study, the maximum direct punch force for each participant was derived from the collected data using computer aided analysis of the force profile. In this case the maximum on each force profile was localized. All force profiles had to be shifted on the time axis (Fig. 4) in order to visualize the mean profile (Fig. 5). The center point is the maximum of the force during the direct punch. In this section the results of each group are presented in tables (Table 1 for men and Table 2 for women). Example mean direct punch force profiles in time for mid-trained participants are depicted in Fig. 5 (men) and Fig. 6 (women).

6 Results Summary and Discussion

In the following table (Table 3) the results presented in previous section are summarized. The mean maximum direct punch force (and standard deviation) for each category is presented alongside with the total number of participants in that category. The data hint that the mean maximum direct punch force is higher for trained persons and significantly higher for men than for women.

There were certain unpredicted limitations during the experiment. For example in order of accurate measurement it is necessary for the person to hit the exact center of the sensor. That proved very difficult for non-trained participants. Some of the extremely low or high entries are due to this reason.

7 Future Research Perspectives

During this long-term research the sorted data for direct punch force profiles of differently trained men and women were collected using the described methodology. The advanced detailed analysis of collected data will be the main goal of future research. Various methods of computational intelligence will be investigated. Among the most promising the classification using neural network based classifiers, fuzzy classifiers or neuro-fuzzy classifiers. As for the classification processes preparation, the data will be preprocessed by various data mining techniques as the most dominant features need to be identified. Alternately various data filtering can also be applied.

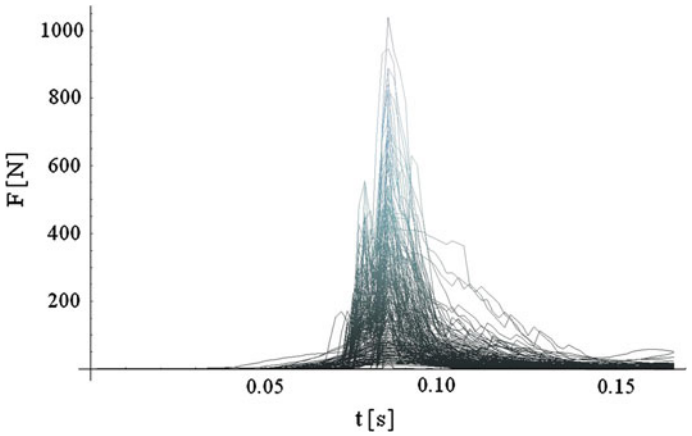


Fig. 4 Direct punch force in time (Men—mid-trained—all)

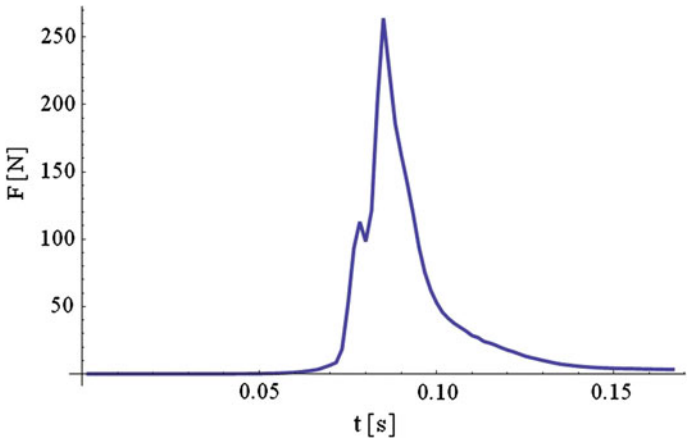


Fig. 5 Mean direct punch force in time (Men—mid-trained)

Table 1 Results overview—men

Men	Mean value (N)	Median (N)	Std. dev. (N)	Maximum (N)	Minimum (N)
No training	233.7	151.7	234.1	955	5.2
Mid-trained	264.2	212.6	220.9	1,039.9	10.9
Trained	371.6	228.3	370.3	918.9	111.1
Self-trained	273.9	196	274.6	1,284.2	8.6

Table 2 Results overview—women

Women	Mean value (N)	Median (N)	Std. dev. (N)	Maximum (N)	Minimum (N)
No training	21.8	22.6	7.1	31.9	12.8
Mid-trained	21.1	19	11.3	47.3	2.2
Self-trained	44.6	21	48.3	117	19.2

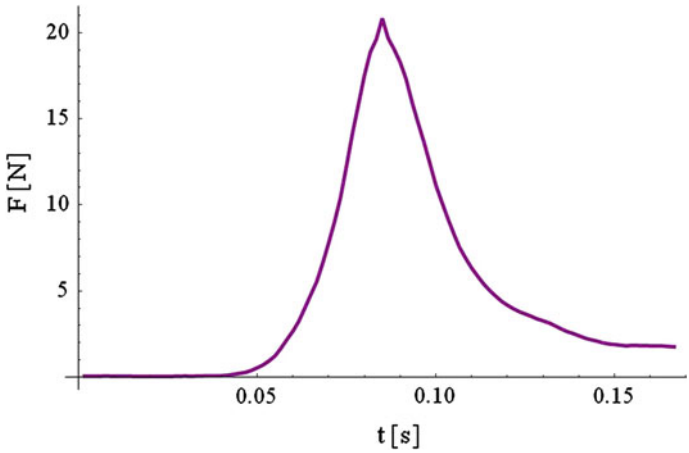


Fig. 6 Mean direct punch force in time (women—mid-trained)

Table 3 Results summary

Category of participants	N. of participants	Mean maximum force F (N)
Men—no training	90	233.7 ± 234.1
Men—mid-trained	80	264.2 ± 220.9
Men—trained	2	371.6 ± 370.3
Men—self-trained	26	273.9 ± 274.6
Women—no training	3	21.8 ± 7.1
Women—mid-trained	16	21.1 ± 11.3
Women—self-trained	2	44.6 ± 48.3

8 Conclusion

In this long-term research the direct punch force profiles of more than 200 participants were measured using tensometric sensors and complex measuring station. The results were afterwards processed and analyzed using the Wolfram Mathematica environment.

Results presented in Sect. 5 and summarized in Sect. 6 support the claim that (answering the first research question) the mean maximum direct punch force is significantly higher for men than for women. Also the profile of the force in time is different. For men it is usually much sharper profile than for women.

The answer for the second question is partly possible for men category. It seems that the mean maximum direct punch force is increasing with the increasing level of training. However it is not that case for women most likely due to very small number of participant in the group with no training.

Due to the small number of participants in some categories (especially women) it is necessary to see presented results as a first hint of possible interesting trends that need to be proved by future and larger studies. The aim of this paper is to inform about these findings and describe the methodology that was used, including the computer aided analysis of collected data. The future research will focus among others on employing advanced computer aided data analysis and data mining techniques in order to uncover hidden correlations and possible dependencies in the collected data.

Acknowledgments This research was supported by the Internal Grant Agency at TBU in Zlín, project No. IGA/FAI/2013/017 and IGA/FAI/2014/10 by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089.

References

1. Blower, G.: Boxing: Training, Skills and Techniques. Crowood, Marlborough (2007)
2. Gianino, C.G.: Physics of karate. Kinematics analysis of karate techniques by a digital movie camera. Lat.-Am. J. Phys. Educ. **4**(1), (2010)
3. Levine, D., Whitman, J.: Complete Krav Maga. Ulysses Press, Berkeley (2007)
4. Lapkova. D., Pospisilik, M., Adamek, M., Malanik, Z.: The utilisation of an impulse of force in self-defence. In: Paper presented at the XX IMEKO world congress: metrology for green growth, Busan, Republic of Korea (2012)
5. Reguli, Z.: Innovation of the bachelor's study program special education of the security forces and the master's study program Applied sport education of the security forces: Biomechanics of combat sports and martial arts. (In Czech), Available from: http://www.fsps.muni.cz/inovace-SEBS-ASEBS/elearning/biomechanika/biomechanika-upolovych-sportu (2011)
6. VTS Zlín, Tensometers, (In Czech). Available from: http://www.vtsz.cz/polovodicove-tenzometry.php (2010)
7. Wolfram Research, Inc., Mathematica, Version 9.0, Champaign, IL (2013)

Part III
Software Engineering

Application of Semantic Web and Petri Calculus in Changing Business Scenario

Diwakar Yagyasen and Manuj Darbari

Abstract The paper highlights correlation between Adaptive Business Environment and Web Semantic, Petridynamics, Adaptive Semantic Web. The Business environment use of Activity Theory and Web Semantic help in formatting the Ontology.

Keywords Semantic web · Petri calculus · Changing business scenario

1 Introduction

In dealing with dynamic aspect of business we have to be include with the elimination of older systems and tools and replace with new one. The paper focuses on developing a model which embeds micro-economic theory with ontology. The model will provide an ontology based semantic annotation to achieve interoperability [22, 23]. The main idea is to automate the process of building knowledge base [5]. The process of conversion was Activity Theory (AT) for building a taxonomy for an enterprise [15].

D. Yagyasen
PhD Programme, Babu Banarsi Das University, Lucknow, Uttar Pradesh, India
e-mail: dylucknow@gmail.com

M. Darbari (✉)
Department of Computer Science, Babu Banarsi Das University, Lucknow, Uttar Pradesh, India
e-mail: manujuma@gmail.com

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_44,

1.1 Introduction Semantic Web

"The semantic web is an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation."—Tim Burners-Lee, James Hendle, On Lassila [2].

Semantic analysis is the process of relating syntactic structures from phrases sentences, removing features specific to particular linguistic and cultural contexts. The key research areas are natural language, text and image understanding, speech, data mining and finally process mining. According to the definition given by Fensel Semantic Web can be classified into three basic categories:

(a) Information Extraction: It supports wrapping technology for uniform extraction of information.
(b) Processable Semantics: It deals with capturing information structures as well as meta-information about the nature of information.
(c) Ontologies: This is the process of converting taxonomies generated into resource description format that reflects the consensual and formal specification of the domain.

In order to deal with Semantic Heterogeneity [1, 8, 17] we have to deal with natural understanding of interchanged data. It is the ability of two or more computer systems to exchange information without changing the meaning of the information. It is classified under three basic types: single-ontology approach, multiple-ontology approach and hybrid ontology approach.

2 Literature Survey

We are inspired by the work of Deshpande [7] on Adaptive Query Engine and used his work in building business changes and its adaption with new situations. The paper [13] describes effective method of matching different strategies which are suitable for various types of tasks and contexts. Brambilla et al. [3] on exceptional handling has very well described the movement of links in Ontology where a link automatically discards the older link in case of closure of the business i.e. expired link. The paper by Howse et al. [13] on visualizing the ontology has opened new dimensions in the field of obsening the changes in the business link vis-a-vis to ontologies. Keddara [9] has extended the above work by focusing on the modalities of changes which includes the duration of change and its time frame. Sasthyive have used the concept of Activity Theory [14, 18] dealing with Engestrom Triangle.

3 Activity Theory: An Overview

"Activity Theory is a philosophical framework for studying different forms of human practices as development process, with both individual and social levels interlinked at the same time". AT is therefore committed for understanding both individual and collective aspects of human practices from a cultural and historical perspective. The ideas presented in activity Theory have their origins in Vygotskyian [22] concept of tool mediation and Leontiev's [14] notion of Activity. Vygotsky [21] originally introduced the idea that human beings interactions with their environment are not direct ones but are instead mediated through the use of tools and signs. In this paper we will be extending the concept of Engestrom [10] model known as "Activity Triangle Model" which incorporates the components like: subjects Object, Community with mediators of human activity namely Tools, rules and Division of Labour.

The 'object' component portrays the purposeful nature of human activity, which allows individuals to control their own motives and behavior when carrying out activity.

The 'subjects' components of the model portrays both the individual and collective nature of human activity through the use of tools in a social context so as to satisfy desired objectives. The subject's relationship with the object or objective of activity is mediated through the use of tools.

The 'tools' component of the model reflects the mediational aspects of human activity through the use of both physical and psychological tools. Physical tools are used to handle or manipulate objects, they therefore extend human being's abilities to achieve targeted goals and satisfy objectives. Psychological tools are used to influence behavior in one way or another.

The "Community" component represents stakeholders in a particular activity or those who share the same overall objective of an activity. The community puts the analysis of the activity being investigated into the social and cultural context of the environment in which the subject operates.

The "Rules" component highlights the fact that within a community of actors, there are bound to be rules and regulations that affect in one way or another means by which the activity is carried out. These rules may either be explicit, or implicit, for example, cultural norms that are in place within a particular community. The component of the Activity triangle model also helps to establish environmental influences and conditions in which activity is carried out.

The "Division of Labour" component reflects the allocation of responsibilities and variations in job roles and responsibilities amongst subjects involved in carrying out a particular activity within a community.

The "Activity System" consists of several sub-activities that are interconnected and united through the shared objective on which activity is focused. As a result of this inter-connectedness, disturbances or contradictions can occur within and

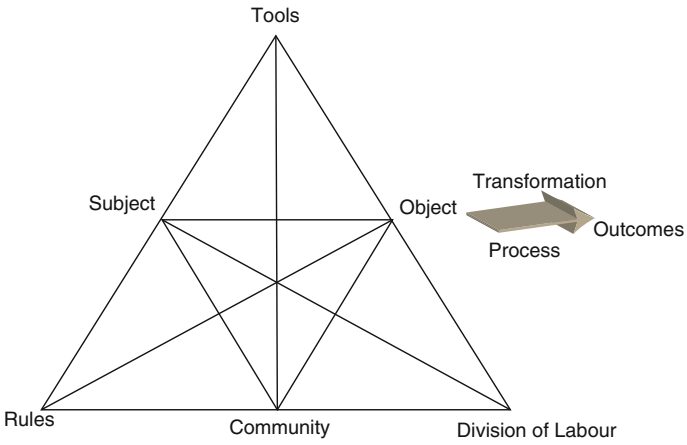


Fig. 1 Activity triangle model

between sub activities that could affect the transition of the collective activity system. The terms "contradictions" is used in AT to refer to misfits, disturbances, problems or breakdowns, that occur in an Activity System or human practices being examined (Fig. 1).

3.1 Activity Theory Dynamics

The work on implementation of supply chain coordination using Semantic Web services helped us in linking Activity Theory to business dynamics using Semantic Web. "An object is being treated as an entity which can modified and transformed by the participants of an activity." The principle of mediation plays a major role in Activity Theory. An activity is composed of various types of Artifacts (example: instruments, signs, procedure, machines, materials, laws, forms of work or organisation). Under normal condition Artifact performs the role of mediator performing a bridge between elements of Activity. The tools are used to shape the way human being interact with their context. A tool can also be used as a medium to transform the process of Object which includes material tool and thinking. The relationship between subject and community is mediated through rules similar a relationship between object and community is mediated by division of labour. Finally the division of labour categorises the role that each individual will be playing and the task it is responsible for (Fig. 2).

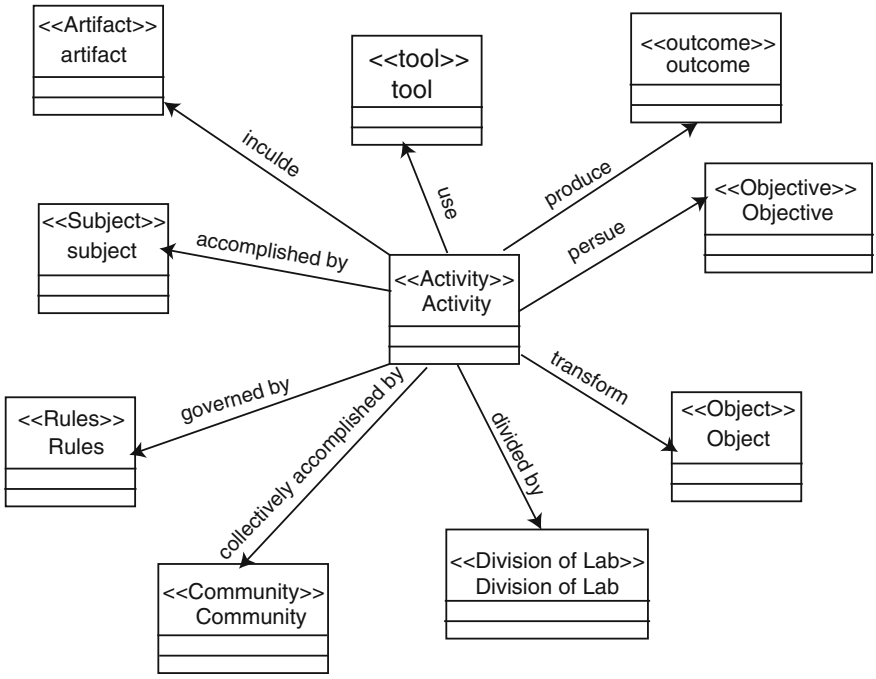


Fig. 2 Object oriented activity theory model using UML framework

4 Application of Web Semantic for Adaptive Business Scenario Using Activity Theory

In order to develop Semantic interoperability with business terminology we use a common platform which can support a relationship between Activity Theory notation, Business Taxonomy and Resource Description Framework [3, 12, 13]. The equivalence relationship can be written as:

M-Sugar manufactured sugar using production planning and control module. During the course of manufacturing it has to time up with various vendors to maintain smooth flow of production. In course of action if any of the drops its business or changes the specification of the component then M-Sugar must be intelligently switch to another vendor (Table 1) in real time mode. This is achieved by the help of Semantic Web Ontology link. Due to this it has to maintain its price accordingly. Sugar being produced by many producers suffers from heavy competition. Initially the firms profitability was given by Fig. 3. This figure shows that previously the Marginal Revenue (MR) curve was cutting the Marginal Cost (MC) curve at point A thereby maintaining a fixed profitability P shown by the shaded area.

Table 1 Mapping of AT, DBE and semantic Web

AT notation	Semantic web	Adaptive business situations
Activity	From a pairinf of entity	Adjust in Taxonomy
Object	Drop in schema tree	Product/services stopped
Subject	Information	Domain
Outcome	Semantic upgradation	Dropping of product/services
Objective	Real time binding	Update information
Tool	IDT like Prometheus	Process modelling tool
Community	Website	Stakeholders

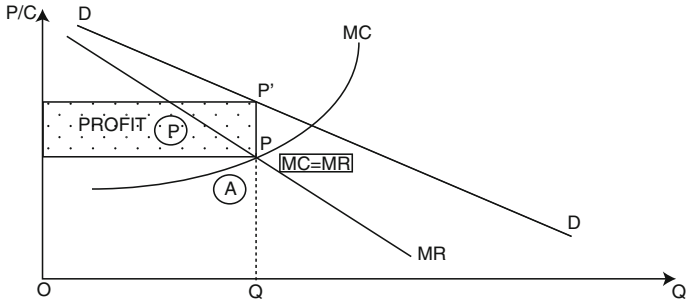


Fig. 3 Impact of demand *curve* shift and its relation with *LMC* and *LAC*

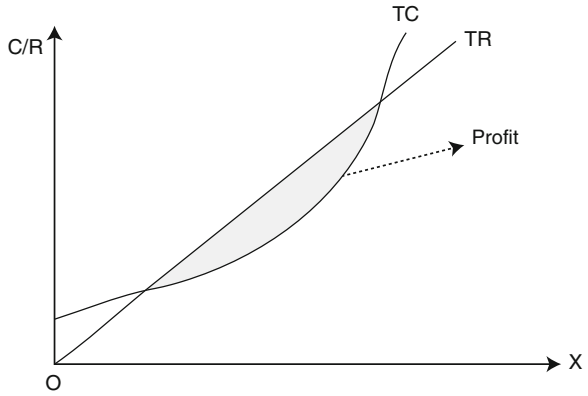


Fig. 4 An increase in variable cost due to entry of new entrants

Now in the above case there were no new vendors being selected, the company has to select manually the alternative vendor even the supply chain logistics was not able to trade. the … Now in the above case because of dynamic linking of the vendors at run time environment. M-Sugar is able to maintain a smooth flow logistic of component thereby change in Marginal Revenue of the curve takes place (Fig. 4). Intelligent collaboration of vendors lead to decrease the overall cost of production sustaintially. It is because of the change management being taken in

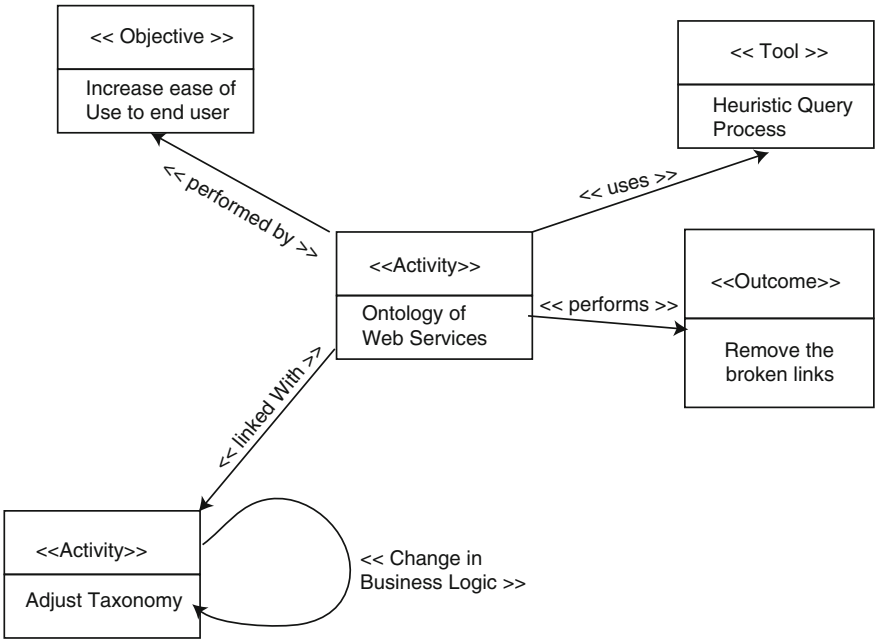


Fig. 5 AT—business—semantic Web framework

proactive manner. The changes are reflected in RDF formation accordingly. Figure 5 shows a common framework cohere the adaptive nature of RDF [1, 2] framework is identified. It provides unique Lin-chain rule where three basic steps are performed capturing change, notifying change and handling change is observed.

(a) **Capturing Change:** This step captures events and store the exception in the work-flow.
(b) **Notifying Change:** It records the number of exceptions occurred from the users perspective like broken link of Vendor etc.
(c) **Handling Exception:** It defines the set of rules that are managed by adaptive ontology in order to provide smooth selection to the User as stated by Lin, Bonguetta and Salmon [16, 17, 20].

5 Petrinet Calculus

We can represent the entire RDF graph (Fig. 7) using Petrinets [19, 6] as it will verify the basic workflow model (Fig. 6).

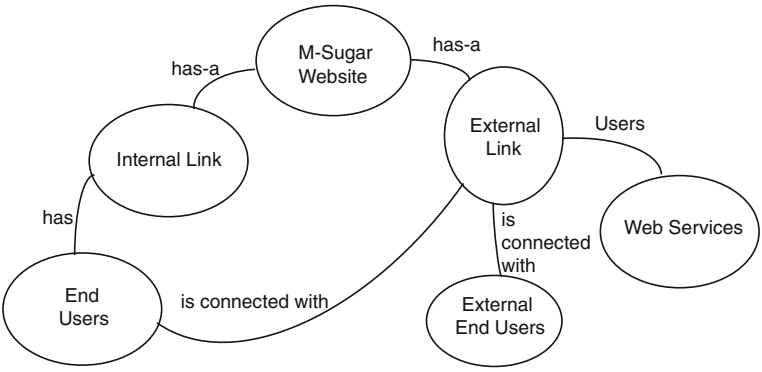


Fig. 6 An ontology tree

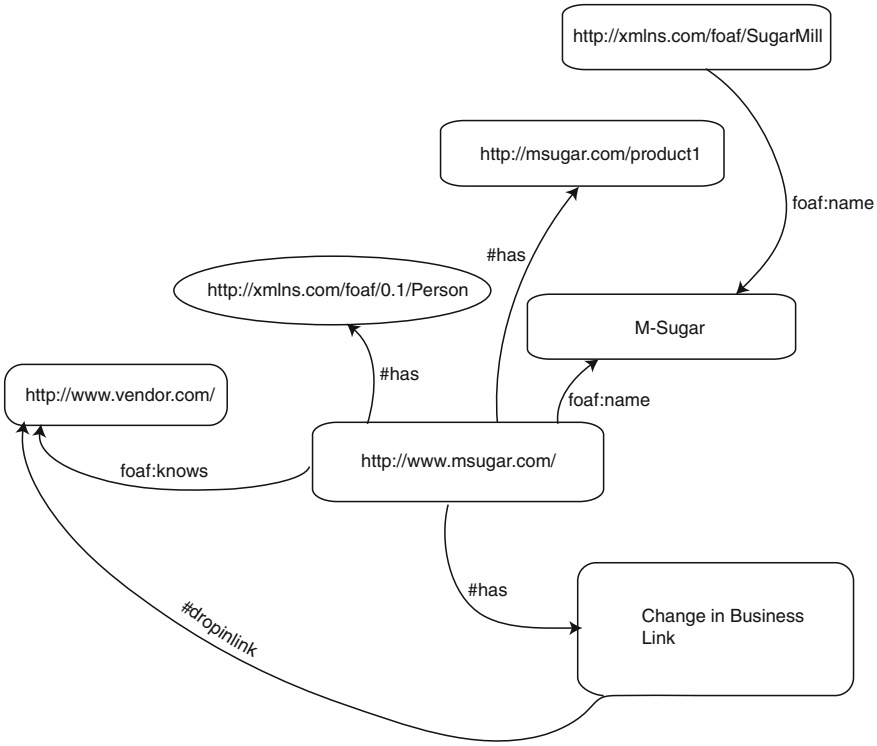


Fig. 7 Basic RDF graph of "Morolo"

It was invented by Carl Adam Petri [20] in sixties. Petrinets provides a strong foundation to formalisation, its pictographical nature (Fig. 8), expressiveness. Petrinet is a directed bipartite graph with two nodes types called places and

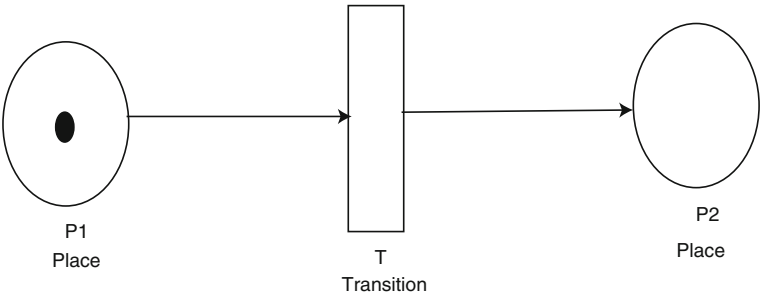


Fig. 8 A simple Petrinet

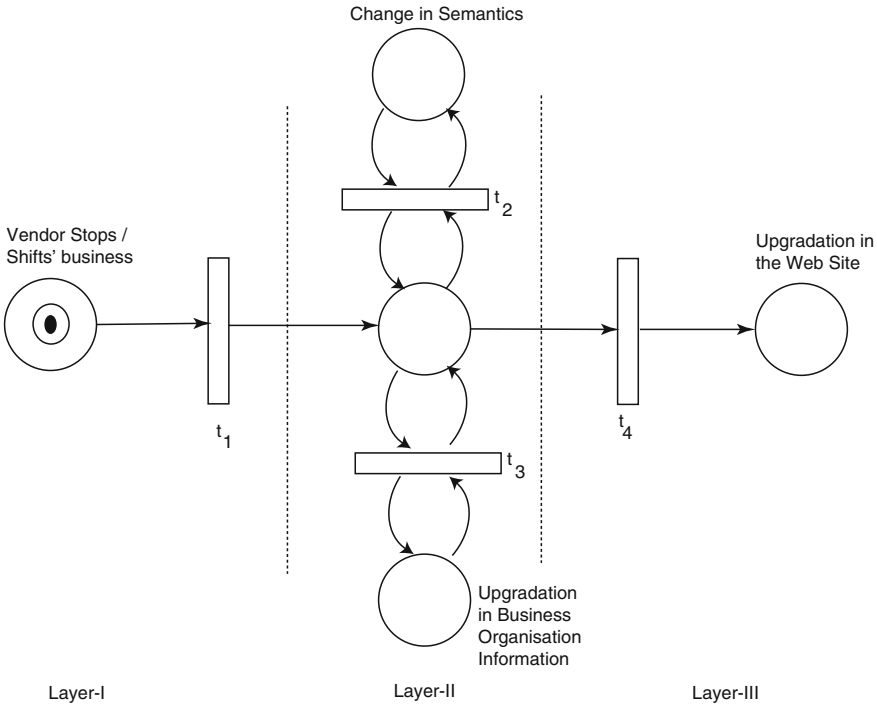


Fig. 9 Layered structure of process flow

transitions. The nodes connected via directed arcs. The simple definition of Petrinet is a triple (P, T, F). P is finite set of places. T is finite of Transitions:

$(P \cap T = \phi)F \subseteq (PXT) \cap (TXP)$ is a set of Arc. $(Flowrelations)$.

A place p is called an input place of transition t iff there exists a directed arc from place p to t. A transition t is said to be enabled iff each input place p of t continues to have at least one token. If transition t fires, then t consume one token

from each input place p of t and produces one token in each output place p of t. The initial marking is given as $\{1, 0, 0, 0\}$ corresponding to $\{t_1, t_2, t_3, t_4\}$. The above model can be analysed by using:

- Reach ability graph
- Place and transitions graph
- Simulation

We can represent the Process flow as in Fig. 9. Layer-I shows shops/shift its business, which results in change/up-gradation of business organisation information, resulting in change in semantic (Layer-II). Finally the outcome is represented in Layer-III eliminating the link dynamically from the website.

The following algebraic form represents the movement of token in Petrinets. The main purpose of representing it into algebraic form is its easy convertibility into PNML Notations, which can be converted into XML file. It is embedded with Eclipse to support the linking with HPSIM2.0. XML file can be directly converted into RDF notation. It provides concise and readable textual notations of the graphical model. The format represents the basic steps:

Step 1:

$$(ads : has\_product\_id, has\_Exception, has\_sub\_process, super\_concept\_of)$$

Step 2:

$$(ads, psad : workflow\_pattern\_id, has\_inActivity, has\_outActivity, dropping\_in\_service, stakeholder)$$

Step 3:

$$(psad : schema\_tree, real\_time\_binding)$$

where

$$ads = adaptive\ business\ situation$$

$$psad = process\ semantic\ dynamics.$$

These steps can be developed in Petrinet Calculus where the two basic entities adaptive business situation(ads) and “process semantic dynamics” can be implemented. Any drop in the business by the vendor demands a real time dropping of the link which is named as Vendor’s Link. There is an entity known as Link_Connector which maintains the counter of the links and logic-connector values (Fig. 9).

The above equations can also represented by Petrinet language notation by using PNML. Entire Petrinet equations can be converted to XML file which can directly represent RDF.

$$D[adbs\ x:\ int\ action\ n, psd] := consume[psd\ from\ x\ with\ x > 0,\ n\ from\ x\ with\ x < 0]$$
$$in\ (trans[psd]\ merged\ to\ trans[n]\ merged\ to\ trans[n]\ merged\ to\ place[x : int\ init\ empty])$$

The semantics of this script is as follows:

$$D = (x[(((psd, x), (psd, x), (ads, x)|x > 0)]).$$
$$x[((n, x); (n, x)|x < 0)].N$$

where x and n are defined as

x number of places required to represent business

n number of times action is taken.

6 Conclusion

The paper end-up by drawing a conclusion that embedding petri-net in semantic web provides a formal method of developing a business process in systemic method. The use of Activity Theory provided easy conversion of Activity Theory framework of business into Activity Oriented Model which is extremely useful in building Ontology relationships. We will extend the above model by applying Visual Cognitive Language (VCL) linking it with Web-Eco-AT framework for changing Business Environment.

References

1. Benatallah, B., Medjahed, B., Bouguettaya, A., Elmagarmid, A.K., Beard, J.: Composing and Maintaining Web-based Virtual Enterprises. TES (2000)
2. Berners-Lee, T., Fischetti, M., Foreword By-Dertouzos, M.L.: Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor. HarperInformation (2000)
3. Brambilla, M., Ceri, S., Comai, S., Tziviskou, C.: Exception handling in workflow-driven Web applications. Proceedings of the 14th International Conference on World Wide Web. ACM (2005)
4. Darbari, M., Asthana, R., Ahmed, H., Ahuja N.J.: Enhancing the capability of N-dimension self-organizing petrinet using neuro-genetic approach. Int. J. Comput. Sci. Issues (IJCSI) **8**(3), 569–571 (2011)
5. Darbari, M., Dhanda, N: Applying constraints in model driven knowledge representation framework. Int. J. Hybrid Inf. Technol. **3**(3), 16–21 (2010)

6. Darbari, M., Singh, V.K., Asthana, R., Prakash. S.: N-dimensional self organizing petrinets for urban traffic modeling. Int. J. Comput. Sci. Issues (IJCSI) **7**(4), 37–40 (2010)
7. Deshpande, A., Ives, Z., Raman, V.: Adaptive query processing. Found Trends Databases **1**(1), 1–140 (2007)
8. Du, W., Ensan, F.: Technologies and Applications. Canadian Semantic Web. Springer, Berlin (2010)
9. Ellis, C., Keddara, K.: ML-DEWS: Modeling language to support dynamic evolution within workflow systems. Comput. Support. Coop. Work (CSCW) **9**(3–4), 293–333 (2000)
10. Engestrom, Y.: 23 Innovative Learning in Work Teams: Analyzing Cycles of Knowledge Creation in Practice. Perspectives on Activity Theory, p. 377 (1999)
11. Hollingsworth, D., Hampshire, U.K.: Workflow management coalition the workflow reference model. Workflow Manage. Coalition 68, 2–55 (1993)
12. Howse, J., Stapleton, G., Taylor, K., Chapman, P.: SAWSDL-iMatcher: A customizable and effective Semantic Web Service matchmaker. Web Semant. Sci. Serv. Agents World Wide Web **9**(4), 402–417 (2011)
13. Howse, J., Stapleton, G., Taylor, K., Chapman.: Visualizing Ontologies: A Case Study. The Semantic Web-ISWC 2011, pp 257–272. Springer, Berlin (2011)
14. Leontjev, A.N.: Problems of the Development of the Mind. Progress, Moscow (1981)
15. Liu, X., Akram, S., Bouguettaya, A., Papazoglou, M.: Change Management for Semantic Web Services. Springer, Berlin (2011)
16. Roman, D., Lausen, H., Keller, U., de Bruijn, J., Bussler, C., Domingue, J., Stollberg, M.: D2v1. 2. web service modeling ontology (WSMO). WSMO Final Draft April 13 (2005)
17. Sheth, A., Miller, J.A.: Web services: technical evolution yet practical revolution. IEEE Intell. Syst. **18**(1), 78–80 (2003)
18. Siddiqui, I.A., Darbari, M., Bansal, S.: Application of activity theory and particle swarm optimization technique in cooperative software development. Int. Rev. Comput. Soft. **7**(5), 2126–2130 (2012)
19. Srivastava, A.K., Darbari, M., Ahmed, H., Asthana, R.: Capacity requirement planning using petri dynamics. Int. Rev. Comput. Soft. **5**(6), 696–700 (2010)
20. van der Aalst, W.M.P.: The application of Petri nets to workflow management. J. Circuits Syst. Comput. **8**(01), 21–66 (1998)
21. Vygotsky, L.S.: Mind and Society: The Development of Higher Mental Processes. Harvard University Press, Cambridge (1978)
22. Yagyasen, D., Darbari, M., Shukla, P.K., Singh, V.K.: Diversity and convergence issues in evolutionary multiobjective optimization: application to agriculture science. IERI Procedia **5**, 81–86 (2013)
23. Yagyasen, D., Darbari, M., Ahmed, H.: Transforming non-living to living: a case on changing business environment. IERI Procedia **5**, 87–94 (2013)

Method-Level Code Clone Modification Environment Using CloneManager

E. Kodhai and S. Kanmani

Abstract The primary objective of code clone research is to provide techniques and tools through which the topics such as clone detection and clone management. A number of techniques have been proposed for clone detections and sure to have even more detectors in future. Some limited methods have been proposed for clone modifications. A technique that helps for clone modification is refactoring. But this is not possible for all the clones, as there are clones which cannot be modified. Moreover, some of the clones have to exist to maintain the consistency of the problem. Most of the programmers modify the clone and need to make the modification throughout all the identical clones. We propose a method, which provide a modification environment of the clones for the programmer. We use the clone detection tool CloneManager. We embedded this feature as an enhancement of the clone detection tool, CloneManager.

Keywords Software clones · Refactoring · Software metrics · Clone detection

1 Introduction

Clones are often the result of copy-paste activities. Such activities are very easy and can significantly reduce programming effort and time as they reuse an existing fragment of code rather rewriting similar code from scratch. Various kinds of code

E. Kodhai (✉)
Department of Computer Science Engineering, Pondicherry Engineering College, Puducherry, India
e-mail: kodhaiej@yahoo.co.in

S. Kanmani
Department of Information Technology, Pondicherry Engineering College, Puducherry, India
e-mail: kanmani@pec.edu

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_45,

clone detection methods have been devised, and a lot of practical code clone detection tools have been developed and used [1]. Research shows that a significant amount of code (7–23 %) of a software system is cloned code [2].

There are four clone types in total, in which the first three are textual and the last one is functional [1]. Roy and Cordy [1] have performed one of the most comprehensive studies in comparing and evaluating clone detection tools and techniques. They provide a qualitative comparison and evaluation of clone detection techniques and tools and organized these large set of information into a framework with coherence. They classified, compared and evaluated the tools and their approaches in two different directions.

Several studies show that lightweight text-based techniques can find clones with high accuracy and confidence, but detected clones often do not correspond to appropriate syntactic units [3, 4]. Moreover, by refactoring the clones detected, one can potentially improve understandability, maintainability and extensibility, and reduce the complexity of the system [5].

All code clones detected by a code clone detection tool are not appropriate for refactoring. For example, language-dependent code clones [6] are clearly inappropriate for refactoring. It means code clones that indispensably exist in a source code due to the specifications of used program language. Although, the number of approaches and tools has been proposed for clone detection [7, 8], only some knows about which detected code clones are possible for refactoring and how to extract them.

One of the major difficulties with the replicated fragments is that if an error is identified in those codes, all the code fragments which are similar to it should be analyzed to identify the same error [9]. Moreover, when enhancing or adapting a piece of code, duplicated fragments can multiply the work to be done [10].

This paper presents the proposal for the creation of an environment for simultaneous clone modification. Modification activities are very easy and can significantly reduce programming effort and time as they reuse an existing fragment of code rather rewriting similar code from the scratch. It uses CloneManager tool [11] to detect the type1 clones. On the retrieved clones, modification is performed which gets automatically updated in all the other similar code clones and helps in reducing the code complexity. Thus it provides an environment which helps the programmers to perform simultaneous modification in clones. This environment is appended as an additional feature to the existing tool CloneManager tool.

This paper is organized as follows; Sect. 2 discusses the related work of the paper. Section 3 describes the CloneManager tool as a background of the proposal. The Sect. 4 presents the proposal of the paper. Section 5 discusses the experiment and results and finally Sect. 6 concludes the paper.

2 Related Work

There has been some research on tools for clone maintenance. Higo et al. [10] have proposed a simultaneous modification support method and developed a tool named Libra. First a maintainer identifies the code fragment that must be modified and then code clones between the code fragments and the original source files of the system are detected. He first used CCFinder [7] to detect the clones. CCFinder can detect code clones with free code fragments, which further need to analyze the clones for modification.

Similar defects is detected in the large set of source code was proposed by Yoshida et al. [12]. This system gets input as a query with a code fragment which is containing a defect, and gives back the code fragments which are containing the same or synonymous identifiers. This system finds similar bugs in the large set of source codes, thus helps to debug the defects.

We propose a method for the creation of an environment for simultaneous clone modification. Our proposal handles clone clusters for clone modification which is lagging in Ekoko's work. We also use the existing tool CloneManager [11] for clone detection phase. This tool detects clones at method-level rather than free code segments as CCFinder as used by Yosshiki Higo. Since method-level clone detection are more appropriate for further clone management. On the retrieved clones, modification is performed which gets automatically updated in all the other similar code clones in the clone cluster.

3 CloneManager Tool

The existing code clone detection tool called CloneManager [11] is used for the detection of clones. Since it is a metric-based clone detection approach, it is more suitable as it can easily target the refactoring operations. The CloneManager tool is developed for code clone detection effectively and accurately. It detects all four types of code clones with the granularity of method level. This tool is implemented using Java to detect clones in C or Java source codes through metrics and textual analysis.

All four types of clones are detected by the tool and then classified and clustered as clone clusters as results. The granularity level for this code clone detection process is methods. The tool gets the project as input containing the files, which is further analyzed for clone detection. The user can choose the type of clones to be detected.

The output of the clone detection tool is specified as clone pairs. Clone Pair (CP) is the pair of code fragments which is same or similar to each other. After, detecting the clone pairs in each type, the result has to be given in an appropriate form for further analysis. The detected clone methods in each clone type are groups into clone clusters (CC). All the members of the clone clusters are associative as each clone pairs are commutative. It means that every member of the

clone cluster becomes a clone pair of every other member of clone pair of the same clone cluster. Finally, the detected clones and clusters as a result are further stored in the text files.

4 Proposed Method

The proposal of the system is to create an environment for the simultaneous clone modification. This approach includes creating an environment to perform modification. The programmer must ensure that when a modification is done in one part of the clone gets automatically updated in all the other parts of the clone. This modification activity not only reduces the programmer's work but also saves time.

We use the existing clone detection tool, CloneManager [11] which detects the clone pairs and clusters and stores the results in a text file. The system uses the clone type options tool to detect the type 1 clones. Two windows are opened, with one holding the type 1 clones and the other with original source code. The type 1 clones which are listed under on cluster are all highlighted together with different colors than the original background. Later, if the developer makes any changes in the source code in clones, then the same will be updated in all the clones.

The Fig. 1 depicts the System Architecture of the clone maintenance. There are four phases in our proposed system that explains how our modification process is carried out. They are as follows:

1. Checking source for clone detection
2. Clone detection using CloneManager
3. Highlighting the clones
4. Clone environment.

4.1 Checking Clone Detection

The first step is to check if clones for source code in the input file are already detected. The main purpose of this step is to reduce the time used for clone detection as files can be varying size and detecting clones for the same may take some time. Thus input is checked with database to find if clones are already detected in input. The database contains the hash value of the files used. If clones are detected in the input then the next step namely clone detection can be skipped and detected clones are displayed. Thus if it is available it directs to the database for the search of detected clones results. If, not it directs to the clone detection tool, CloneManager to detect the type 1 clones using the clone type options.

The database contains all the input files for determining the exact code clones of a particular file. The database will contain the filename and also the last modification date done to the respective input file. It also helps us to retrieve the exact

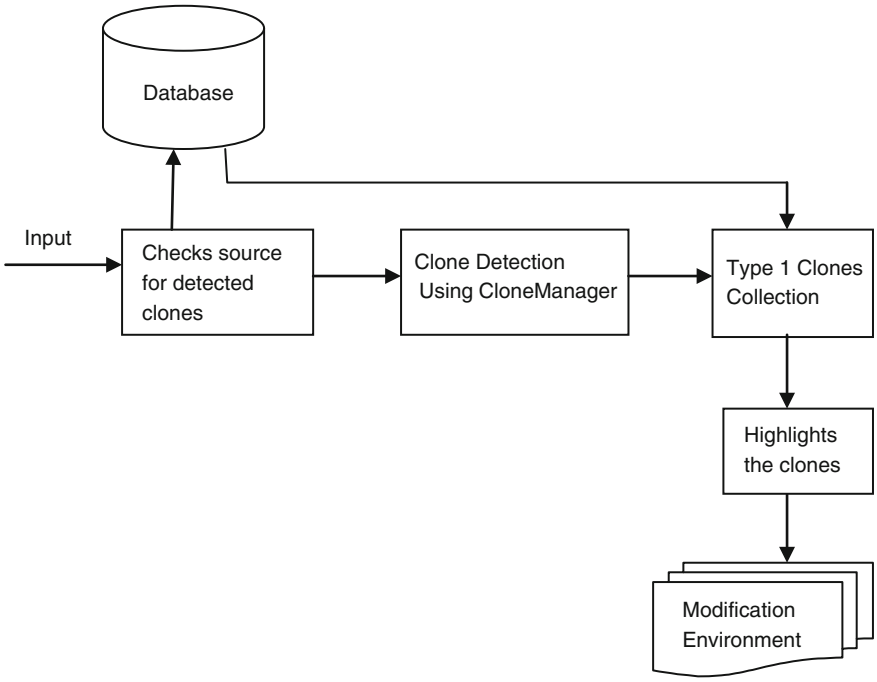


Fig. 1 System architecture for clone maintenance

clones if it is already detected in the program by indicating the start position lines of the clones that appear in the program. This helps in notifying the exact clones at once when respective input file is viewed. If an input file does not have any clones in the program, clone manager takes the responsibility to retrieve the exact clone segments and its varying position of occurrence in the program.

4.2 Clone Detection

This phase uses the clone detection tool CloneManager for clone detection process. It detects the type1 clones alone and stores it in the database. This is because the detection process will be completed in time as we do not want other clone detection types 2, 3 and 4. Moreover these other clone types cannot be modified simultaneously. Thus detection type 1 clone alone will be detected in few seconds.

The required clones can be selected and edited. When a single clone is edited the changes are reflected in all the clones of the same type. It is easy to modify all clones since changes to any one clone are automatically and simultaneously made to all similar clones and it is easy to modify a single clone instead of without changing editing modes. This process reduces the user's processing time and hence can improve productivity.

4.3 Highlighting the Clones

The collected clone helps us to identify the code clones in the original source code. The exact location of the clones in the source code can be determined in clone collector. Using the string matcher technique the collected clones finds the similarity between the original source code and the cloned codes. Using the highlighter method the occurrences of the code clones in the source code are highlighted. The clones are integrated together using the integrator method. The clones are collected together using the file integrator.

The location of the code clones are found out using the string matcher. They are highlighted using a different background color such as yellow color to display the accurate location and the presence of the clones in the original source code. The remaining codes as usual are left with the white background color as usual.

4.4 Clone Environment

The final stage of our work is carrying out modification to the clones that are highlighted. The modification done to retrieved clones will lead to automatic updation on the rest of the clone segments. This CloneManager helps in automatic modification and reduce programmer's effort by modifying each and every line of a program. It is possible to have an overall idea on the other files containing other similar copies of that fragment and modify in a better effective manner using CloneManager. Figures 2 and 3 shows the example of how modification is carried out in the source code.

5 Experimental Analysis

5.1 Experimental Setup

The proposed method is implemented and experimented with C and Java Projects. We have chosen the dataset which have been already evaluated in the literature, so that it will be helpful for us to do comparison. We compared our results with the existing tool.

To measure the accuracy of our proposed work, we use precision, recall, and F-measure, the three standard metrics.

- *Precision* is the amount of clones found by the detection tool that are correct, over the total number of clones found by the detection tool
- *Recall* is the amount of clones found by the detection tool that are correct, over the total number of clones in the application analyzed

Fig. 2 Original source code as input

```
1  static void setstringOption(String  value, String option, List dest)
2  {
3  if (value != null && !value.trim().equals(“”)
4  { // NO118N
5  List subList = new ArrayList();
6  subList.add(option);
7  subList.add(value);
8  }
9  System.out.println(option + “ “ + value);
10 }
```

Fig. 3 Modification done to the source code

```
1  static void setstringOption(String  value, String option, List dest)
2  {
3  if (value != null && !value.trim().equals(“”)
4  { // NO118N
5  List subList = new ArrayList();
6  subList.add(option);
7  dest.Addall(subList); // modification to the clone
8  subList.add(value);
9  }
10 System.out.println(option + “ “ + value);
11 }
```

- *F-measure* is the harmonic mean precision and recall, the F-measure or F-score:

$$F = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

Ideally, precision and recall should be 100 %. Finally, we also cross checked the results by manual inspection of the open source projects.

5.2 Datasets

We have analyzed with a medium sized program called JHotDraw which is for structured drawing editors of approximately 70,000 lines and to the large size program called Apache-httpd with 275,000 lines. Table 1 lists the features of open source projects which are taken for the performance analysis of our CloneManager tool.

In Table 1, the second column lists the open source program names of the input project. The third column is the number of files. The fourth column is the no. of lines in the source code in thousands. The last column is the program language of

Table 1 Projects chosen as dataset for clonemanager

S.no	Project name	# files	LOC in K	Language
1	Canna 3.6	96	100	C
2	Canna 3.6p1	96	100	
3	Apache-httpd-2.2.8	496	275	
4	Apache-httpd-2.2.9	498	275	
5	JHotDraw 5.4b1	466	70	Java
6	JHotDraw 5.4b2	484	72	
7	Ant 1.6.0	627	181	
8	Ant 1.6.1	631	160	

each project. Table 2 shows the detected clones by CloneManager and results for all chosen dataset.

5.3 Results

In Table 2, the second column lists the open source program names of the input project. The third column is the number of clones. The fourth column is the no. of type 1 clones. The fifth column is the no. of type 1 clone clusters. The last column is the number of clusters where modification is carried out.

The first project is taken for analysis is canna open source code. When compared with the canna 3.6 and canna 3.6p1 version the number of clusters modified done was 12 clusters. In the same way all the values are calculated for the remaining projects and they are displayed is the last column of the Table 2. From the results, one can observe that the no. of modification carried out in the version was not directly proportional to number of clusters in the project.

5.4 Evaluation of the Tool

The Table 3 shows the evaluation of CloneManager produced for all the datasets. The column 3 holds the recall values produced in percentage. The Column 4 holds the precision values in percentage. The column 5 holds the F-measure in percentage. The last column is the run-time of each project in seconds, the next evaluation parameter of the tool.

From the results produced, the precision and recall parameters are calculated for each refactoring patterns for all the chosen datasets. We observed that the precision and recall percentage is above 88 for all the datasets. Even though it is not feasible to get 100 % for all methods, we had few 100 % results. Thus our tool proves to provide high precision and recall, which are the best parameters for the

Table 2 Detected clones by clonemanager and results

S.no	Project name	# clones	# type 1 clones	# type 1 clusters	# clusters where modification is carried out
1	Canna 3.6	5,467	1,028	259	12
2	Canna 3.6p1	5,472	921	381	
3	Apache-httpd-2.2.8	1,146	183	107	23
4	Apache-httpd-2.2.9	1,086	152	48	
5	JHotDraw 5.4b1	1,198	291	137	10
6	JHotDraw 5.4b2	1,206	272	112	
7	Ant 1.6.0	16,572	1,562	375	17
8	Ant 1.6.1	16,038	1,564	383	

Table 3 Evaluation of clonemanager

S.no	Project name	Recall %	Precision %	F-measure	Run-time in s
1	Canna 3.6	96	100	97.99	6.5
2	Canna 3.6p1	94	95	94.50	6.8
3	Apache-httpd-2.2.8	88	96	91.83	3.4
4	Apache-httpd-2.2.9	90	100	94.74	3.6
5	JHotDraw 5.4b1	100	95	97.44	4.2
6	JHotDraw 5.4b2	96	88	91.83	4.3
7	Ant 1.6.0	100	90	94.74	9.2
8	Ant 1.6.1	98	96	96.99	9.1

evaluation of code clones tools. The next parameter F-measure is the harmonic mean of precision and recall. The F-measure also shows higher values. Moreover, the run-time of the tool shows that it is capable of executing in less time.

5.5 Comparison of the Tool

Having computed the results we compared our evaluation results with that of the existing tool. Table 4 shows the comparison of our CloneManager with Libra. The tool considered for analysis is Libra [10]. Libra developed by Yoshiki et al. [10] is a simultaneous modification support tool. Yoshiki et al. has tested the tool with two open source program canna 3.6 and Ant 1.6.0.

From the Table 4, we compared our tool results; which shows high values in recall, precision and F-measures. Moreover, the precision and recall for their tool is only above 81 %, where as our tool is above 90 %.This reveals that our tool is able to find and does modification process better than Libra. The time taken is also lesser as shown in the table.

Table 4 Comparison of clonemanager with libra

Parameter	CloneManager		Libra	
	Canna	Ant	Canna	Ant
Recall %	96	100	81	100
Precision %	100	90	100	83
F-measure	97.99	94.74	89.50	90.71
Run-time in s	6.5	9.2	8	90.71

6 Conclusions

Thus we have implemented our proposed method for clone maintenance by retrieving the similar clones using CloneManager that does two functions. After retrieving the exact clones, in the highlighting phase, they are highlighted in the source code by which it separates the type 1 clones from the other code. In modification phase, a change in the code fragment will lead to automatic updation in all the highlighted area. Therefore automatic modification to the exact clones reduces the programmer's effort and complexity.

References

1. Roy, C.K., Cordy, J.R., Koschke, R.: Comparison and evaluation of clone detection techniques and tools: a qualitative approach. Sci. Comput. Program. **74**, 470–495 (2009)
2. Kapser, C., Godfrey, M.: Supporting the analysis of clones in software systems: a case study. J. Softw. Maintenance Evol.: Res. Pract. **18**, 61–82 (2006)
3. Bellon, S., Koschke, R., Antoniol, G., Krinke, J., Merlo, E.: Comparison and evaluation of clone detection tools. IEEE Trans. Softw. Eng. **33**, 577–591 (2007)
4. Rysselberghe, F.V., Demeyer, S.: Evaluating clone detection techniques. In: Mens T., Ramil, J.F., Godfrey, M.W., Down B (eds.) International Workshop on Evolution of Large-scale Industrial Software Applications, Vrije Universiteit Brussel, Brussel, 25–36 (2003)
5. Fowler, M.: Refactoring: Improving the Design of Existing Code. Addison-Wesley, Boston (2000)
6. Higo, Y., Kamiya, T., Kusumoto, S., Inoue, K.: Method and implementation for investigating code clones in a software system. Inf. Softw. Technol. **49**, 985–998 (2007)
7. Kamiya, T., Kusumoto, S., Inoue, K.: Ccfinder: a multilinguistic token-based code clone detection system for large scale source code. IEEE Trans. Softw. Eng. **28**, 654–670 (2002)
8. Jiang, L., Misherghi, G., Su, Z., Glondu, S.: Deckard: scalable and accurate tree-based detection of code clones. In: International Conference on Software Engineering '07. (2007)
9. Li, Z., Lu, S., Myagmar, S., Zhou, Y.: CP-Miner: finding copy- paste and related bugs in large-scale software code. IEEE Trans. Softw. Eng. **32**, 176–192 (2006)
10. Higo, Y., Ueda, Y., Kusumoto, S., Inoue, K.: Simultaneous modification support based on code clone analysis. In: 14th Asia-Pacific Software Engineering Conference, pp. 262–269

11. Kanmani, S., Kamatchi, A., Radhika, R., Saranya, B.V.: CloneManager: a tool for detection of type1 and type2 code clones. In: International Conference on Recent Trends in Business Administration and Information Processing, Springer digital library, Trivandrum, Kerala, India, March 26 & 27 (2010)
12. Yoshida, N., Hattori, T., Inoue, K.: Finding similar defects using synonymous identifier retrieval. In: International Workshop on Software Clones '10, Cape Town, South Africa (2010)

An Educational HTTP Proxy Server

Martin Sysel and Ondřej Doležal

Abstract The efficiency and safety of Web access can be enhanced by the deployment of an http proxy server in many cases. The first part of this paper provides an introduction to the issue of an HTTP proxy server. The second part of the paper describes used technologies and an implementation of a multithreaded HTTP proxy server with an embedded WWW server used for the graphics user interface. In its current state, the developed proxy server can be used to monitor the WWW traffic of a local area network and, with further development of its functionalities, can include such areas as content filtering or access control.

Keywords Proxy server · HTTP · Socket · Thread

1 Introduction

The global computer network Internet, from its beginnings, expanding rapidly in the 70th and 80th of the 20th century. There are connected to a network more than 1.67 billion users now. The most common use of end users access to the Internet is WWW—World Wide Web. This is so dominant that even in normal communication can be traced merging concepts of Internet and WWW. The reasons for such popularity are more—architecture enabling seamless collaboration between completely heterogeneous systems, intuitive user interface and also the ability to create complex applications.

M. Sysel (✉) · O. Doležal
Faculty of Applied Informatics, Department of Computer and Communication Systems, Tomas Bata University in Zlín, Zlin, Czech Republic
e-mail: Sysel@fai.utb.cz

O. Doležal
e-mail: Dolezal_o@fai.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_46,

The actual World Wide Web can be characterized as a distributed client-server information system with thin client, transmitting information in the form of HTML pages and other objects using HTTP application protocol [1, 2]. It is a typical protocol of request-response, which controls the data transfer between server and client (such as a web browser). HTTP traffic that is point-to-point communication but there are a lot of use cases where this communication can benefit from inclusion of additional active element—proxy server [3].

Proxy server is the middle element in communication between the server and the client. This element may only redirects communication or checks the application protocol. HTTP proxy server may accelerate access to resources and perform inspection or monitoring operations. Protect the privacy of users can be provided too [4].

The aim is to analyze the requirements for Internet HTTP proxy server, specify the ways of their solution and create application program design that will implement educational HTTP proxy in GNU/Linux including necessary documentation. The program is created using free software and developed product itself is published under an open source license GNU/GPL.

2 Proxy Server

A proxy server is usually a computer system—a combination of hardware platforms and software applications—which serves as an intermediary in the network communication between the parties. The client-server systems provide an intermediary in the communication between the client (usually sending requests) and the server (sending the response), see Fig. 1. Generally, to the proxy is not limited for client-server systems, suitable proxy can provide the exchange of messages in networks such as peer-to-peer.

The basic principle of operation of the proxy server is to receive client requests (against which looks like a server), these requirements are analyzed and then send the target servers (to whom they are acting as a client), and then the answers to pass the original client—in original or modified form. The proxy server operates on the 7th layer ISO/OSI model (application) to analyze incoming requests [5, 6]. Therefore, it's also called this proxy as an application proxy. Proxy server works with the same application protocol such as serviced clients. Operation with various protocols can be achieved by different proxy servers, or multi-protocol servers.

In addition to this application proxy, there are also application-independent ones, providing only transport packets without the knowledge of application layer protocols. Their using however requires the use of specific communication protocol to communicate with the proxy server. A typical representative of the universal proxy protocol is SOCKS protocol working on the 5th layer of ISO/OSI model of the session. SOCKS routes network packets between a client and server through a proxy server. SOCKS5 additionally provides authentication so only authorized users may access a server [5]. Deployment SOCKS proxy client

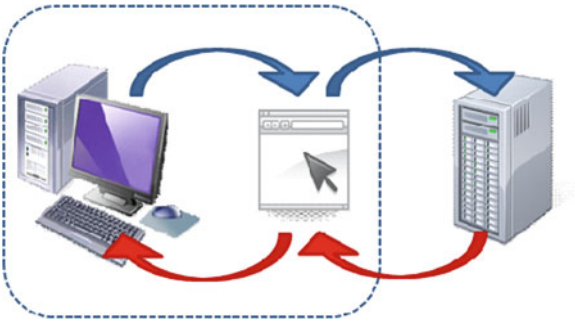


Fig. 1 Proxy server communication

application requires adaptation—modification of the network code. But there are client implementations that after running redirect network traffic to the client SOCKS proxy in the protocol, eliminating the need to modify the client code.

2.1 Reasons for Using the Proxy Server

The primary reason for deploying of the first proxy server: allow access to external sources of computer facilities inside the firewall-protected network or otherwise directly inaccessible. Proxy server in this case is installed on the computer with a firewall, and serves as a gateway for intermediating network traffic of the application protocol. A similar effect—serving resources—can be achieved in the presence of a suitable firewall with a rule opened for outbound traffic; then communication between the client and the server is routed normally [3, 7].

An application proxy offers next functionality [4, 8]:

- In most cases, more clients can access to proxy server. All responses to requests pass through proxy server, and if the proxy server stores the contents of the answers can improve response times and reduce bandwidth to repeat. Requests use the stored response from the previous identical requests. Such a proxy server is called a caching proxy. Most of the HTTP proxy servers implement this functionality. The validity of stored responses is very important; this issue is described by HTTP protocol specification [1, 2].
- Proxy server allows processing of the finer requirements. Generally, the organization can restrict access to individual client computers by destination address, protocol or type of resource. Specialized HTTP proxy servers also support a time limit within a day or rate control, which can reduce inefficient using of bandwidth during working hours or using for improper purposes.
- Filtering proxy can be used to detect and block malicious content. Again thanks processing at the application level proxy server can perform scanning incoming content and block access to the infected sources. Similarly, the inspection of outgoing data for viruses and generally known malware can be done.

- Modification of the content of other platforms, such as access to WWW resources from mobile device. Application proxy can perform dynamic re-compression to reduce data flow transcription content to skip unsupported components, etc. It is possible to combine functionality with caching and save the results to avoid their recurrence.
- Increased security can be achieved by using an application proxy server for logging and audit client requests, as opposed to logging on lower layers enables logging at the application layer easy access to all of the client/server transaction property.
- Application proxy server may in appropriate cases, make transfers between different protocols on the client side and the server side. In practice, it works as a translator at the level of application protocols. Clients can access the resources or client software programmer saves many lines of code.
- Last but not least, the proxy server can be used for anonymous access to the target server. This effect can be used to bypass website restrictions applicable to the relevant source address of the client, the appropriately configured proxy also mask the client system attributes such as the type and version of software, etc.

3 An Implementation of HTTP Proxy

The practical part of the paper is an implementation of a simple educational HTTP proxy server processing incoming requests in separate threads. The implementation is realized in the programming language C++. The program was created and is working in a Linux environment.

Server implements full support for standard HTTP 1.0, i.e. methods GET, POST, and HEAD, and supports a subset of the HTTP 1.1 standard to allow seamless communication of an existing implementations of client and server (i.e., Web browsers and Web servers) [1, 2].

Implementation of proxy server also contains the HTTP server, which implements the graphics user interface of proxy server. This interface allows the surveillance of communication proxy via WWW browser. It contains overviews of the overall server status, status of individual threads and overview of recent requests to HTTP requests, including the result of their execution. In addition to this overview, proxy server also records complete record of all requests to the log files, separately for proxy subsystem and the HTTP server subsystem.

3.1 System Architecture

The application is programmed as a multi-threaded application. Threads are implemented by producer-consumer model [9]. One thread creates (producer) jobs (incoming requests) and inserts them into the queue, Fig. 1. The new jobs in the

queue are processed by own thread (consumer). This processing is performed in parallel for as many requests as there are currently running consumer threads.

After starting function main() it is opened socket accepting new connections and its launched the loop containing blocked waiting for the new arrival communication. New incoming communication is checked if it is not exceed the maximum limit of connections, and if not, new thread is created with an entry point handleClient() and the new socket is created. Otherwise, the main thread waits to releasing a free thread. It is also updated counter of free threads.

Working thread first receives a client request, it tries to decode using function parseRequest() and, if it is successful (it is correct and supported request), handle response from the remote server by calling the GetResponse() function. This response then returns to the client, sync updated information for using of thread and then thread is finished.

3.2 Memory Requirements

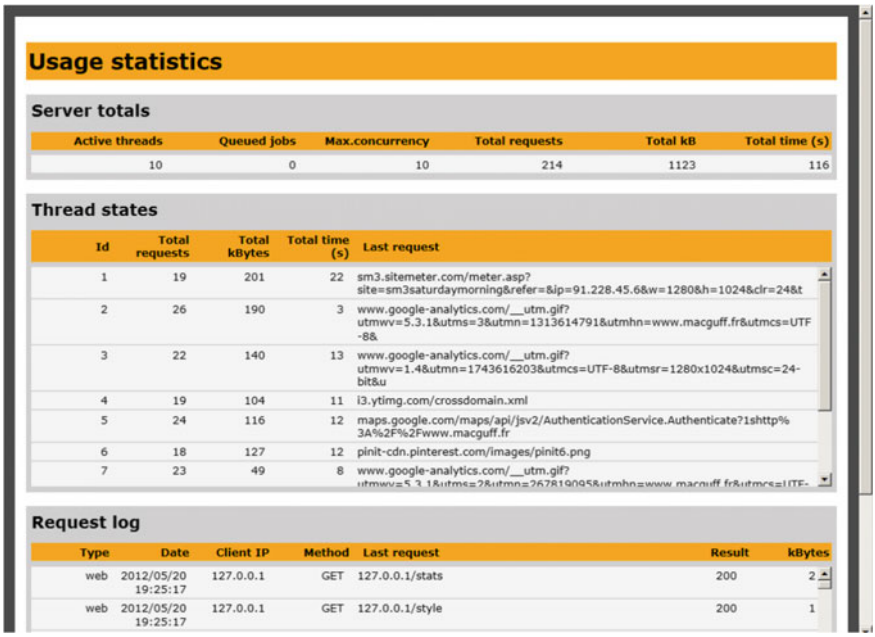
Server uses a minimum of global memory allocated on the heap. Individual threads use a stack (in the current version of glibc fixed-size 2 MB per thread), and a dynamically allocates memory from the heap as needed. To reduce memory consumption, HTTP communications between the client and the target server is solved by interleaving. It does not wait for retrieving of the whole client's request (containing requirements for HTTP entities such as POST) or whole responses of proxy server, but this communication is processed and forwarded immediately after incoming communication. The proxy server avoids the need to allocate memory for a potentially big transmitted communications and makes it with a small buffer (about 2 kilobytes) in real time (depending on the specifics of the used TCP/IP subsystem and network). This mechanism also has a positive effect on the speed of response, because the client receives a reply before the proxy server receives a whole message.

Total memory requirement of an HTTP proxy process is by default about 23 MB (for 10 threads) after starting the application.

3.3 CPU Requirements

Server is effective in terms of processor time consumption. This effectiveness is achieved in particular by eliminating the active waiting. Proxy server realizes all operation of the network input and output as blocking state. Receiving new connections in the main thread in the process, reading requests and outgoing responses of connected sockets in each thread—everything uses blocking state.

As with most network applications, even when proxy server is the main bottleneck of bandwidth network connection—typical current processors are capable



Usage statistics

Server totals

Active threads	Queued jobs	Max.concurrency	Total requests	Total kB	Total time (s)
10	0	10	214	1123	116

Thread states

Id	Total requests	Total kBytes	Total time (s)	Last request
1	19	201	22	sm3.sitemeter.com/meter.asp?site=sm3saturdaymorning&refer=&ip=91.228.45.6&w=1280&h=1024&clr=24&t
2	26	190	3	www.google-analytics.com/__utm.gif?utmwv=5.3.1&utms=3&utmn=1313614791&utmhn=www.macguff.fr&utmcs=UTF-8&
3	22	140	13	www.google-analytics.com/__utm.gif?utmwv=1.4&utmn=1743616203&utmcs=UTF-8&utmsr=1280x1024&utmsc=24-bit&u
4	19	104	11	i3.ytimg.com/crossdomain.xml
5	24	116	12	maps.google.com/maps/api/jsv2/AuthenticationService.Authenticate?1shttp%3A%2F%2Fwww.macguff.fr
6	18	127	12	pinit-cdn.pinterest.com/images/pinit6.png
7	23	49	8	www.google-analytics.com/__utm.gif?utmwv=5.3.1&utms=2&utmn=26781909&utmhn=www.macguff.fr&utmcs=UTF-

Request log

Type	Date	Client IP	Method	Last request	Result	kBytes
web	2012/05/20 19:25:17	127.0.0.1	GET	127.0.0.1/stats	200	2
web	2012/05/20 19:25:17	127.0.0.1	GET	127.0.0.1/style	200	1

Fig. 2 Proxy server gui—Usage statistics [10]

process data much faster than the network can provide data. The application spends waiting for the arrival of additional data or further connections most of the time. When testing an application which consisted of processing 10,000 requests, it was detected by tool gprof total CPU load only 2.5 %.

3.4 Logging and Statistics

Proxy server collects statistical information of operations and stores a record of completed requests in log files and memory buffer that is used to generate web pages with traffic reports, see Fig. 2. All records are written to log files—separately for the Web part, and separately for proxy requests. Statistical information includes the sum of the total number of requests, the sum of transferred data and the lasting time of the operation, and further details of the final processing of this request. The information is recorded for each thread and also for the entire application.

4 Conclusion

This work proved that this developed educational HTTP proxy server is a useful piece of software. It can be used to accelerate access to resources, perform access control or traffic logging. It can be also used to enhance anonymity of its users. Nowadays, one of the most used HTTP proxy servers is the Squid proxy cache. Further discussion was targeted towards technologies necessary for building current HTTP proxies—sockets and threads. A multi-threaded HTTP proxy server in C language was created using discussed technologies which incorporates an embedded HTTP server used to access the runtime information and usage statistics. Target platform was GNU/Linux. Further testing of this server proved that a multi-threaded design is very useful for server applications, and that the created proxy server neither uses significant amounts of system resources nor it degrades the WWW performance in an important way. In its current state, the proxy server can be used to monitor the WWW traffic of a local area network and is suitable for student education. Further work on this project will be given to extending the functionality, as content filtering or access control.

References

1. RFC 1945. Hypertext Transfer Protocol—HTTP/1.0, IETF, http://tools.ietf.org/html/rfc1945
2. RFC 2616. Hypertext Transfer Protocol—HTTP/1.1, http://tools.ietf.org/html/rfc2616
3. RFC 3143, known HTTP Proxy/Caching problems, http://tools.ietf.org/html/rfc3143
4. Books, LLC, General Books LLC. Proxy Servers: Proxy Server, Wingate, Tor, Proxomitron, Proxy Auto-Config, Java Anon Proxy, Sun Java System Web Proxy Server, Web Cache, General Books LLC (2010)
5. RFC 791. Internet Protocol, IETF, http://tools.ietf.org/html/rfc791
6. RFC 793. Transmission Control Protocol, IETF, http://tools.ietf.org/html/rfc793
7. RFC 2617. HTTP authentication: basic and digest access authentication, IETF http://tools.ietf.org/html/rfc2617
8. Dolezal, O.: An HTTP proxy server. UTB in Zlin, Zlin (2012)
9. Gay, W.: Linux socket programming by example. Que, 557 p. (2000)
10. Ubuntu, Squid—Proxy Server—official Ubuntu Documentation, https://help.ubuntu.com/lts/serverguide/squid.html

The Software Analysis Used for Visualization of Technical Functions Control in Smart Home Care

Jan Vanus, Pavel Kucera and Jiri Koziorek

Abstract The article describes the analysis of software environment used for communication between the user and the control center and to processes data during visualization application environment creation to achieve comfortable control of operational and technological functions in intelligent (smart) buildings and finally, the use of the application in smart houses which provide nursing and assistant services for handicapped people and for the elderly.

Keywords Analysis · Visualization · Control · Smart home care · Software

1 Introduction

Persons living in households and requiring daily assistance presents a challenge which includes many everyday functions such as turning the lights ON and OFF when leaving the house, turning OFF the stove or closing windows. Intelligent (smart) electrical installations and systems offer many options how to achieve comfortable control, how to lower consumption of energies, and how to improve in-house safety. Visualization system used to control technical functions in smart houses offer better (higher class) function control. A person may not only control and interfere with the management of the house or monitor the house but also

J. Vanus (✉) · P. Kucera · J. Koziorek
Department of Cybernetics and Biomedical Engineering, VSB TU Ostrava, 17. listopadu 15
708 33 Ostrava, Czech Republic
e-mail: jan.vanus@vsb.cz

P. Kucera
e-mail: kucerapav@seznam.cz

J. Koziorek
e-mail: jiri.koziorek@vsb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_47,

change directly the actual course of actions, monitor and archive important phenomena/situations, create procedures or rules and gradually produce automated processes with the intention to eliminate routines in everyday actions and improve the overall quality of life. Graphic presentation demonstrating the status of household electrical wiring offers comfortable user control when controlling household functions such as management of energies and option to use in-house safety features—which can be hardly achieved by regular electrical systems.

1.1 Current Status

Many research centers are focused with developing activities, that aims is to build up a smart home environment, where people with disabilities can improve their abilities to cope with daily life activities by means of technologically advanced home automation solutions [1]. For the control of operational and technical functions in intelligent buildings are used some comfort remote control user interfaces like for example channel service for the digital home oriented textile consumption, which analyses the system requirement involved and develops a prototype to demonstrate the framework of the set-top box embedded simulation system for functional textile products consumption [2]; project aims at developing a new user friendly technology for home automation based on voice command [3, 4]; the service pattern-oriented smart bedroom based on elderly spatial behaviour patterns [5]. The aim of the study with design of wireless technology in smart home is to assess older adults' perceptions of specific smart home technologies (i.e., a bed sensor, gait monitor, stove sensor, motion sensor, and video sensor) [6]. Very important is to respect of security, privacy, and dependability in developing smart homes technologies [7] control with view to the senior citizeńs needs [8–10] and with power saving [11].

2 Operational and Technical Functions Control

Visualization application programme used to control technical functions in smart houses/buildings must provide communication between the user (client side), the visualization application programme, and the controlled components offered by xComfort wireless system (approach from the server side) (Fig. 1).

Client's approach is designed to allow the user to control actuators/components of the xComfort wireless system using a web browser—through a network or via the Internet. A client here represents a web browser used by a person to access the application. Information provided to the person using a web browser is dynamically loaded from a database—in our case from MS SQL (Microsoft Server Structured Query Language). Information between the controlling computer and the client is transferred via HTTP protocol. HTML (HyperText Markup Language)

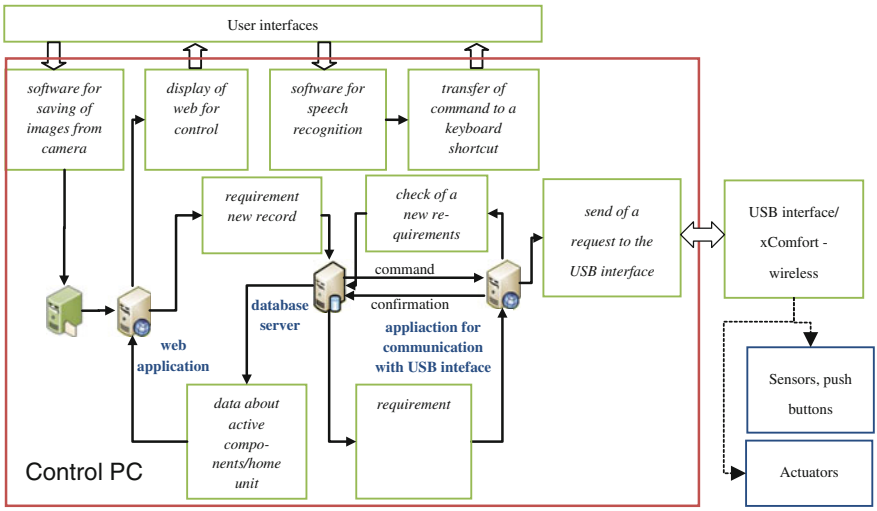


Fig. 1 The flow diagram of visualization of operational and technical functions in the intelligent building service with wireless components xComfort

and CSS (Cascading Style Sheets) are used to structure the document. JavaScript is used to provide higher form of comfortable control. System components chosen on the server side are selected based on new trends, scalability and on future expandability of the system. Server side refers to products which allow the application used to control smart electrical installation run, through the use of components of the xComfort wireless system. To do so, two programmes are used. Smart Home App and USBinterface Transfer. Smart Home App has been created in ASP.NET (Active Server Pages. Network) environment and USBinterface Transfer in NET. Both programmes were created in C# language/code. In order to create own visualization application programme, is necessary to perform analysis of software environment used for communication between the user and the control system as well as analysis of data processing method.

3 Software Analysis

3.1 Analysis of Smart Home App Web Visualization

Web visualization smart Home App is used by the user to control intelligent electrical system components. The key function is to control actuators through direct changes in the electrical system and collective/bulk modification of the electrical system (Fig. 2).

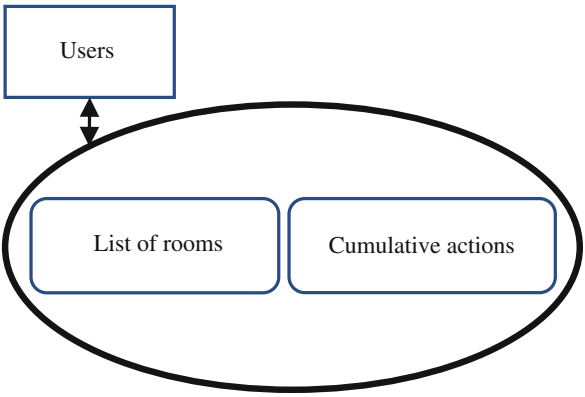


Fig. 2 DFD diagram 0 Level

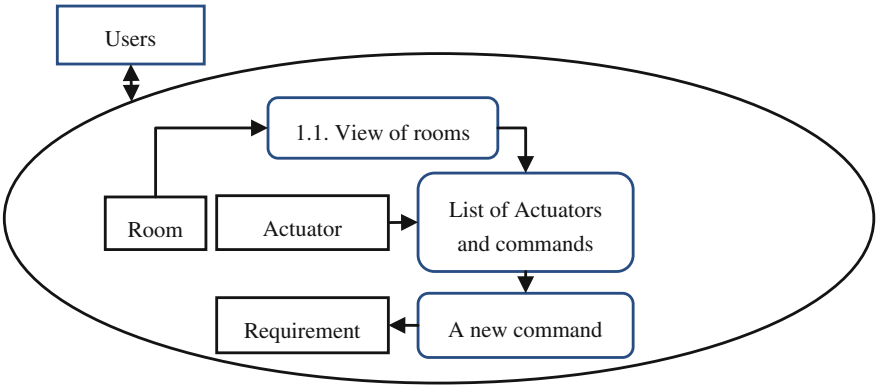


Fig. 3 First level DFD diagram showing control of active elements in a room

The method used to control active components in a room is shown in the following picture (Fig. 3):

1. Load all rooms in the database from Room table (Table 5), assign room.id to each room.
2. Load all active elements in the database from Actuator table (Table 6) which have idRoom = room.id, and display allowed actions for each Actuator type—according to Actuator type.
3. Save a new command to the Requirement table (Table 1) based on the communication protocol.

Bulk/collective control of active elements in the electrical system is done as follows (Fig. 4):

Table 1 The data dictionary for the table requirement

Requirement					
Attribute	Data type	Null	Key	Index	I/O describe
idReqAcc	Int(10)	No	Yes	Yes	Unique scheme key
bit0	bit	No	No	No	value of bit 0
bit1	bit	No	No	No	value of bit 1
bit2	bit	No	No	No	value of bit 2
bit3	bit	No	No	No	value of bit 3
bit4	bit	No	No	No	value of bit 4
bit5	bit	No	No	No	value of bit 5
bit6	bit	No	No	No	value of bit 6
bit7	bit	No	No	No	value of bit 7
bit8	bit	No	No	No	value of bit 8
executed	bool	No	No	No	Information about the command execution
datatimeReq	datatime	No	No	No	Date and time receiving the command

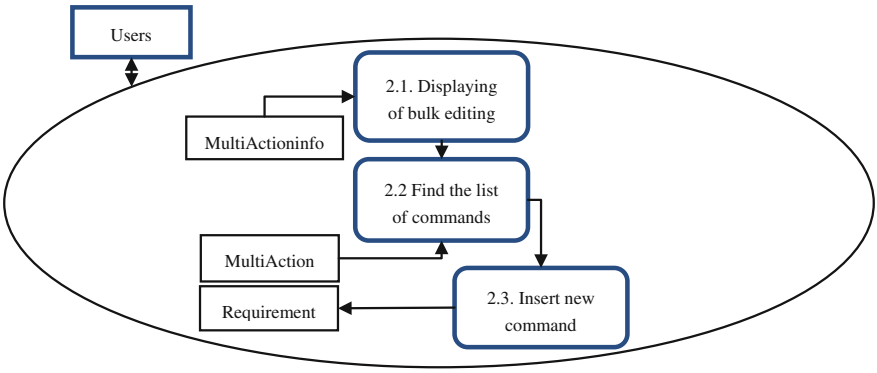


Fig. 4 First level DFD diagram showing control of active elements using bulk/collective setup of electrical system

1. Load all records from MultiActionInfo table (Table 3), and assign MultiActionInfo.idma to each record.
2. Load all records from MultiActiontable (Table 4), where MultiActionInfo.idma is the same as MultiActionInfo.idma.
3. Enter records in the Requirement table (Table 1).

3.2 Data Analysis

Data analysis is described through a conceptual database diagram (Fig. 5) and through a database scheme (Fig. 6).

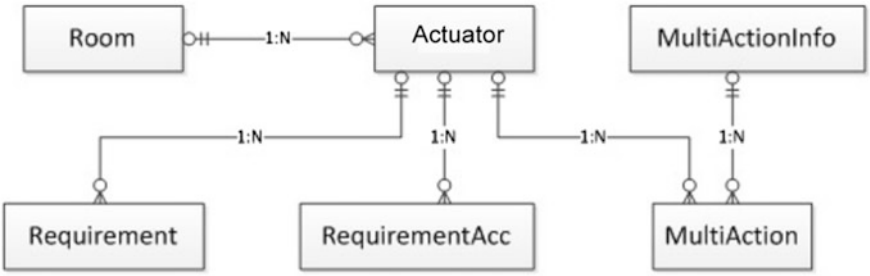


Fig. 5 Conceptual database diagram

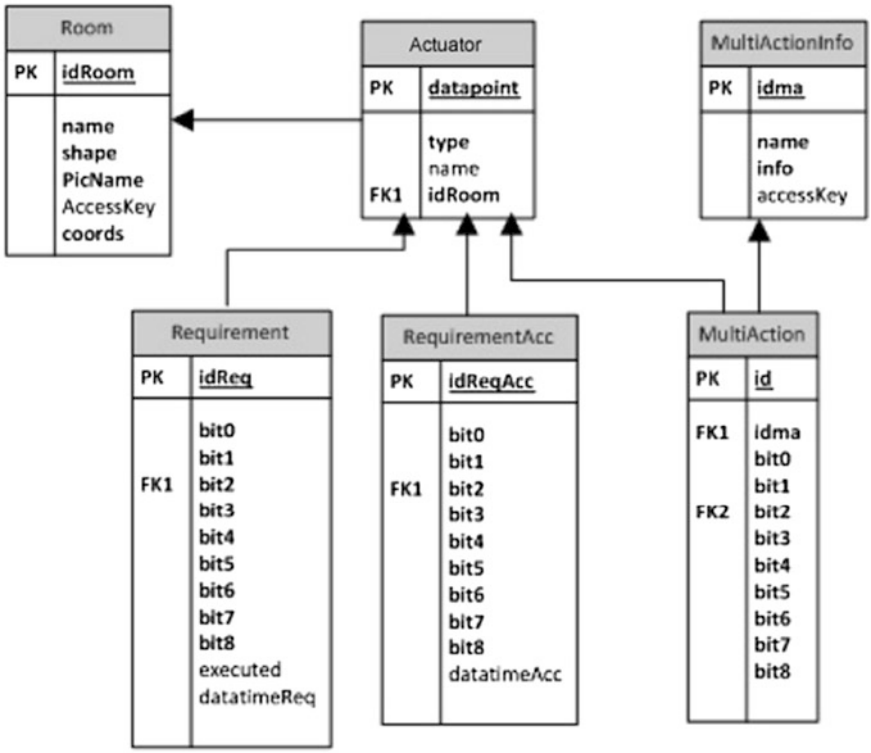


Fig. 6 Database scheme

Table 2 The data dictionary for the table RequirementAcc

RequirementAcc					
Attribute	Data type	Null	Key	Index	I/O describe
idReqAcc	Int(10)	No	Yes	Yes	Unique scheme key
bit0	bit	No	No	No	value of bit 0
bit1	bit	No	No	No	value of bit 1
bit2	bit	No	No	No	value of bit 2
bit3	bit	No	No	No	value of bit 3
bit4	bit	No	No	No	value of bit 4
bit5	bit	No	No	No	value of bit 5
bit6	bit	No	No	No	value of bit 6
bit7	bit	No	No	No	value of bit 7
bit8	bit	No	No	No	value of bit 8
datatimeReq	datatime	No	No	No	Date and time about the command execution

Linear recording of types of entities and links is done as follows:

```
Room(idRoom, name, shape, cords, PicName, accessKey)
Actuator(datapoint, type, name, idRoom )
Requirement(IdReq, bit0, bit1, bit2 , bit3, bit4, bit5,
bit6, bit7, bit8, executed, datatimeReq )
RequirementAcc(IdReqAcc, bit0, bit1, bit2 , bit3, bit4,
bit5, bit6, bit7, bit8, datatimeAcc )
MultiAction(id, idma , bit0, bit1, bit2 , bit3, bit4,
bit5, bit6, bit7, bit8)
MultiActionInfo(idma, name, info, accesKey)
```

_____ - scheme key,
…….. - foreign key.

Data dictionaries/terms used in "Requirement" table (Table 3), "RequirementAcc" table (Table 2), "MultiActionInfo" table (Table 3), "MultiAction" table (Table 4) and "Room" table (Table 5) are also described.

3.3 USBInterface Transfer Software Analysis

USBinterface Transfer tool is used to send requests to USB interface. USB interface will send requests to active elements in the intelligent electrical system. The programme repeatedly searches for incomplete/unexecuted records and then marks them as completed/done. It also captures execution confirmations and enters new records describing execution of orders/requests based on the following procedure (Fig. 7):

Table 3 The data dictionary for the table MultiActionInfo

MultiActionInfo					
Attribute	Data type	Null	Key	Index	I/O describe
idmac	Int(5)	No	Yes	Yes	Unique scheme key
Name	varchar(50)	No	No	No	Name
Info	varchar(150)	No	No	No	Text informing about the importance
AccessKey	char(1)	Yes	No	No	Key shortcut

Table 4 The data dictionary for the table MultiAction

MultiAction					
Attribute	Data type	Null	Key	Index	I/O describe
id	Int(10)	No	Yes	Yes	Unique scheme key
idma	Int(5)	No	No	No	The unique identifier of the mass action
bit0	bit	No	No	No	value of bit 0
bit1	bit	No	No	No	value of bit 1
bit2	bit	No	No	No	value of bit 2
bit3	bit	No	No	No	value of bit 3
bit4	bit	No	No	No	value of bit 4
bit5	bit	No	No	No	value of bit 5
bit6	bit	No	No	No	value of bit 6
bit7	bit	No	No	No	value of bit 7
bit8	bit	No	No	No	value of bit 8

Table 5 The data dictionary for the table room

Room					
Attribute	Data type	Null	Key	Index	I/O describe
idRoom	Int(5)	No	Yes	Yes	Unique scheme key
Name	varchar(50)	No	No	No	Name
Shape	varchar(50)	No	No	No	Shape of the area
Coords	varchar(50)	No	No	No	Display points of the room area
PicName	varchar(50)	No	No	No	The image name for rooms display
AccessKey	char(1)	Yes	No	No	Keyboard shortcut for the room

Table 6 The data dictionary for the table actuator

Actuator					
Attribute	Data type	Null	Key	Index	I/O describe
Datapoint	bit	No	Yes	Yes	The unique identifier of the active element
Type	bit	No	No	No	Type of the active element
Name	varchar(50)	No	No	No	Name
idRoom	Int(5)	No	No	Yes	The unique identifier of the room

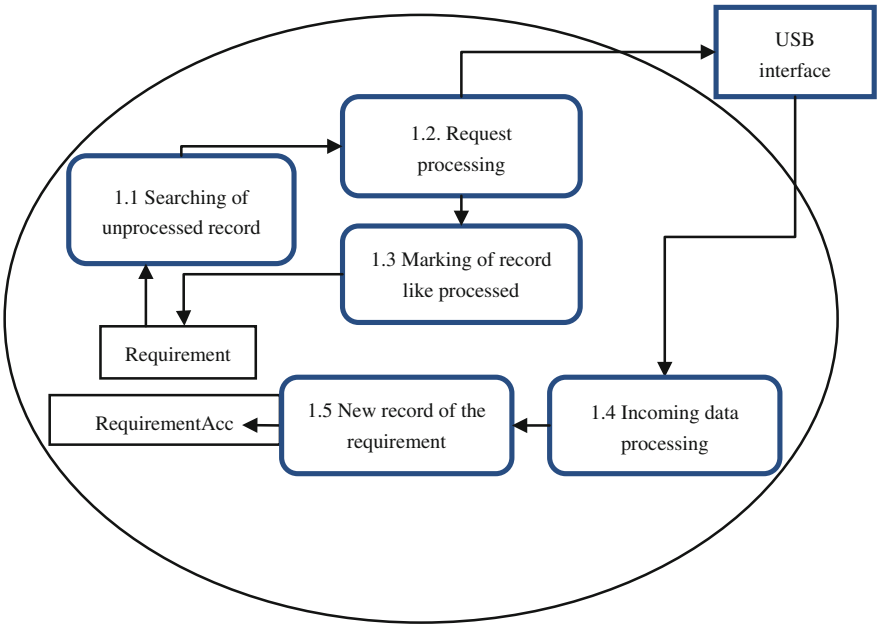


Fig. 7 DFD diagram of USBinterface transfer

1. Load one record from Requirement table (Table 1), executed = 0.
2. Enter the record into the data field and send the field to USB interface.
3. Perform UPDATE of the Requirement table (Table 1), executed =1.
4. Enter data in the field, USB interface incoming data.
5. Execute INSERT data from data field.

4 Conclusion

This visualization application is a software tool used to control operational and technical functions of the relevant electrical system. Based on the process described in the analysis this tool may be created using various programming languages/codes. Rapid development of technologies installed in households creates a trend where one household is equipped with different technologies used to control various technical and operational functions. Therefore, integration of all various and installed technologies has become an important focus of many manufacturers producing such devices. Radiofrequency electrical systems may be used to achieve such integration thanks to visualization of operational and technical control functions used in intelligent buildings. The article describes analysis method for smart Home App web visualization, data analysis, and analysis of

USBinterface Transfer software tool used to create visualization application environment which is used to communicate with the user and the control system and to processes data in intelligent (smart) buildings providing nursing and assistance services for handicapped people and for the elderly.

Acknowledgments This paper has been elaborated in the framework of the project Opportunity for young researchers, reg. no. CZ.1.07/2.3.00/30.0016, supported by Operational Programme Education for Competitiveness and co-financed by the European Social Fund and the state budget of the Czech Republic. This work was supported by project SP2014/156, "Microprocessor based systems for control and measurement applications." of Student Grant System, VSB-TU Ostrava.

References

1. Andrich, R., Gower, V., Caracciolo, A., Del Zanna, G., Di Rienzo, M.: The DAT project: a smart home environment for people with disabilities. In: Miesenberger, K., Klaus, J., Zagler, W., Karshmer, A. (eds.) Proceedings of Computers Helping People with Special Needs. pp. 492–499. (2006)
2. Su, Z.L., Wang, R.M., Wang, Z., Xie, H.J.: Channel service for the digital home oriented textile consumption. Text. Bioeng. Inform. Symp. Proc. **1–3**, 154–158 (2010)
3. Portet, F., Vacher, M., Golanski, C., Roux, C., Meillon, B.: Design and evaluation of a smart home voice interface for the elderly: acceptability and objection aspects. Pers. Ubiquit. Comput. **17**, 127–144 (2013)
4. Vanus, J., Koziorek, J., Hercik, R.: The design of the voice communication in smart home care. 36th International Conference on Telecommunications and Signal Processing, TSP 2013, art. no. 6613996, pp. 561–564. (2013)
5. Lee, H., Park, S.J., Kim, M.J., Jung, J.Y., Lim, H.W., Kim, J.T.: The service pattern-oriented smart bedroom based on elderly spatial behaviour patterns. Indoor Built Environ. **22**, 299–308 (2013)
6. Demiris, G., Hensel, B.K., Skubic, M., Rantz, M.: Senior residents' perceived need of and preferences for "smart home" sensor technologies. Int. J. Technol. Assess. Health Care **24**, 120–124 (2008)
7. Busnel, P., Giroux, S.: Security, privacy, and dependability in smart homes: a pattern catalog approach. In: Lee, Y., Bien, Z.Z., Mokhtari, M., et al. (eds.) Aging Friendly Technology for Health and Independence, pp. 24–31. (2010)
8. Vanus, J., Koziorek, J., Hercik, R.: Design of a smart building control with view to the senior citizenś needs. PDeS: IFAC Proceedings Volumes (IFAC-Papers Online) 12 (PART 1), pp. 411–415. Velké Karlovice, Czech Republic, Sept 25th–27th, (2013)
9. Penhaker, M., Stankus, M., Prauzek, M., Adamec, O., Peterek, T., Cerny, M., Kasik, V.: Advanced experimental medical diagnostic system design and realization. Elektronika Ir Elektrotechnika **117**, 89–94 (2012)
10. Machacek, Z., Slaby, R., Hercik, R., Koziorek, J.: Advanced system for consumption meters with recognition of video camera signal. Elektronika Ir Elektrotechnika **18**, 57–60 (2012)
11. Krejcar, O., Frischer, R.: Designing a real time redactor for power saving utilization. Elektronika Ir Elektrotechnika **19**, 59–64 (2013). doi:10.5755/j01.eee.19.6.4564

Visualization Software Designed to Control Operational and Technical Functions in Smart Homes

Jan Vanus, Pavel Kucera and Jiri Koziorek

Abstract To control operational and technical functions in Smart Homes using wireless system xComfort a visualization software application was developed. Visualization was created as a web application for operational data storage. In terms of communication between a database using visualization and active elements, a software driver was created which makes this communication possible. Visualization was made with regard to user requirements, web interface, ability to control the software through a mobile phone and also with regard to easy expandability, scalability and modularity.

Keywords Visualization · Wireless · Control · Smart home · Software

1 Introduction

Visualization software was created under a solution focusing on control of operational and technical functions in Smart Home using wireless system xComfort as a web application using DBMS (Database Management System) MS SQL (Microsoft Server Structured Query Language) to store operational data. In terms of communication between the database and the visualization and active elements, a software driver was designed, which makes this communication possible.

J. Vanus (✉) · P. Kucera · J. Koziorek
Department of Cybernetics and Biomedical Engineering, VSB TU Ostrava, 17. listopadu 15
708 33 Ostrava, Czech republic
e-mail: jan.vanus@vsb.cz

P. Kucera
e-mail: kucerapav@seznam.cz

J. Koziorek
e-mail: jiri.koziorek@vsb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_48,

Visualization was created with regard to web interface requirements, ability to control the software via a mobile phone and also with regard to easy expandability, scalability and modularity. Using modern technologies ASP.NET (Active Server Pages. Network), MS SQL, NET C# allowed us to meet all these requirements. Availability of optimized web visualization for various devices is possible thanks to the use of HTML 5 (HyperText Markup Language) and CSS 3 (Cascading Style Sheets) technologies. In addition to automatic saving process of requirements for active system element behaviour, data may also be used for presentation to demonstrate the history of the system use. Data may also be presented in a certain way as to allow later optimization of the entire intelligent electrical installation.

2 Current Status of the Research Dealing with Smart Home Visualization System

In order to determine the current status of the research dealing with intelligent building visualization systems, the following terms—which are related to these issues, were researched: “Smart”, “Home”, “Visualization”, “Building”, “Control”, “Monitoring”. Topics of selected articles may be divided into several areas of visualization systems implemented in Smart Home or Smart Home Care: wireless sensor network [1]; user reactions to health monitoring [2], physiological signal monitoring [3], consumption in smart living environments [4], smart camera - activity recognition [5], the visualization data processing [6], energy management in Smart Home [7], visualization with implemented RFID (Radio Frequency Identification) technology [8], 3D visualization [9] or Smart Grids technology [10]. Next is described design of technical solution of the visualization application software for control of operational and technical functions in Smart Homes with wireless technology xComfort. Very important is to respect of security, privacy, and dependability in developing smart homes technologies control with view to the senior citizeńs needs [11–13] and with power saving [14].

3 Describe of Technical Solution of the Visualization Application Software

It is a set of technical solutions covering various levels of controls, monitoring and visualization of statuses of controlled devices located in intelligent buildings using wireless system xComfort (Fig. 1).

One of the main parts of the system is SQL server, where control instructions are saved (Fig. 2). Control instructions are saved /recorded and each new record is recognized by the application and sent through control interface to the respective controlled elements. Then the controlled element sends information confirming

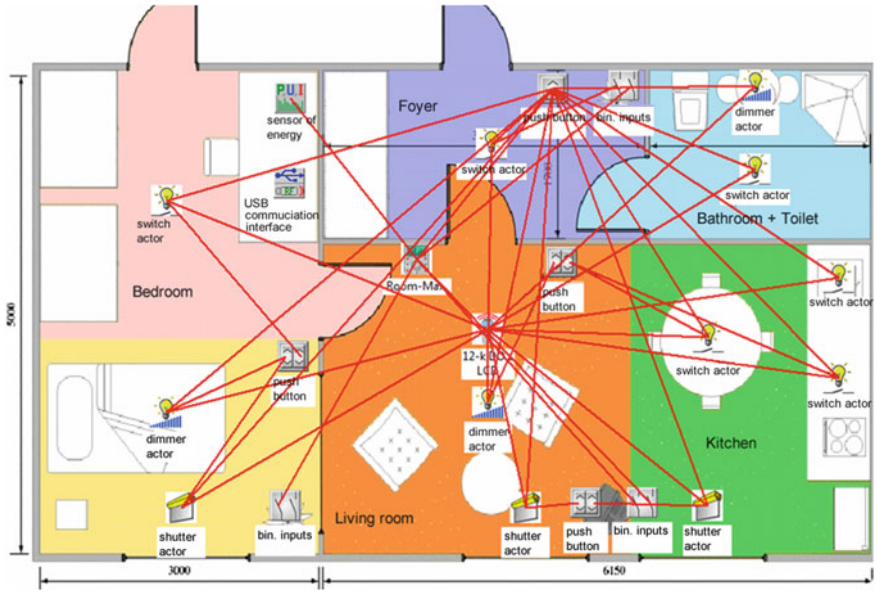


Fig. 1 Parameterised and interconnected actuators of xComfort wireless system on an apartment ground floor layout

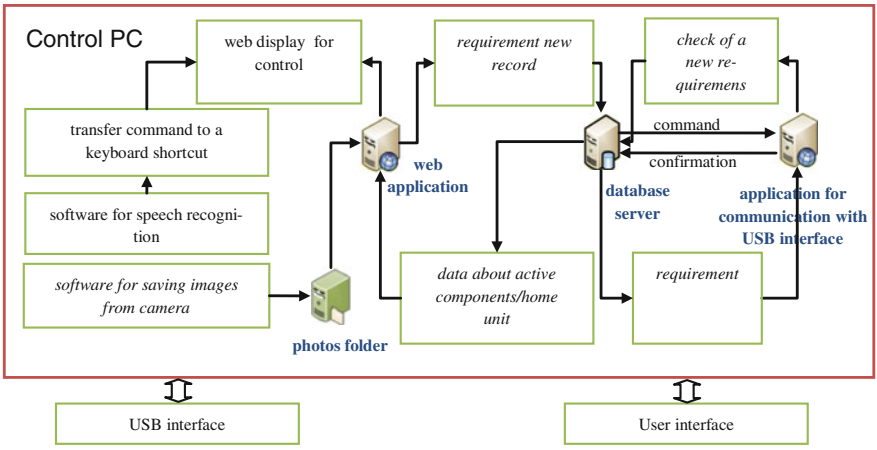


Fig. 2 Block diagram showing interconnection of software components in visualization application used to control operational and technical functions in smart home using xComfort wireless system

whether the request was actually carried out. If information specifying, that the requests was done is received, the status of the control device will change and the request is marked as completed. Here, the control unit represents visualization

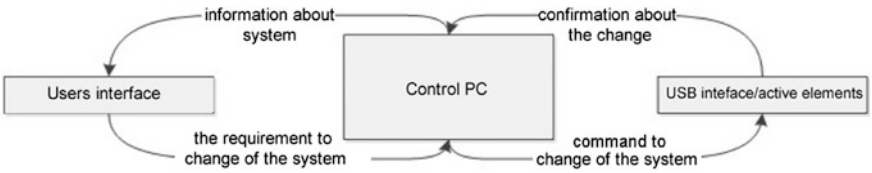


Fig. 3 System visualization block diagram

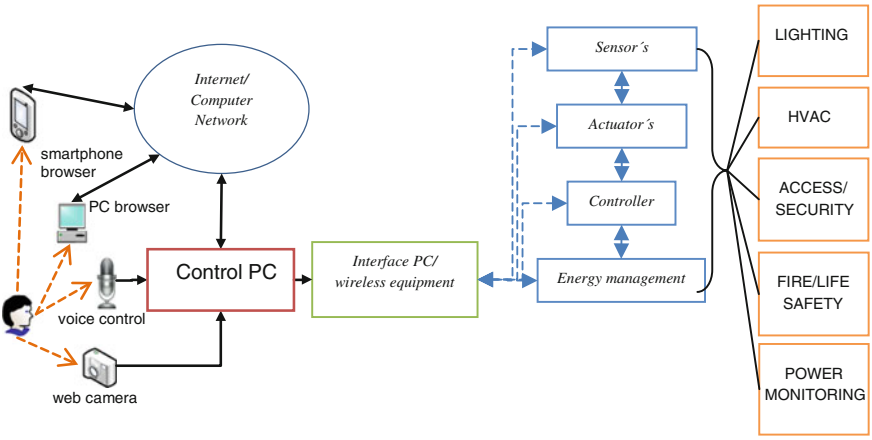


Fig. 4 Block diagram depicting a comfortable control of operational and technical functions using visualization environment

software, displays statuses of individual devices and options for their configurations. Thanks to the database server it is simple to monitor recorded instructions and processes, to evaluate results of individual instructions and also to find an optimal solution. Selected request definitions for visualization system, online access via Internet, control via USB interface and control requirements executed via mobile phone, these are all the reasons why we selected these technical elements.

Their mutual interconnection and system functions are described in Fig. 3. The system consists of three logical parts. User interface acts as a layer between the user and the controlling computer. It displays system status information and provides inspection and control elements used to control active elements in the apartment.

Control computer acts as a layer between a devise using USB interface and the user interface. It provides a comfortable environment for smooth operation of software elements and for active element control process provided by xComfort wireless system (Fig. 4).

Accepts and registers system requests. Checks and sends requests for system changes. Accepts change confirmations and forwards them to the system and to

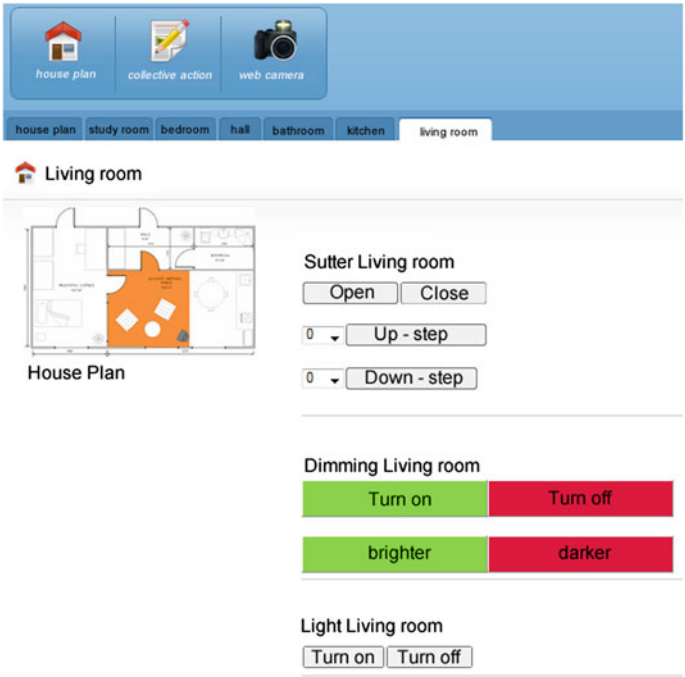


Fig. 5 A sample of visualization environment used to control lights and window blinds in a living room inside smart house using xComfort wireless system

user interface. It also stores information used later for optimization of system actuators installed in the apartment (Fig. 5).

USB interface sends requests to active elements in the apartment and receives answer which is then forwarded to the control computer. Thanks to mutual interconnections between these layers, we have a visualization system which controls xComfort radiofrequency system. Software and computer requirements:

- .Net Framework 4.0 or equivalent environment,
- MS SQL or other DBMS,
- IIS 6 (Internet Information Services),
- Windows 7 or higher version.

3.1 Visualization Implementation Process: Client

Selection of technical elements necessary for the respective visualization is directly dependent on our efforts to make actuator controlling accessible via web browser and through a computer network or the Internet. This chapter contains

descriptions and sample uses of technical elements, which were used to create visualization environment for the client/at the client's facility. Please note that client represents a web browser used by the user to access the application. If voice control system is used, it refers to an add-on programme, which controls the web browser. Information provided to the user of the relevant web browser is dynamically loaded from the database, in this situation from MS SQL. Transfer of information between the control computer and client is done via HTTP protocol. Document is structured using HTML and SCC. JavaScript is used as a higher form of a comfortable control system. Technologies used for visualization:

- language HTML,
- language CSS,
- language JavaScript.

HTML language includes a large set of characters, tags and attributes. Characters close documents sections into blocks which determine their meaning and therefore semantics. Attributes for labels determine other properties in terms of appearance, behaviour and possible connections [15]. CSS language also known as cascade style was created to separate appearance from the contents, as contents are addressed through HTML language and appearance through CSS. The cascade style allows change HTML tag display through attribute definition. All changes may be done at the same time (in bulk), if you want to affect the same tags according to their classes, if you want only certain tags to contain these changes, or if you wish to affect an individual tag directly, changes will be applied only to this particular tag. JavaScript is an object-oriented scripting language on the client side. It is most common use is to control user interface, where it controls the appearance of tags, their interaction, or possibly picture animation. Attribute name or id is used for connection, or some events, onClick [16] may also be used. Http protocol works based on "request-reply" system. Usually, the user sends a request for a document via a web browser, as plain text containing information about the browser, label of the desired document, authorization, and other information. Server then replies with a text describing the result of the request, document type and other information. Data of the requested document follow.

3.2 Sample of Use: Room Division

Visualization is done as a clickable map. Floor layout of the apartment unit serves as the base for the clickable map. Based on visualization requirements, the layout is divided into logical blocks according to rules and type of their use (Fig. 6), each room has its own light source and specific use. This division gives the user better view of rooms and simpler excess to the desired active element. Html code sample with JavaScript:

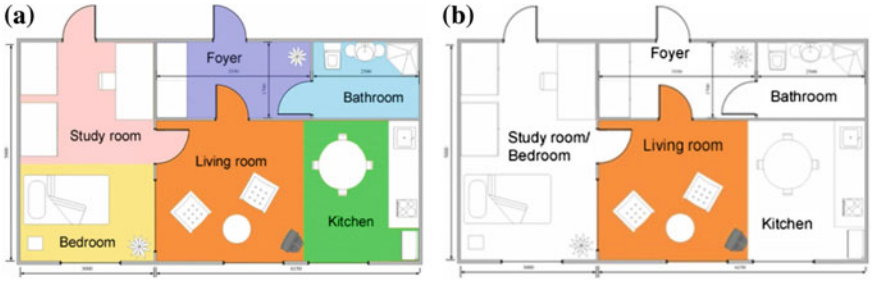


Fig. 6 The design of a visualisation environment for comfortable control of a building service system in smart home care with components of the wireless xComfort system. **a** floor projection; **b** floor projection—change of colours in visualization environment by moving of the PC mouse

```
<img
src=''roomImage/Aw.jpg''
id=''flat''
usemap=''#testroom''/>
<map name=''testroom»
<area
shape=''rect''
coords=''31,68,289,294''
onMouseOver=''document.getElementById ('flat').src=
'roomImage/workroom.png'''
onMouseOut=''document.getElementById ('flat').src='room-
Image/Aw.jpg'''
alt=''Pracovna''
href=''pracovna.html»
</map>
```

In real life scenario, values such as shape and coordinates are loaded from database. Individual rooms are assigned to individual floors. You may render multi-floor buildings. However, these are not in the sample apartment unit. When you place mouse cursor over the selected room the system displays coloured picture of the apartment unit for a particular room. Coloured pictures of rooms are placed on a white floor layout (Fig. 6b). This increases data transfer speed and at the same time it eliminates issues with layout size changes. It is necessary to point out that in real life scenario, this step will be done by technicians or by operator in real time. While performing these steps it is advisable to make your work easier and minimize errors in your work.

3.3 Voice Control

The MyVoice software is used for voice control. MyVoice was created for handicapped people to help them to use computer and information technologies. This software allows the user to control computer using voice commands only. Voice commands are set for particular keyboard commands, shortcuts or mouse actions. Commands may be combined in many ways allowing the user to create truly interactive environment with state-of-the-art voice control system. In order to create truly useful web application using MyVoice software, it is necessary to set keyboard shortcuts for all control elements and then assign relevant voice commands. If control elements are generated dynamically from the database, keyboard shortcuts for the final control elements must be stored in database as well. Sample of HTML language button with a keyboard shortcut:

```
<input
type=''submit''
name=''ctl00$MainContent$RepeaterActuatorList$8 Z''
value=''ZAPNOUT''
id=''MainContent RepeaterActuatorList 8 Z''
accesskey=''Z''
>
```

Then you set keyboard shortcut in MyVoice to Alt + Z, which turns lights ON and OFF when you say Czech word “zapnout” /”TURN ON”. When you say the command MyVoice presses the relevant keys and the button is engaged.

3.4 Mobile Version of Web Interface

Thanks to rapid Internet growth and thanks to huge development of technologies using HTML and CSS language we no longer need to create specialized versions of web documents for mobile devices. Thanks to HTML 5 and CSS 3, separate designs for mobile devices and for PC may be created. The division is done using media screen attribute, by selecting display size and by defining proper attributes, the web document will be rendered/displayed differently depending on the size of the device [15]. Sample of display divisions based on display size:

```
/* Smartphon --------- */
@media only screen
and (max-width : 320px) {
/*CCS pro mobilní zařízení*/
}
```

```
/* PC a notebook --------- */
@media only screen
and (min-width : 1224px)) {
/*CCS pro PC*/
}
```

This allows creation of different appearances on different devices as it is not desirable to display unnecessary contents on mobile devices. Displaying apartment unit is one of such examples. Control may also be displayed through direct references/links and therefore, it is not necessary to display the entire clickable map on a mobile device. Undesirable elements for mobile versions are set using the attribute `display: none;`. This configuration for mobile devices will not display unnecessary contents.

3.5 Visualization Implementation Process: Server

System elements in server are selected based on new trends, scalability and easy system expandability. The server side refers to all logical software products ensuring proper operation of applications used to control xComfort intelligent electrical installations. To achieve this, two programmes are used: smartHomeApp and USBinterfaceTransfer. The first programme is created in ASP.NET environment, and the second one in.NET environment and both use C# language.

4 Conclusion

The visualization application programme described above is designed as a web application using DBMS MS SQL to save operational data. In terms of communication between the database and the visualization and active elements, a software driver was designed, which makes this communication possible. Visualization was designed with regard to web interface requirements, ability to control the software via a mobile phone and also with regard to easy expandability, scalability and modularity. Using modern technologies ASP.NET, MS SQL, NET C# allowed us to meet all these requirements. Availability of optimized web visualization application for various devices was achieved thanks to the use of HTML 5 and CSS 3 technologies. In addition to independent saving process of requirements affecting active system element behaviour, data may also be used for presentation demonstrating the history of the system use. Data may also be presented in a certain way as to allow later optimization of the entire intelligent electrical installation. Further, an option to integrate web cameras and voice control into the

system were also examined. The MyVoice software was used to control the visualization by voice and visualization requirements for integration between those two elements were also described.

Acknowledgments This paper has been elaborated in the framework of the project Opportunity for young researchers, reg. no. CZ.1.07/2.3.00/30.0016, supported by Operational Programme Education for Competitiveness and co-financed by the European Social Fund and the state budget of the Czech Republic. This work was supported by project SP2014/156, "Microprocessor based systems for control and measurement applications." of Student Grant System, VSB-TU Ostrava.

References

1. Basu, D., et al.: Wireless sensor network based smart home: Sensor selection, deployment and monitoring. In: Sensors Applications Symposium (SAS), 2013 IEEE. IEEE, pp. 49–54. (2013)
2. Beaudin, J.S., Intille, S.S., Morris, M.E.: To track or not to track: user reactions to concepts in longitudinal health monitoring. J. Med. Int. Res. **8**(4), e29 (2006)
3. Choi, A., Woo, W.: Daily physiological signal monitoring system for fostering social well-being in smart spaces. Cybern. Syst. **41**(3), 262–279 (2010)
4. Fercher, A.J., Hitz, M., Leitner, G.: Raising awareness of energy consumption in smart living environments. In: Royo D (Hrsg.): Proceedings of 5th International Conference on Intelligent Environments (IE'09). Fairfax (VA), pp. 91–98, IOS Press, Amsterdam, (2009)
5. Fleck, S., et al.: SmartClassySurv-a smart camera network for distributed tracking and activity recognition and its application to assisted living. In: Distributed Smart Cameras, 2007. ICDSC'07. First ACM/IEEE International Conference on. IEEE, pp. 211–218. (2007)
6. Ghidni, G., Das, S.K., Gupta, V.: Fuseviz: a framework for web-based data fusion and visualization in smart environments. In: Mobile Adhoc and Sensor Systems (MASS), 2012 IEEE 9th International Conference on. IEEE pp. 468–472. (2012)
7. Giacomin, J., Bertola, D.: Human emotional response to energy visualisations. Int. J. Ind. Ergon. **42**(6), 542–552 (2012)
8. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of things (IoT): a vision, architectural elements, and future directions. Future Gener. Comput. Syst. **29**(7), 1645–1660 (2013). (the International Journal of Grid Computing and Escience)
9. Shirehjini, A.A.N.: A novel interaction metaphor for personal environment control: direct manipulation of physical environment based on 3D visualization. Comput. Graph.-Uk **28**(5), 667–675 (2004)
10. Wojszczyk, B.: Progress in smart grid deployments global examples & lessons learned. In: Power and Energy Society General Meeting, 2012 IEEE. IEEE, p. 1–1. (2012)
11. Vanus, J., Koziorek, J., Hercik, R.: Design of a smart building control with view to the senior citizenś needs. In: IFAC Proceedings Volumes (IFAC-PapersOnline) 12 (Part 1), PDeS (2013), pp. 411–415, Velké Karlovice, Czech Republic, 25–27 Sept 2013
12. Penhaker, M., Stankus, M., Prauzek, M., Adamec, O., Peterek, T., Cerny, M., Kasik, V.: Advanced experimental medical diagnostic system design and realization: elektronika ir elektrotechnika, pp. 89–94, (2012)
13. Machacek, Z., Slaby, R., Hercik, R., Koziorek, J.: Advanced system for consumption meters with recognition of video camera signal. Elektronika Ir Elektrotechnika **18**, 57–60 (2012)
14. Krejcar, O., Frischer, R.: Designing a real time redactor for power saving utilization. Elektronika Ir Elektrotechnika **19**, 59–64 (2013). doi:10.5755/j01.eee.19.6.4564

15. McDonald, M., Freeman, A., Szpuszta, M.: HTML5 Audio Video. Zoner Press, Brno (2011). ISBN 978-80-7413-5
16. Svehring, S.: Javascript Krok Za Krokem. Computer Press, Brno (2008)
17. Nagel, Ch., Evjen, B., Glynn, J., Skinner, M., Watson, K.: C# 2008 Programujeme Profesionálně. Computer Press, Brno (2009). ISBN 978-80-251-2401-7
18. Brust, A.J.: Mistrovství v Programování SQL. Computer Press, Brno (2007). ISBN 979-80-251-1607-4

Using Analytical Programming and UCP Method for Effort Estimation

Tomas Urbanek, Zdenka Prokopova, Radek Silhavy and Stanislav Sehnalek

Abstract This article is aimed to using the analytical programming and the Use Case Points method to estimate time effort in software engineering. The calculation of Use Case Points method is strictly algorithmically defined, and the calculation of this method is simple and fast. Despite a lot of research on this field, there are many attempts to calibrating the weights of Use Case Points method. In this paper is described idea that equation used in Use Case Points method could be less accurate in estimation than other equations. The aim of this research is to create new method, that will be able to create new equations for Use Case Points method. Analytical programming with self-organizing migration algorithm is used for this task. The experimental results shows that this method improving accuracy of effort estimation by 25–40 %.

Keywords Analytical programming · Self-organizing migration algorithm · SOMA · Use case points · UCP · Effort estimation

1 Introduction

In software engineering, software effort estimation is the prediction of the work effort required to complete a software development project [1]. Wrong estimates may lead to one of two extreme results:

- Underestimate effort estimation
- Overestimate effort estimation.

T. Urbanek (✉) · Z. Prokopova · R. Silhavy · S. Sehnalek
Faculty of Applied Informatics, Tomas Bata University in Zlin, Nad Stranemi 4511, Zlin, Czech Republic
e-mail: turbanek@fai.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_49,

Underestimate effort estimation raises the requirements on the psyche of development team, raises the stress therefore the plan of software project have to be reconstructed, which may leads to raising the financial difficulties of development company. On the other hand, overestimate effort estimation reduces competiveness of company on the market, because the company permanently offers a software projects that are more expensive than the same software project developed by another company. Estimations in software engineering are very complex and very important process. Precision of the estimation is affected by many factors. These factors are for example experience of development team, used programming language, the size of development team, experience of project manager, the size of software project and other factors. Because of estimates are predictions of future actions, it is impossible to guarantee absolute precision of the estimation method. It could be spoken only about accuracy improvement of particular estimation method. The estimation methods are assumed that the software project will be produce according to plan, which is created and maintained by project manager. Unfortunately, random side-effects is entering into developed projects, which can affect the results of estimation. At the same time the estimations, which are precise and reliable are foundations of successful project management. Project managers have to do right decisions during the first stage of software development [2].

There is a fact that the Use Case Points method can offer a possibility of accuracy improvement through manipulation with equation presented by Gustav Karner [3]. For using the Use Case Points method is required a Use case diagram. This diagram is a foundation of software development and is created during the software projecting. First, the certain values are extracted from the Use case diagram. These values are written down to pre-prepared tables. Finally, the estimation is calculated from this tables. These tables also contain some values that known as weights. The weights are used for calibrating the Use Case Points method. Extracted values and weights are combined by certain equation presented by Gustav Karner. This equation is used to calculate the effort estimation.

1.1 Analytical Programming

Analytical programing (AP) is a tool for symbolic regression. The core of analytical programing is set of functions and operands. These mathematical objects are used for synthesis a new function. Every function in the set of analytical programming core has various number of parameters. Functions are sorted by these parameter into general function sets (GFS). For example GFS_{1par} contains functions that have only one parameter like sin(), cos() and other functions. AP must be used with any evolutionary algorithm that consists of a population of individuals for its run [4]. In this paper is used self-organizing migration algorithm

(SOMA) as evolutionary algorithm for analytical programming [5]. The function of AP is following:

A new individual is generated by evolutionary algorithm. Then this individual is remapped to new function by analytical programming. After that this new function is evaluated by cost function. Evolutionary algorithm decide either this new equation is suited or not for next evolution.

This implies the fact, that the analytical programming is a method, which converts input set of numbers to the function.

1.2 Single Estimator

According to work of Shepperd and Cartwright [6] is possible that couple of methods for effort estimation can not be compare in rank with each other, because of the input conditions may be changed.

Because of that is not possible to definitely decide, which method is better than other method [7]. In this paper, we agree with the work of Kocaguneli et al. [7], that ensemble of methods can have a better results than single methods. Nevertheless the ensemble of methods needs to be built by single methods. According to conclusions of work of Kocaguneli et al. [7], there are necessary to have a single methods also known as single estimators. This was also signal to creation of this work. Single estimator is a method for estimations that is classified as stand-alone method.

Hypothesis: **1.**
If we have a single estimator that returns us a more accurate estimate that is possible that this method is better for building ensembles of methods for effort estimation.

1.3 Taxonomy

The main goal of this work was to create a new single estimator. This estimator will be taught on the historical datasets.

Hypothesis: **2.**
There are dependencies between historical datasets and future predictions.

In taxonomical point of view can be this method classified into groups. According to work of Menzies et al. [8] it is possible to classify the estimation method to these groups:

- Model based
- Expert based.

Model based methods use algorithms to process historical dataset to provide estimations. Expert based methods use human experience to predict effort estimation. On these facts, this method can be classified into a model based effort estimation method. Compared to the work of Myrtveit et al. [9] that classifies effort estimation methods into two groups :

- Sparse-data methods
- Many-data methods.

Sparse-data methods use relatively small amount of historical data to provide a prediction. On the other hand many-data methods need a relatively large amount of historical data. The method presented in this article works with relatively small amount of historical data, however there is a possibility that this method can be more accurate with larger historical datasets. The last but not least, work of Shepperd and Schofield [10] classified the methods into three groups :

- Expert-based method
- Algorithmic model
- Analogy.

Expert-based methods use human experience with project management to provide effort estimation. Some expert-based methods use communication ability of project manager to arrange consensus, for example method of wide-band Delphi [11]. Algorithmic-based models use algorithms, which are process the datasets to provide effort estimations. Analogy-based methods search projects in databases and try to find a similar project that was been estimated before. According to this classification, presented method can be sorted into algorithmic-based models, because presented method process a historical datasets.

1.4 Related Work

In 1984 Boehm wrote his work [12], the author presents a software engineering economics challenges and COCOMO method [12], which is the widely accepted and used. In his work was compared a couple of effort estimation models that was existed in that years. Author mentions that despite of scatter and inaccurate datasets was done a lot of work in this field. Nevertheless in last years, software becomes important for mankind. This implies the fact, that we need more accurate effort estimations. This work is strongly inspired by Kocaguneli et al. [7], in this work author mentions, that ensemble of effort estimations are more effective and accurate than single estimators. Today, software managers have a lot of methods for effort estimations COCOMO [12], FPA [13], function points [14], UCP [3], wide-band delphi [11], and many other methods. Nonetheless in last years, researchers in this fields have powerful computational methods; these methods are generally called artificial intelligence. Artificial intelligence is often tested on COCOMO method, likewise in work of Attarzadeh and Ow [2] or Kaushik et al. [15]. Majority of work

Table 1 Data used for effort estimation

Project	C1	C2	C3	A1	A2	A3	TCF	ECF	UCP	Actual effort (h)	$UCP * 20$	Error (h)
A	23	8	0	0	0	4	0,92	0,78	148	3,037	2,960	77
B	6	5	0	0	2	2	0,75	0,81	55	1,917	1,100	817
C	7	4	0	0	0	2	0,90	1,05	76	1,173	1,520	347
D	13	5	1	0	0	3	0,85	0,89	105	742	2,100	1,358
E	13	2	0	0	0	4	0,82	0,79	63	614	1,260	646
F	10	0	0	0	0	3	0,85	0,88	44	492	880	388
G	10	0	0	0	0	2	0,78	0,51	22	277	440	163
H	23	19	0	0	1	4	0,94	1,02	304	3,593	6,080	2,487
I	17	0	0	0	0	4	1,03	0,80	80	1,681	1,600	81
J	26	0	0	0	0	4	0,71	0,73	74	1,344	1,480	136
K	10	3	0	0	0	3	1,05	0,95	89	1,220	1,780	560
L	14	0	0	0	0	4	0,78	0,79	50	720	1,000	280
M	6	0	0	0	2	0	0,96	0,96	31	514	620	106
N	16	2	0	0	0	5	0,90	0,91	95	379	1,900	1,521
Total Error												8,967

on this field relies on neural nets, which is highly suitable for calibrating effort estimation methods, likewise the work of Park and Baek [16], which use a method of function points with neural nets. This article proposed a new method of estimation with use of analytical programming and Use Case Points method. Numerous attempts were realized to improve accuracy of Use Case Points method via calibrating the weights using neural nets, likewise in work of Xia et al. [17] or Jiang et al. [18].

2 Problem Definition

Dataset with Use Case Points method was obtained from Poznan University of Technology [19]. The Table 1 shows Use Case Points method data from 14 projects. Data of Use Case Points method with transitions is used in this paper. There are 10 values for each software project. In the following text, there are used some abbreviations for Simple-T is now C1, Average-T is now C2, Complex-T is now C3, Simple is now A1, Average is now A2 and Complex is now A3.

Gustav Karner in his work [3] derived nominal value for calculation of man-hour from Use Case Points method. This value was set to 20. Thus, effort estimate in man-hours is calculated as $UCP * 20$. Now the error can be calculated.

Table 1 shows also calculated differences. The equation for this error calculation is following:

$$E = |ActualEffort - (UCP * 20)|, \tag{1}$$

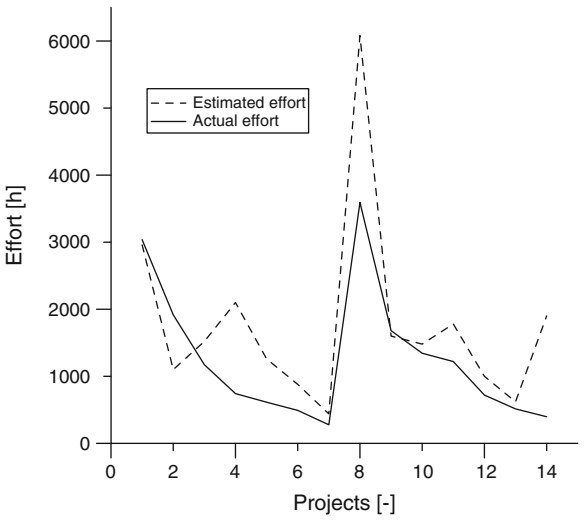


Fig. 1 Difference between estimated and real effort

where E is calculated error for each project in Table 1. Table 1 and Eq. (2) were used for calculation of total effort error.

$$E_t = \sum_{i=1}^{14} |E_i|, \tag{2}$$

where E_t is total error through all project in Table 1.

$$E_t = 8967$$

The conclusion is that, during estimation of 14 software projects was generated error by Use Case Points method and this error had value 8,967 man-hours. New method, which is presented in this paper, tries to minimize this error. As shown in Fig. 1, the Use Case Points method generate a significantly error in project D, H and N.

3 Method

Data set was obtained from Table 1. Matrix A was constructed from this dataset and has size $M \times N$, where $M = 9$ and $N = 14$. Every row of this matrix A contains calculation of Use Case Points method and actual effort.

$$A = \begin{bmatrix} 23 & 8 & 0 & 0 & 0 & 4 & 0,92 & 0,78 & 3,037 \\ 6 & 5 & 0 & 0 & 2 & 2 & 0,75 & 0,81 & 1,917 \\ 7 & 4 & 0 & 0 & 0 & 2 & 0,90 & 1,05 & 1,173 \\ 13 & 5 & 1 & 0 & 0 & 3 & 0,85 & 0,89 & 742 \\ 13 & 2 & 0 & 0 & 0 & 4 & 0,82 & 0,79 & 614 \\ 10 & 0 & 0 & 0 & 0 & 3 & 0,85 & 0,88 & 492 \\ 10 & 0 & 0 & 0 & 0 & 2 & 0,78 & 0,51 & 277 \\ 23 & 19 & 0 & 0 & 1 & 4 & 0,94 & 1,02 & 3,593 \\ 17 & 0 & 0 & 0 & 0 & 4 & 1,03 & 0,80 & 1,681 \\ 26 & 0 & 0 & 0 & 0 & 4 & 0,71 & 0,73 & 1,344 \\ 10 & 3 & 0 & 0 & 0 & 3 & 1,05 & 0,95 & 1,220 \\ 14 & 0 & 0 & 0 & 0 & 4 & 0,78 & 0,79 & 720 \\ 6 & 0 & 0 & 0 & 2 & 0 & 0,96 & 0,96 & 514 \\ 16 & 2 & 0 & 0 & 0 & 0 & 0,90 & 0,91 & 379 \end{bmatrix} \quad (3)$$

The columns from beginning to end are C1, C2, C3, A1, A2, A3, TCF, ECF and actual effort. Whole dataset could not be optimized by evolutionary algorithm, because no data was remained for testing purposes. Because of this problem, the matrix A was divided into two matrices. Matrix B is training dataset and matrix C is testing dataset.

Hypothesis: **3.**
If the difference between training data and actual effort is minimized by analytic programming, the difference between testing data and actual effort will be minimized too.

The Matrix B contains 9 rows for training purposes and the matrix C contains 5 rows for testing purposes. The matrix B was processed by analytical programming with self-organizing migration algorithm. Result of this process was a new equation. This equation contained variables and constants and these variables were C1, C2, C3, A1, A2, A3, TCF and ECF. This new equation also describes relationships between variables in matrix B, moreover in matrix C.

$$B = \begin{bmatrix} 23 & 8 & 0 & 0 & 0 & 4 & 0,92 & 0,78 & 3,037 \\ 6 & 5 & 0 & 0 & 2 & 2 & 0,75 & 0,81 & 1,917 \\ 7 & 4 & 0 & 0 & 0 & 2 & 0,90 & 1,05 & 1,173 \\ 13 & 5 & 1 & 0 & 0 & 3 & 0,85 & 0,89 & 742 \\ 13 & 2 & 0 & 0 & 0 & 4 & 0,82 & 0,79 & 614 \\ 10 & 0 & 0 & 0 & 0 & 3 & 0,85 & 0,88 & 492 \\ 10 & 0 & 0 & 0 & 0 & 2 & 0,78 & 0,51 & 277 \\ 23 & 19 & 0 & 0 & 1 & 4 & 0,94 & 1,02 & 3,593 \\ 17 & 0 & 0 & 0 & 0 & 4 & 1,03 & 0,80 & 1,681 \end{bmatrix} \quad (4)$$

$$C = \begin{bmatrix} 26 & 0 & 0 & 0 & 0 & 4 & 0,71 & 0,73 & 1,344 \\ 10 & 3 & 0 & 0 & 0 & 3 & 1,05 & 0,95 & 1,220 \\ 14 & 0 & 0 & 0 & 0 & 4 & 0,78 & 0,79 & 720 \\ 6 & 0 & 0 & 0 & 2 & 0 & 0,96 & 0,96 & 514 \\ 16 & 2 & 0 & 0 & 0 & 5 & 0,90 & 0,91 & 379 \end{bmatrix}. \quad (5)$$

3.1 Cost Function

The new function that is generated by analytical programming contains these parameters C1, C2, C3, A1, A2, A3, TCF and ECF. There is no force applied to analytical programming that equations generated by analytical programming have to contain all of these parameters. Cost function that is used for this task is following:

$$CF = \sum_{i=1}^{n} \left| B_{n,9} - f\ ce(B_{n,1}, B_{n,2}, \ldots, B_{n,8}) \right|. \quad (6)$$

4 Results

The $n = 1{,}000$ calculations were generated by analytical programming. That means, 1,000 equations were created and tested. Some of these equations were removed for in appropriate pathological structure for example missing parameters and other mistakes. The parameters of self-organizing migrating algorithm were set according to Table 2.

Each calculation was generated in approximately 1 min and 10 s. That means, the total calculation of 1,000 calculations were taken approximately 18 h. During this 18 h the $n = 1{,}000$ function were generated, and only 5 of them was found as the best results (Fig. 2).

The Table 3 contains the results of cost function of the 5 best results. It is very important that cost function for evolutionary algorithm was applied only for training data. The cost function of testing data was calculated only for testing purposes.

The total error between testing data and actual effort is $E_t = 2603$ man-hours calculated from Table 1. The Table 4 shows the total error of each equation. This table also shows the estimation improvement of each equation.

As can be seen in Table 4 that the best result is provided by Eq. (1). This equation is shown as Eq. (7), and improves the accuracy of Use Case Points method by 40 %.

Table 2 Setup of self-organizing migration algorithm

Parameter	Value
Path length	3
Step	0.3
PRT	0.7
Pop size	30
Migrations	30

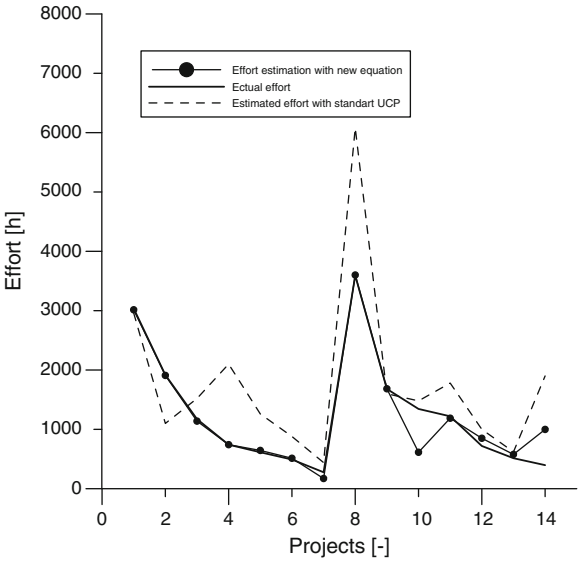


Fig. 2 Comparison between classic use case points method and improved use case points method presented in this paper

Table 3 Calculated cost function for each equation

Equation	CF training data	CF testing data
1	232.443	1557.92
2	194.617	1867.29
3	155.124	1903.95
4	1.52932	1977.76
5	472.826	1946.91

Table 4 Errors on testing data and improvement of each equation

Equation	CF testing data	Improvement (%)
1	1557.92	40
2	1867.29	28
3	1903.95	27
4	1977.76	24
5	1946.91	25

$$
\begin{aligned}
UCP = (&-12.6909 + A1 + TCF + C2 \\
&+ (-10.0616 + C2) * (-1.2091 + C3)) \\
&* (304.916 + 47.6424 * A3^3 - C1 - C3) \\
&+ (-36.4125 + A1 - ECF + C2) \\
&* (-ECF + TCF + C1 + C2) \\
&* (-A2 - C2 + C3 - C1 * C3)
\end{aligned}
\quad (7)
$$

5 Conclusion

In this paper was presented new method for effort estimation improvement. This method is combination of Use Case Points method and analytical programming with self-organizing migration algorithm. Presented method is founded on generating new equations for Use Case Points method. From 1,000 calculated equation was chosen only 5 best equations that had results that improving estimation. Nonetheless, the estimation in software engineering is not a real-time application and these equations can be generated only once at time. Although, the self-organizing migration algorithm is very sensitive to input parameters, there will be necessary to find a proper setup for this type of application. There is also a problem in this method; the cost function minimization may not lead to minimization of error in estimation on data that is not subject of minimization. That means, the equations have to be checked by visual or some kind of algorithm. There is another disadvantage, this method still depend on human experience with Use Case Points method. The benefit of this solution is there are no weights in this method, because these weights are generated by analytical programming as constants in equations. Another benefit is that this method need relatively little amount of data. The subject of further research will be that there is a possibility that this method can generate more accurate equations with larger datasets. Table 3 is partially proved the Hypothesis 3. Although equation four in Table 3 had cost function in training data 1.53 and in testing data 1,977.7 man-hours, which is the worst result. Nevertheless, the fourth equation is still estimation improving equation.

Acknowledgment This study was supported by the internal grant of TBU in Zlin No. IGA/FAI/2013/032 and No. IGA/FAI/2013/039 funded from the resources of specific university research.

References

1. Keung, J W.: Theoretical maximum prediction accuracy for analogy-based software cost estimation. 2008 15th Asia-Pacific Software Engineering Conference, pp. 495–502 (2008)
2. Attarzadeh, I., and Ow, S.: 'Software Development Cost and Time Forecasting Using a High Performance Artificial Neural Network Model', in Chen, R. (Ed.): 'Intelligent Computing and Information Science' (Springer Berlin Heidelberg, 2011), pp. 18–26

3. Karner, G.: Metrics for Objectory. Diploma thesis, University of Linköping, Sweden. No. LiTH-IDA-Ex-9344:21. (1993)
4. Kominkova Oplatkova, Z., Senkerik, R., Zelinka, I., Pluhacek, M.: Analytic programming in the task of evolutionary synthesis of a controller for high order oscillations stabilization of discrete chaotic systems. Comput. Math. Appl. **66**(2), 177–189 (2013)
5. Zelinka, I.: SOMA—Self Organizing Migrating Algorithm in New Optimization Techniques in Engineering, pp. 167–218. Springer, Berlin (2004)
6. Shepperd, M., Cartwright, M.: Predicting with sparse data. IEEE Trans. Softw. Eng. **27**(11), 987–998 (2001)
7. Kocaguneli, Ekrem, Menzies, T., Keung, J.: On the value of ensemble effort estimation. IEEE Trans. Softw. Eng. **38**(6), 1403–1416 (2011)
8. Menzies, T., Chen, Z., Hihn, J., Lum, K.: Selecting best practices for effort estimation. IEEE Trans. Softw. Eng. **32**(11), 883–895 (2006)
9. Myrtveit, I., Stensrud, E., Shepperd, M.: Reliability and validity in comparative studies of software prediction models. IEEE Trans. Softw. Eng. **31**(5), 380–391 (2005)
10. Shepperd, M., Schofield, C.: Estimating software project effort using analogies. IEEE Trans. Softw. Eng. **23**(12), 736–743 (1997)
11. Stellman, A., Greene, J.: Applied Software Project Management, 1st edn. O'Reilly Media, Sebastopol (2005)
12. Boehm, B.W.: Software engineering economics. IEEE Trans. Softw. Eng. **10**(1), 4–21 (1984)
13. Albrecht, A.J., Gaffney, J.E.: Software function, source lines of code, and development effort prediction: a software science validation. IEEE Trans. Softw.Eng. **9**(6), 639–648 (1983)
14. Atkinson, K., Shepperd, M.: Using function points to find cost analogies. In: 5th European Software Cost Modelling Meeting, Ivrea, Italy, pp. 1–5, 1994
15. Kaushik, A., Soni, A.K., Soni, R.: An adaptive learning approach to software cost estimation. 2012 National Conference on Computing and Communication Systems, 1–6 Nov 2012
16. Park, H., Baek, S.: An empirical validation of a neural network model for software effort estimation. Expert Syst. Appl. **35**(3), 929–937 (2008)
17. Xia, W., Capretz, L.F., Ho, D., Ahmed, F.: A new calibration for function point complexity weights. Inf. Softw. Technol. **50**(7–8), 670–683 (2008)
18. Jiang, Z., Naudé, P., Jiang, B.: The effects of software size on development effort and software quality. J. Comput. Inf. Sci. **1**(4), 492–496 (2007)
19. Ochodek, M., Nawrocki, J., Kwarciak, K.: Simplifying effort estimation based on use case points. Inf. Softw. Technol. **53**(3), 200–213 (2011)

Optimizing the Selection of the Die Machining Technology

Florin Chichernea

Abstract The selection of a material for an application in engineering or its replacement with another material, superior in terms of economics, engineering and environmental impact is an important stage in the design process of a product. The paper presents a modern and original method for optimizing the selection of a manufacturing process for a part, for maximizing its performance and minimizing its cost, to attain the sustainable development objectives. The work strategy involves setting the functions of the product, the matrix and the related programs to select the optimal technology, applying the value analysis approach in order to obtain an optimal design—machining process. For the automation of calculations and ease of design work, the author developed calculus programs.

Keywords Value · Modelling · Value analysis · Optimizing · Design · Machining process · Selection strategies

1 Introduction

The Value Analysis Methodology was born in 1947 at General Electric. Faced with a shortage of strategic materials, the company management asked L. D. Miles to identify new materials that cost less. At that point he gradually set in practice a rigorous plan followed by a 40 % reduction of costs [1]. Value Analysis was quickly used in industries facing economic and strategic deficiencies.

The guideline of Value Analysis is the Function Analysis. Starting from the idea that a product is purchased because it performs something that matches a buyer's need, this property was called main function. For the main function to be performed,

F. Chichernea (✉)
Department of Materials Science, Transilvania University of Brasov,
Blvd. Eroilor, Nr. 29 500036 Brasov, Romania
e-mail: chichernea.f@unitbv.ro

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_50,

to the product should be added a series of secondary functions, which are of interest only when they contribute to the normal performance of the main function. It is estimated that, overall, only 20 % of the manufacturing costs of the products are caused by the core functions and 80 % by the secondary functions [2–4].

Only the product bears value and its subassemblies or components contribute to the usefulness of the product [5–7].

Based on experience in the field, on the relevant examples of applying the Value Analysis approach to products, the author propose a new method for optimizing the selection of design—manufacturing technologies for various parts.

The optimization is achieved by setting the functions of the constituent elements, of the operations that lead to changing the structure of the design—machining technological processes accompanied by a reduction in costs without altering the performances.

Forging dies are devices employed in the fields where cold or hot plastic deformation is used for processing in order to obtain large quantities of semi-products.

2 Materials and Semi-products Employed

The dies used to manufacture forged semi-products are made of alloyed steels highly resistant to shocks, such as 34MoCN15 or 41MoC11 (STAS 791-88; EN 10083), to which, after the roughing processing, an improvement heat treatment is applied, which results in a hardness of 32–36 HRC.

The employed semi-products are freely forged from laminated semi-products, for the small sized dies, or from cast semi-products for the large sized dies. The free forging operation must be performed with extreme care in order to avoid cracks [8].

3 Technological Processes

The first technology used to manufacture dies was based on the method of manual engraving using chisels. Preliminarily there was performed a roughing processing for the die slot using various technological procedures (milling, drilling, turning). This procedure was replaced (for dies) with other technologies when the copying milling and the electro erosion processing machines appeared.

The engraving method requires a lengthy period and highly qualified workers and the dimensional and shape precision for the die slot is small.

Depending on the equipment existing in the workshop, there can be adopted one of the technological processes presented below [8].

For forging dies there are used the variants of manufacturing technological processes presented in Tables 1, 2, 3, and 4. Finishing the slot (obtained by

Table 1 Technological process of dies forging, variant 1

Name of operation
1-Steel ingot casting 34MoC15
…
16-Closing the two semi-dies and obtaining the final control part. Checking the control part. Checking the other dimensions of the die

Table 2 Technological process of dies forging, variant 2

Name of operation
Same as first variant, including operation 13, then the operations below may follow
…
16-Closing the two semi-dies and obtaining the control part. Final check

Table 3 Technological process of dies forging, variant 3

Name of operation
Same as first variant, including operation 13, then the operations below may follow
…
16-Final check

Table 4 Technological process of dies forging, variant 4

Name of operation
Same as first variant, including operation 7, then the operations below may follow
…
12-Final check

copying milling), using electro erosion and then manually, allows for smaller shape and dimension deviations, and a continuous surface is obtained.

Complete machining, roughing and finishing the slot, using electro erosion, offers the highest dimensional accuracy but requires appropriate machines and using at least two electrodes, for roughing and finishing.

The method is used to manufacture small and medium high precision dies [8].

4 Selecting the Machining Process

Selecting the machining process for the dies is done using the Value Analysis method. There shall be selected the optimal machining technological process for a forging die, made of different semi-products and there shall be shown the implications of selecting each semi-product from the point of view of mechanical

Table 5 The classification of the functions

Symbol	Functions	Type of function*
F4	Provides machining	FS
F2	Provides material composition	FS
F1	Provides semi manufacturing production	FS
F3	Provides structural changes	FS
F6	Provides imposed parameters (hardness, wear resistance, shock resistance,…),	FC
F7	Resist environmental actions	FC
F9	Allows restoration	FC
F8	Allows control	FC
F10	Provides the user interface	FE
F5	Allows easy assembly, disassembly	FC

* *FS* service function; *FC* constraint functions; *FE* estimation function

processing and costs. The functional shape of the part is represented by: size, shape tolerances and the position of surfaces, size tolerances, surfaces quality, hardness and operating conditions.

5 Modeling Technological Process Using Value Analysis

In this article the author highlight and present:

1. the particular role of applying the Value Analysis approach to the selection of the die machining processes,
2. the working mode for optimizing the value/cost ratio,
3. a valid and useful guide for specialists to optimize the value/cost ratio of the die machining technological processes.

In this article the Value Analysis product/set is considered to be a die machining technological process. The components of this product/set are the stages/operations of the die machining technological process.

There shall be presented the iterations and conclusions drawn from applying the Value Analysis approach in the assumptions described above.

Table 5 shows the classification of the functions starting from the function analysis of the product—respectively—die machining process.

5.1 Iteration 1

Throughout the two iterations of the Value Analysis there shall be kept the 10 functions outlined in Table 5. Table 6 shows the value weighting of the functions.

Table 6 Value weighting of the functions (*coordinate X)

Functions	F4	F2	F1	F3	F6	…	F5	Total
No. of points	10	9	8	7	6		1	55
Ratio	0.18	0.16	0.14	0.12	0.10		0.01	1
Percentage*	18.2	16.4	14.5	12.7	10.9		1.82	100

Table 7 Cost weighting of the functions (**coordinate Y, cost $)

Parts	Cost of parts**	Functions						
		F4	F2	F1	F3	F6	…	F5
…	…							
Total cost	1,200	256	119	155	133.5	181.5		61.7
Ratio		0.21	0.09	0.12	0.11	0.151		0.05
Cost of functions of percentage		21.3	9.91	12.9	11.1	15.13		5.14

The author have developed a software that calculates all the values from the shown tables and draws all the diagrams necessary for presenting the findings, in all the iterations of the approach. The calculus is made using the least squares method.

Value weighting of the functions are the values in the last row of Table 6.

The allocation of costs to functions is made in economic dimensioning of the functions phase.

The allocation of costs to functions was performed in the matrix functions—costs from Table 7. In Table 7 the cost is distributed on the function/functions it is part of.

In the first iteration the machining process is as follows:

- alloying elements at minimum values,
- obtaining semi-product (cast + forged),
- primary heat treatment,
- roughing (milling, drilling, milling copying, electro erosion),
  - secondary heat treatment (improvement, Thermochemical nitro–ferrox treatment),
  - manual semi finishing,
  - manual finishing (grinding, …),
  - manual check (with template, ultrasonic),
  - obtaining test pieces,
  - repairs.

Cost weighting of the functions are the values in the last row of Table 7.

The check of this identity is performed using regression analysis by determining the linear function (the regression line) that represents the average proportionality.

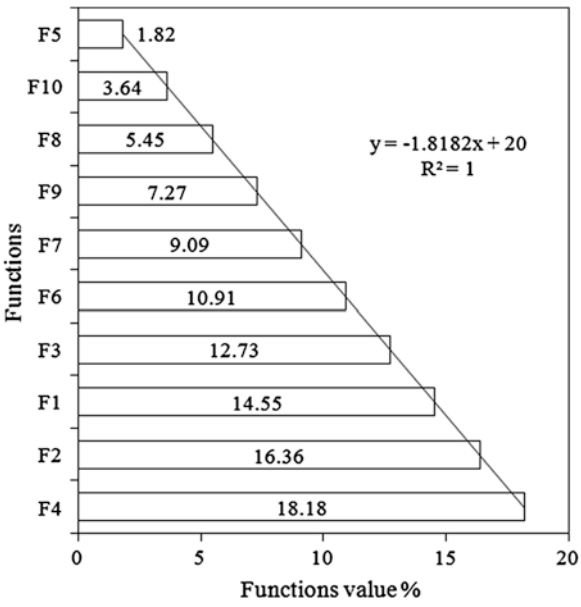


Fig. 1 Value weighting of the functions

The regression line passes through the origin, as it is considered that a function with "0" value costs "0".

The line has the shape:

$$y = a * x. \tag{1}$$

In the case of perfect proportionality all points are on the line (1). In order to simplify, the calculation is tabulated.

The coordinates x_i and y_i are given in Tables 6 and 7 and based on the data calculated in this tables the diagrams from Figs. 1, 2 and 3 are drawn:

- the value weighting of the functions (Fig. 1),
- the cost weighting of the functions (Fig. 2) and
- the cost and value weighting of the functions (Fig. 3).

The diagram in Fig. 1 shows the value ranking, prioritization and weighting of the functions. The assessment of the functions which is shown in Fig. 2 highlights the most expensive functions.

The diagrams allow comparisons between the total costs of the functions and, within the total costs, there are highlighted:

- the very expensive functions, with the highest weighting in the total cost of the product,
- the functions whose implementation requires disproportionate costs as compared with other functions.

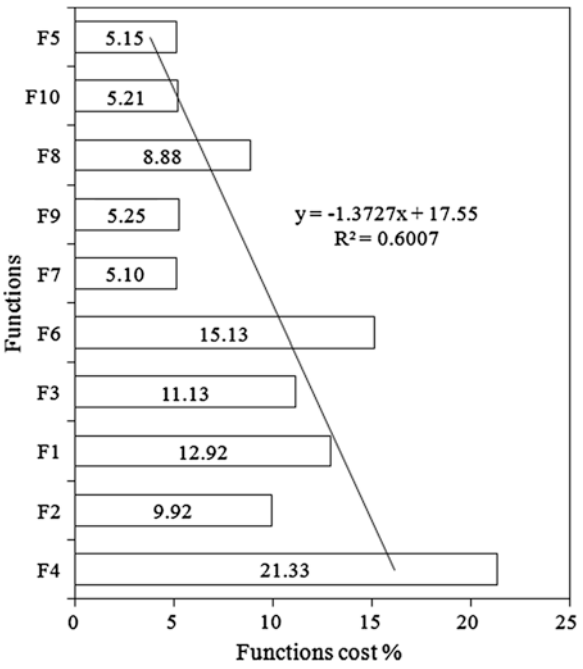


Fig. 2 Cost weighting of the functions

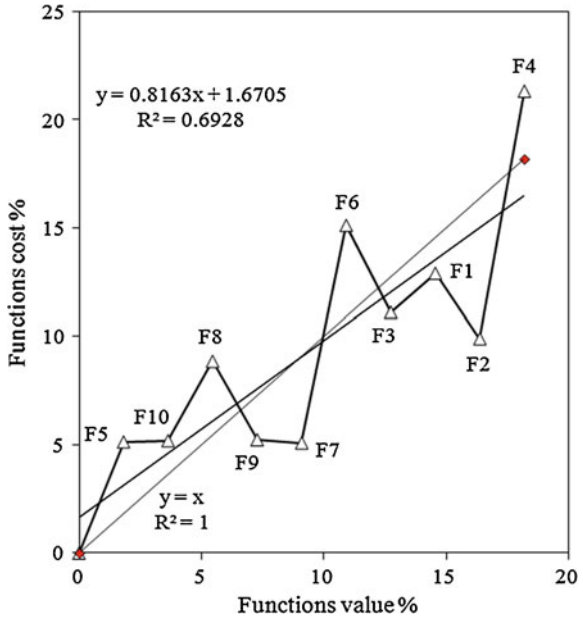


Fig. 3 Weighting of the functions in value and cost

The diagram shows a Pareto type distribution, i.e. 20–30 % of the total number of functions comprises 60–75 % of the total cost of functions. These functions are shown in the example from Fig. 2, functions F4, F1, F3 and F6.

Regression equation describing this distribution is $y = -1.3727 * x + 17.55$ with R, squared value on chart $R^2 = 0.6007$. If there is such a distribution, the first functions in the order of costs, representing 20–30 % from the total number of functions, the functions are considered expensive.

In diagram from Fig. 3 can be seen the regression line drawn using the method of the least squares and the comparison of functions in terms of value and costs:

- the equation line $y = x$ (the first bisector) the line that averages the weighting of functions in value and cost, expresses the ideal situation of the disparity between the two weightings, the weighting of functions in value and costs,
- the regression line of equation $y = 0.8163 * x + 1.6705$, which approximates the arrangement of the points, expresses the real situation of the disparity between the two weightings, the weighting of functions in value and costs,
- functions F4, F6, F8, F10 and F5 are situated above the lines aforementioned. The weighting of the cost is larger than the weighting of the value of these functions.

These functions are deficient and attention should be focused on them.

The cost of these functions should be reduced.

In order to reduce the disparity between the two weightings, the weighting of functions in value and costs the points should be aligned as perfectly as possible on the equation line $y = a * x$, the first bisector from Fig. 3.

The criterion of this reduction often leads to carrying out the Value Analysis studies in cascade, the optimization of the constructive solution being thus an iterative process. There are analysed first of all the functions situated above the ideal regression line (1) and these are made cheaper, the real regression line is drawn again and afterwards is found that other functions are above it; these functions are analysed looking for solutions to decrease their cost and the regression lines are drawn again, etc., the constructive solution being improved from one iteration to another.

In the second iteration of the Value Analysis approach there shall be considered the functions situated above the ideal regression line (1): F4, F6, F8, F10 and F5.

5.2 Iteration 2

For the second iteration there shall be presented only the results in tabulated form.

The weighting of the functions in value is the same as for the first iteration as no functions were added or removed from the system (Table 5).

As the functions that cost more are highlighted in Fig. 3, solutions shall be suggested for reducing the cost of these functions.

The cost of these functions can be reduced by answering the following questions:

- can there be used less expensive semi-products?
- can the thermal regimes of the heat treatments be reduced?
- can there be eliminated a heat treatment operation?
- can there be used another thermal, thermo-chemical operation?
- can the die be milled at a lower cost?

These questions must be answered in such manner so that the properties, characteristics and performance of the die's alloy are not affected, but improved if possible!

In the second iteration, actions were taken for the following cost elements, manufacturing process are the follows:

- alloying elements to the maximum values,
- obtaining semi manufactured (cast + forged),
- primary heat treatment in steps,
- roughing (milling 3D),
- secondary heat treatment (improvement),
- manual finishing (grinding, …),
- 3D check (ultrasonic),
- obtaining test pieces,
- repairs.

The allocation of costs to functions was performed in the matrix functions—costs from Table 8. In Table 8 the cost is distributed on the function/functions it is part of.

Cost weighting of the functions are the values in the last row of Table 8.

Coordinates x_i and y_i are given in Tables 6, 7 and, 8 based on the data calculated and presented in this tables, the diagrams from Figs. 4 and 5 are drawn:

- the value weighting of the functions (identical with Fig. 1—iteration 1),
- the cost weighting of the functions (Fig. 4),
- the cost and value weighting of the functions (Fig. 5).

The critical assessment of the functions presented in Fig. 4 highlights the most expensive functions.

Regression equation describing this distribution is $y = -1.481 * x + 18.146$ with R, squared value on chart $R^2 = 0.7741$.

The diagram in Fig. 5 represents the regression line drawn using the least squares method and presents the comparison of functions in terms of value and cost.

In the diagram from Fig. 5 there can be seen the following lines:

- the equation line $y = x$ (the first bisector), the line that averages the weighting of functions in value and cost, expresses the ideal situation of the disparity of the two weightings,

Table 8 Cost weighting of the functions (**coordinate Y, cost $)

Parts	Cost of parts**	Functions						
		F4	F2	F1	F3	F6	…	F5
…								
Total cost	985	172.8	134	156.5	113.5	125		41.75
Ratio		0.175	0.136	0.159	0.115	0.127		0.042
Cost of functions of percentage		17.54	13.6	15.89	11.52	12.69		4.239

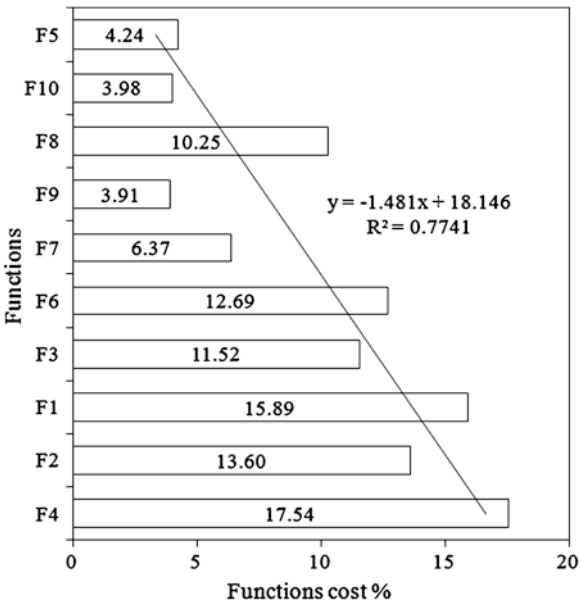


Fig. 4 Cost weighting of the functions

- the regression line, of equation y = 0.8609 * x + 1.2644, which approximates the arrangement of the points, expresses the real situation of the disparity of the two weightings,
- functions F1, F6, F8 and F5 are situated above the lines aforementioned. The weighting of the cost is larger than the weighting of the value of these functions.

These functions are deficient and attention should be focused on them. The cost of these functions should be reduced.

Following the second iteration there can be seen in Figs. 3 and 5 that the value of some functions increased, of other functions decreased, but the cost of those that increased eventually decreased due to the decrease of the cost of the "product".

The functions F1, F6, F8 and F5 are situated above the regression line.

There can be seen comparatively to Fig. 3 (iteration 1) in Fig. 5 (iteration 2) that the functions are grouped closer to the ideal regression lines.

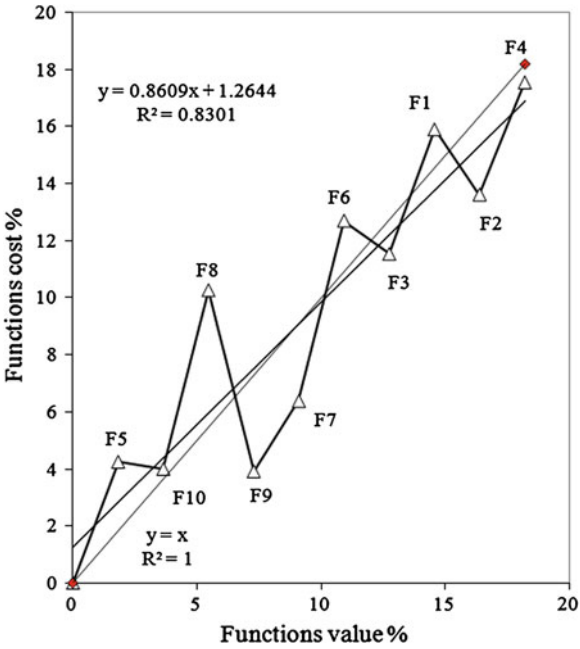


Fig. 5 Weighting of the functions in value and cost

Below are presented comparatively the equations of the regression lines (the real situation) and the correlation coefficients R^2 for the two iterations:

Iteration 1: $y = 0.8163 * x + 1.6705$, $R^2 = 0.6928$,
Iteration 2: $y = 0.8609 * x + 1.2644$, $R^2 = 0.8301$,

There can be seen an increase in the value of the correlation coefficient R^2 in the second iteration as compared to the first iteration, thus resulting that the dispersion of the points decreased in relation with the regression line.

The iterations continue until the correlation coefficient R^2 tends to value 1 and the regression line (the real situation) tends to $y = x$ (the ideal situation).

6 Results

In two iterations of the Value Analysis method, the die machining technology was redesigned and optimized from the following points of view:

1. engineering:
   - the die machining process was modified,
   - the percentages of the alloying elements were modified from a minimum, in the first iteration to a maximum, in the second iteration,

Table 9 Comparing costs in the two iterations of value analysis method

	F4 (%)	F6 (%)	F9 (%)	F5 (%)
First iteration	21.33	15.13	5.25	5.15
Second Iteration	17.54	12.69	3.91	4.24
The reduction of percentage	17.76	16.12	25.52	17.67

- the nitro–ferrox heat treatment was eliminated without changing in any way the properties required to such steel,
- the primary continuous heat treatment was replaced with a heat treatment in stages,
- the classic milling process (less expensive) of the die print was replaced with high velocity milling, using 3D processing machines (workmanship, more expensive equipment but higher productivity and precision),
- the manual control process, using templates (less expensive) was replaced with a 3D machines control process (workmanship, more expensive equipment but higher productivity and precision),
- as the machining processes are more precise in the 2-nd iteration, the final remedies are fewer than in the first iteration,

2. economics:

- the cost of the product decreased from 1,200$, in the first iteration to 985$ in the second iteration, a 25,41 % decrease,
- the cost of functions F4, F6, F9 and F5 decreases in the second iteration compared to the first iteration (Table 9).

In the third iteration of the Value Analysis method there shall be analyzed the functions situated above the regression line $y = x$ (F1—Provides semi manufacturing production, F6—Provides imposed parameters (hardness, wear resistance, shock resistance, …), F8—Allows control and F5—Allows easy assembly, disassembly), there shall be analyzed the components participating to achieving these functions and solutions shall be proposed for reducing the costs.

For functions F4 and F3 there shall be searched alloying and thermal treatments elements that decrease their cost, but maintain the qualities and the properties of the die material.

7 Applications and Conclusions

This guide can be used for optimizing the value/cost ratio for different types of machining technological processes for various parts:

Important is to achieve:

- functional modelling (with the help of functions) of the technological processes used to machine parts (considered as a set) within the Value Analysis approach,

- the valorisation of the functions,
- the allocation of the cost of technological stages/operations on the function/functions they are part of,
- the manner of interpreting the results from the diagrams that represent the weighting of the functions in value and cost,
- the proposal of variants with a lower cost for processing operations, heat treatment operations, control operations,
- the working manner using the programs made available by the author.

Designing a mathematical—economic model for making decisions regarding the optimization of the technological processes for machining parts in terms of the Value Analysis approach is an absolute novelty in the field.

This modelling has an important role because it opens a wide range of engineering applications.

The modelling of the Value Analysis products, with various applications that range from engineering, medicine,…, to services, has lead in the last 65 years of applications to undeniable progress. Along with this study the Value Analysis method takes an important step, opens new horizons for applications and keeps up with the directions of the science to penetrate in different fields of activity, highlights the weaknesses (increased costs) of the technological processes of machining different parts/elements and guides, knowingly, the engineer towards the deficient points and helps to remedy them.

References

1. http://www.value-eng.org/valueworld/older_issues/1968_April.pdf
2. www.scav-csva.org/v1/html/fr/Methode.html
3. michel.jean.free.fr/cours.AV.html
4. michel.jean.free.fr/AV/introduction.html
5. michel.jean.free.fr/AV/glossaire.html
6. www.valorex—constantineau.francine
7. NF EN 12973, NF X-50.150, NF X-50.151, NF X-50.152, NF X-50.153
8. Bejan, V., *The technology for manufacturing and repairing technological equipment*, vols. I, II, Bucharest (1991). ISBN 973-95458-7-4

Object-Oriented FSM-Based Approach to Process Modelling

Jakub Tůma, Vojtěch Merunka and Robert Pergl

Abstract We presents with this paper approach based on combination of the FSM and the Object-Oriented Approach, which is convergent. This convergent approach to modelling of business requirements and software development is main idea of this paper. The paper is divided into three parts, motivation and discussion is about needs connect two areas business requirements and software engineering, the idea of modelling of processes [3] and business situations as FSM and the third part is mapping of the proposed approach to BPMN-based and UML-based models. Mapping provides interesting new findings resulting from the proposed approach. This approach is based on our experience with our recent practical projects concerning business modelling and simulation in various application areas (e.g. health care, gas supply industry, regional management, administration process design of a new faculty of a university, administration process of building permission) and subsequent software development in these application areas.

Keywords FSM · OOP · BORM · UML · Mealy automata

J. Tůma (✉) · V. Merunka · R. Pergl
Department of Information Engineering, Czech University of Life Sciences Prague,
Kamýcká 129 165 21 Praha 6, Czech Republic
e-mail: jtuma@pef.czu.czpergl@pef.czu.cz

V. Merunka
e-mail: vmerunka@gmail.com

R. Pergl
e-mail: robert.pergl@fit.cvut.cz

V. Merunka
Department of Software Engineering, Czech Technical University in Prague,
Trojanova 13 120 00 Praha 2, Czech Republic

R. Pergl
Department of Software Engineering, Czech Technical University in Prague,
Thákurova 9 160 00 Praha 6, Czech Republic

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*,
Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_51,

1 Introduction

Software application development for business and similar domain-specific areas shifts the attention at the requirement analysis and design activities, e.g. from the programming level to the modeling level. Model-Driven Architecture (MDA) [11] is the recent approach based on strategy of the application development based on requirements, conceptual and design modeling. The typical tool used in this area is the UML—Unified Modeling Language [20].

Our idea of continuous model-driven engineering aims to fill in the gap between of two worlds of "Business", which is process-based and requires deep management and economical knowledge, and "IT", which uses its own modern software development tools and techniques. This is to minimize the failure rate of information systems through the application of proper simulation and modeling techniques before the system is built. We want to advance the discipline of conceptual modeling in the area between the use of business domain knowledge and the use of modern advanced programming techniques and tools such as object-oriented programming environments (.NET, XCode, Visual-Works,…), prototyping environments (Self, Squeak), non-traditional programming languages (Smalltalk, Objective-C) etc.

The goal of our paper is to converge the BPMN and UML modeling using approach, which will enable to use only one modeling and simulation paradigm trough the entire software system development life-cycle. Our paper contributes to the area of system modeling methodologies, tools and techniques to enable simulation, verification and validation activities.

2 Motivation

2.1 Our Experience

In our experience, any modeling and simulation tool and diagramming technique used at this kind of business projects should be comprehensible to the stake-holders, many of whom are not software engineering literate. Moreover, these diagrams must not deform or inadequately simplify requirement information. It is our experience that the correct mapping of the problem into the model and subsequent visualization and possible simulation is very hard task with standard diagramming techniques. We believe that the business community needs a simple yet expressive tool for process modeling; able to play an equivalent role to that played by Entity-Relation Diagrams, Data-Flows Diagrams or Flow-Charts over the past decades. One of the strengths of these diagrams was that they contained only a limited set of concepts (about 5) and were comprehensible by problem domain experts after few minutes of study. Unfortunately UML approach (as well as BPMN) lost this power.

That is why we developed and successfuly used our own BORM process diagraming technique [10] and our own way to start business system analysis.

The initial work on Business-Object Relation Modeling (BORM) was carried out in 1993 under the support of the Czech Academic Link Programme (CZALP) of the British Council, as part of the Visual Application Programming Paradigms for Intergated ENvironmentS (VAPPIENS) research project; further devel-opment and recent practical projects in the last decade has been carried out with the support of Craft.CASE Ltd.—the British software consulting company supporting innovative technologies. (VAPPIENS was funded by the British Governments CZALP, administered by the British Council. The authors acknowledge the support they received from this source, which enabled them to meet and carry out the initial work, out of which BORM grew.) BORM has been used in last 15 years for a number of business consulting and software engineering projects including

- The identification of business processes in metropolitan hospital,
- The modelling of properties necessary for the general agricultural commodities wholesale sector requested by the Agrarian Chamber,
- As a tool for business process reengineering in the electricity supply and gas supply industry,
- As a tool for business process reengineering for telecommunication network management,
- In organizational modelling and simulation of regional management project concerning the analysis of the legislation and local officials' knowledge such as living situations, law, urban planning etc.,
- Several business process simulation projects in area of simulation of marketing chains for Makro, and
- Visualization of safety and fire regulations in the electric power engineering sector.

However during the last decade there was an significant upgrade of UML, and also a new standard for business process modeling BPMN has been developed and we recognized that our approach is close to both of them. We think that based on our past experience, we can propose a new approach. This new approach is based on the object-oriented concepts, is using the well proven technology of FSM for modeling and simulation. It has almost the same expressive power as the UML and the BPMN together, but it is expressed in a uniform and simpler manner.

2.2 Gap Between Business and Software Modeling

Nowadays, there is a great variety of tools and techniques for business modeling. Unfortunately, there is no standard for business modeling like UML [6] for software engineering yet. Nevertheless, we can presume that the approach to become a standard will be the BPMN. [2, 7, 16] in Europe, Aris and its EPC diagram is still very popular; however BPMN authors say this technique becomes obsolete and is almost unknown outside the Europe.

When OOP started to be used in practice, it was assumed that object-oriented technology will become mainstream in software development (which was correct)

but also that object-oriented approach will affect the approach to business process modeling, organizational engineering and the whole area of activities preceding the formulation of information system requirement. That, unfortunately, did not happen. These pre-implementation activities are still car-ried out the old fashioned way, which results in a semantic gap between the world of business modeling and the world of software modeling.

Latest publications even express opinions such as "OOP has failed, OOP is a dead end, etc.". These are written by people who have no experience with pure object-oriented languages and environments, but only with hybrid ones (e.g. Delphi, Java, etc.) and they generalize their negative practical experience with failed projects to the whole paradigm.

Expected output of the business engineering activities is information or data in a form that can be directly used as an input for implementation of the system in the spirit of software engineering. However, this is not the easy case; there are following issues described by Illgen and Hulin in [9] and Van der Aalst in [1]:

1. *Oversimplification*—while trying to at least finish business and organizational model we are forced simplify the problem being modeled and
2. *Inability*—some important details cannot be recorded because of the method being used.

We believe together with Schach [17] that this is the reason why software system modeling and subsequent design of business applications is a pretty hard deal: Today's world of software development still works with algorithmic and imperative style of thinking, and therefore the produced models concentrates on functions, function separation, continuity etc. But this "behavioral" and "straight timeline" approach goes against the business modeling paradigm which mainly focuses on states, situations, rules—often even in a form of statutory regulations and directives and concurrency. Behaviors are of less importance and make sense only if we know what they are good for. That is why it is much more natural to start analyzing a business process from the description of initial situation (e.g. we have goods, we want to sell it, the customer wants our goods and has money,…) and the description of the desired final situation (e.g., the paying customer makes us a good profit). In this perspective, the sequence of behaviors is only a result of an effort to get from the initial to the final situation, thus, being much more declarative then imperative style of modeling. The significance of the declarative approach is expected to grow even in the software programming itself, where the imperative (e.g. behavior-oriented) modeling style is still prevalent, although unnatural for business and organizational modeling.

3 Proposed Solution—FSM and OOP Combination

One possible solution is based on the reuse of old thoughts from the beginning of 1990s regarding the description of object properties and behavior using finite state machines (FSM). The first work expressing the possible merge of Object-Oriented

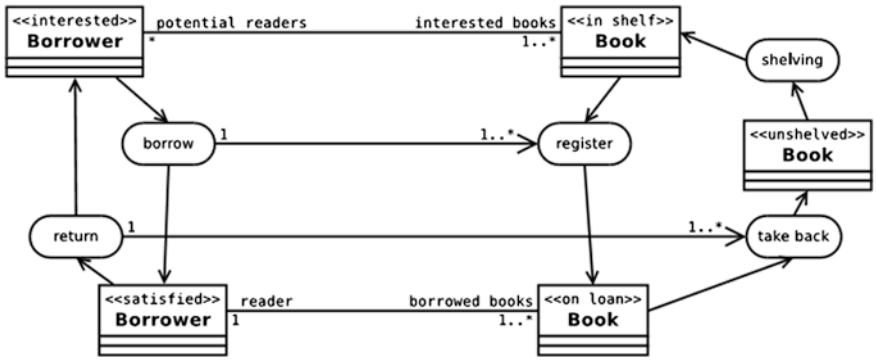


Fig. 1 Object-oriented FSM-based model of a library process

Paradigm (OOP) and FSM was the Shaler's and Mellor's book [15]. One of the best books speaking about the applicability of OOP to the business modeling was written by Taylor [19]. These works together with our practical experience [13] is why we believe that the business requirement modeling and simulation and software modeling could be unified on the platform of OOP and FSM.

Definition 1
A graph G is an ordered pair (V(G), E(G)) consisting of a set V(G) of vertices and a set E(G), disjoint from V(G), of edges [4].

Definition 2
(*Mealy automata*) Let A and B be arbitrary sets. A Mealy automaton (S, φ) with inputs in A and outputs in B consists of a set of states S and a transition function $\varphi : S \rightarrow (B \times S)^A$. This function maps a state $s_0 \in S$ to a function $\varphi(s_0) : A \rightarrow (B \times S)$, which produces for every input $a \in A$ a unique pair (b, s_1), consisting of the output b and the next state s_1 [21].

Figure 1 shows and example of a model of a book in a library represented in a form of a finite state machine with three states: a book on a shelf, a book on loan and a returned book to be put back on a shelf. These states are easily recognizable through an interview with domain experts.

4 Discussion

The presented modeling approach unifies UML-style object modeling and business process modeling in the business and organizational engineering techniques and tools style. Models like UML, BPMN and other can be easily derived from this model.

From the viewpoint of this schema, we can consolidate the terms used in modeling with BPMN and UML into a Table 1.

Table 1 Modeling concepts coverage

Our approach	BPMN-based approach	UML-based approach
Objects	Swimlines only (process participants)	Yes
Objects states	Can be expressed by various events	Yes
Associations between (data links)	No	Yes
Associations between object states (data links)	Yes	No
Object behaviors (activities)	Yes	Yes
Communications between behaviors (messages)	Yes	Yes
Generalization-specialization relationship (inheritance)	No	Objects only
Whole-part relationship (composition)	No	Objects only

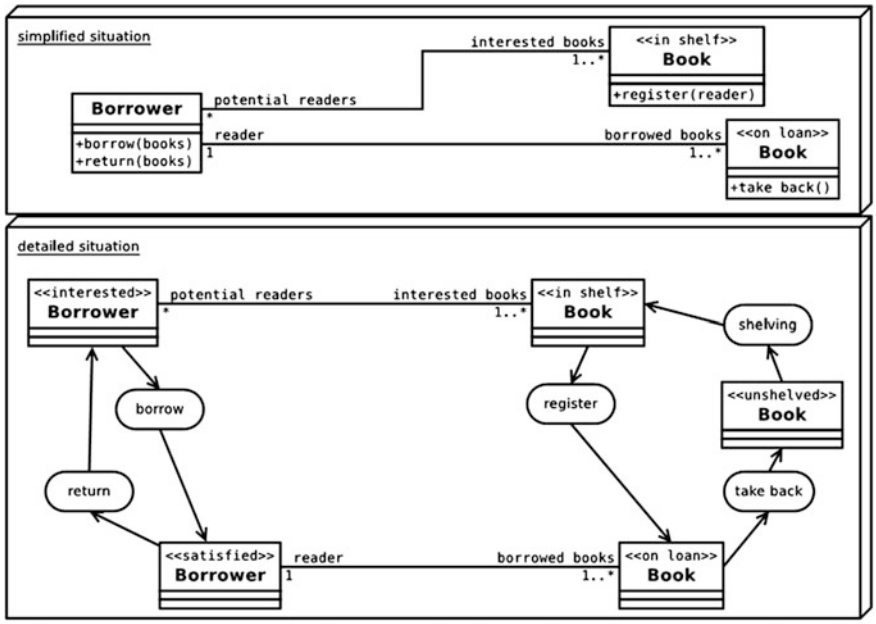


Fig. 2 Simplified model transformed to the ordinary class-diagram

The proposed unified approach generalizes object modeling. In this perspective, modeling in UML and BPMN could be perceived as mutually following stages of the new more universal approach according to the MDA principles [11]. Figure 3 shows mapping or our model to the standard BPMN and Fig. 2 shows simplified model transformed to the ordinary class-diagram.

Figure 2 shows class model. Simplified situation shows class model with methods and detailed situation shows classes with their states. For example class

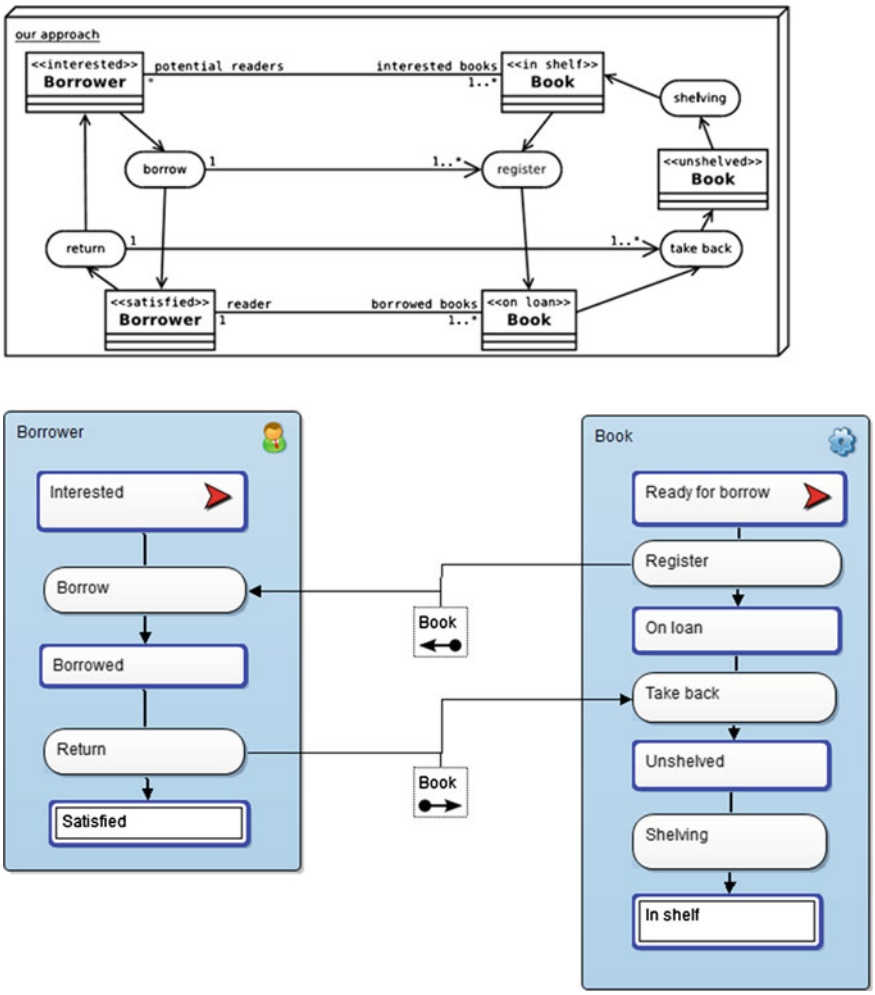


Fig. 3 Our approach and BORM method

Borrower after borrowing book became Borrower reader. Simplified situation is synthesis (generalization) of detailed situation on Fig. 2.

Possible advantages of our convergency approach follows:

1. BPMN and UML both cover only a subset of the entire exploitable space of modeling concepts (see Table 1).
2. The most important concepts are states of objects. Behaviors represent only the necessary glue between them. Both business processes and software components should be therefore modeled by starting with their states—situations of participating objects in the requested structure in some time. Modeling can be easier, more precise and less behavioral-imperative then it is today.

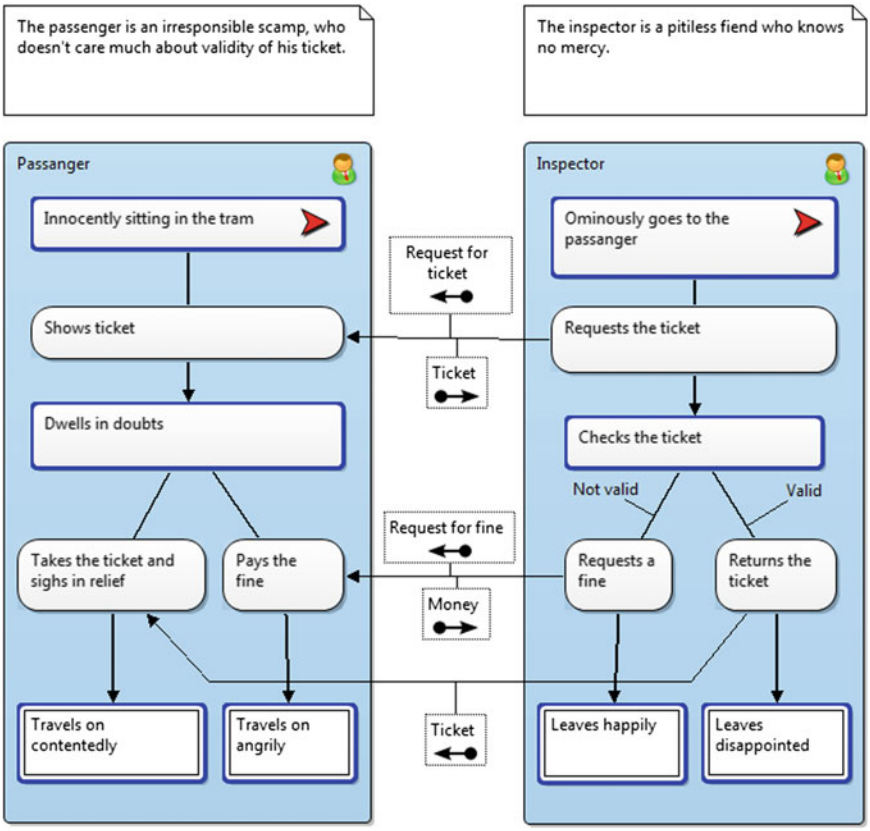


Fig. 4 Process of tickets checking

5 Conclusion

In this paper we presented the idea of the convergent approach to modeling of business requirements and software development. Our approach combining the object-oriented approach and finite-state machines is based on our practical experience with recent BORM project, which were aimed to help the teams made by business consultants and software developers from various areas (e.g. health care, gas supply industry, regional management). We feel that the highest value of our approach is generated by the way of modeling, which smoothly connects two different worlds: business engineering and software engineering. We believe that this approach can help in future possible integration of BPMN and UML models for complex projects requiring the strong collaboration between software system architects and problem domain experts in area of organization structures modeling and subsequent simulation as it is predicted by Scheldbauer in [16].

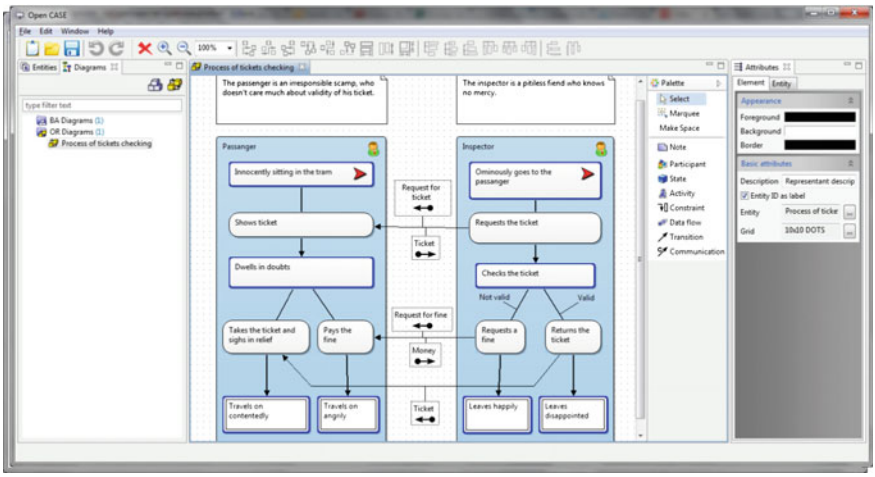


Fig. 5 OpenCASE [14]

Our future work will be focused on implementation of the proposed concepts in selected CASE tools and on incorporation of this approach into the BORM methodology [10]. The hot candidate for this project is the MetaEdit CASE tool by the Finnish company Metacase Ltd. [12]. Recently, the project on object-oriented CASE tool supporting this approach sponsored by a consortium of software companies has been started concrete process diagram on Fig. 4 and CASE you see on Fig. 5 [14]. Our future research will focus on describing the rules of our object-oriented normal forms as a sequence of refactoring steps.

Acknowledgments The authors would like to acknowledge the support of the grant ČZU v Praze, IGA project number 20121059, Metody automatizovaných Transformací modelů v informačních systémech. Also would like to thanks to grants support SGS11/166/OHK4/3T/14 and NAKI MK-S-3421/2011 OVV.

References

1. Aalst van Der W.: Business process simulation revisited, keynote speech at the EOMAS workshop 2010, [cit: 2011-04-10 http://www.eomas.org] (2010)
2. Allweyer T.: BPMN 2.0, Books on Demand GmbH, Norderstedt (2010). ISBN: 978-3-8391-4985-0
3. Barjis J.: Developing executable models of business systems. In: Proceedings of the ICEIS—International Conference on Enterprise Information Systems, pp. 5–13. INSTICC Press (2007)
4. Bondy, J. A., Murty, U.S.R.: Graph Theory. Springer, Berlin (2008). ISSN: 0072-5285, ISBN: 978-1-84628-969-9
5. Degen W., Heller B., Herre H., Smith B.: GOL—towards an axiomatized upper level ontology. In: Proceedings of FOIS'01, Ogunquit, Maine, USA, ACM Press (2001)

6. Eriksson, H., Penker, M.: Business modeling with UML. Wiley, New York (2000). ISBN: 0-471-29551-5
7. Grosskopf A., Decker G., Weske M.: Business process modeling using BPMN. Meghan Kiffer Press, New York (2006). ISBN: 978-0-929652-26-9
8. Hohenstein U.: Bridging the gap between C ++ and relational databases In: Proceedings of ECOOP 1996, Springer Lecture Notes in Computer Science, vol. 1098/1996, pp. 398–420 (1996)
9. Ilgen D., Hulin C. L.: Computational modeling of behavior in organizations—the third scientific discipline. American Psychological Association, Washington DC. ISBN 1-55798-639-8 (2000)
10. Knott, R. P., Merunka, V., Polak, J.: The BORM methodology: a third-generation fully object-oriented methodology. In: Knowledge-Based Systems Elsevier Science International New York (2003). ISSN: 0950-7051
11. MDA—The Model Driven Architecture, OMG—The Object Management Group, http://www.omg.org
12. MetaCase—domain-specific modeling with MetaEdit + , http://www.metacase.com
13. Molhanec, M.: Conceptual normalisation formalised. In: Enterprise and Organizational Modeling and Simulation, pp. 159–172. Springer, Berlin (2011). ISBN: 978-3-642-24174-1
14. Pergl, R., Tůma, J.: OpenCASE—a tool for ontology-centred conceptual modelling, pp. 511–518. Springer, Berlin, 2012-01-01. ISBN: 978-3-642-31068-3
15. Shlaer, S. Mellor, S.: Object Lifecycles: modeling the world in states. Yourdon Press, Upper Saddle River (1992). ISBN: 0136299407
16. Scheldbauer M.: the art of business process modeling—the business analyst guide to process modeling with UML and BPMN, Cartris Group, Sudbury MA (2010). ISBN 1-450-54166-6
17. Schach S.: Object-Oriented software engineering. McGraw Hill, Singapore (2008). ISBN 978-007-125941-5
18. Silver B.: BPMN method and style. Cody-Cassidy Press, Aptos CA (2009). ISBN: 978-0-9823681-0-7
19. Taylor, D. A.: Business engineering with object technology. John Wiley, New York (1995). ISBN: 0-471-04521-7
20. The UML standard, OMG—The Object Management Group, http://www.omg.org, ISO/IEC 19501
21. West B. D.: Introduction to graph theory, Pearson Education, Upper Saddle River (2002). ISBN 81-7808-830-4

Performance Analysis of Built-in Parallel Reduction's Implementation in OpenMP C/C++ Language Extension

Michal Bližňák, Tomáš Dulík and Roman Jašek

Abstract Parallel reduction algorithms are frequent in high performance computing areas, thus, modern parallel programming toolkits and languages often offer support for these algorithms. This article discusses important implementation aspects of built-in support for parallel reduction found in well-known OpenMP C/C++ language extension. It shows that the implementation in widely used GCC compiler is not efficient and suggests usage of custom reduction implementation improving the computational performance.

Keywords C/C++ · GCC · OpenMP · Reduction · Performance · Analysis · Improvement

1 Introduction

A parallel reduction can be implemented on SMP computers [1] by using OpenMP [4, 5] in many ways. In addition to the built-in support for the reduction operations a programmer can implement it himself by using standard work-sharing constructs provided by the OpenMP or by using "manual" distribution of the work among the available CPUs.

M. Bližňák (✉) · T. Dulík · R. Jašek
Faculty of Applied Informatics, Department of Informatics and Artificial Intelligence, Tomas Bata University in Zlin, Nad Stráněmi 4511 760 05 Zlín, Czech Republic
e-mail: bliznak@fai.utb.cz
URL: http://www.utb.cz/fai

T. Dulík
e-mail: dulik@fai.utb.cz

R. Jašek
e-mail: jasek@fai.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_52,

The article shows that the built-in specialized reduction clause [5] is easy to use but it is neither time- nor cost-optimal, especially for small sets of operands reduced by an operator with high time complexity.[1] It also shows how to implement time- and cost-optimal parallel reduction algorithm suitable for any problem size.

It is supposed the reader has knowledge of ANSI C/C++ programming language and at least basic notion of OpenMP library.

2 Parallel Reduction on PRAM

Consider EREW PRAM [6, 7] parallel reduction algorithm with p processors $P_0, P_1, \ldots, P_{n-1}$ and with n shared-memory cells $M[0], M[1], \ldots, M[n-1]$ described in Algorithm 2. It is obvious that $p = n$. In contrast to the sequential version of the reduction algorithm with time complexity $\Theta(n)$, the optimal parallelized algorithm can compute the result with time complexity

$$T_{(n,p)} = O(\log_2(p)) \tag{1}$$

Algorithm 1 Trivial parallel reduction

```
for j =1 ,..., ⌈log2 n⌉ do sequentially
  for all i =0 to n − 1 step 2^j do in parallel
    P_i : M[i]= M[i] ⊕ M[i +2^(j−1)]
  end for
end for
result ← M[0]
```

As can be seen from the Algorithm 1 the computation consist of $\log_2 n$ sequential phases $j = 1, \ldots, \lceil \log_2 n \rceil$ where just $\frac{n}{2^j}$ processors do useful work in each phase as shown in Fig. 1.

Now let us evaluate the *cost* [7] of the parallel reduction algorithm. Generally, the cost of a parallel algorithm solving a problem of size n on p processors denoted by $C_{(n,p)}$ is defined as

$$C_{(n,p)} = p \times T_{(n,p)} \tag{2}$$

Assume that a sequential *upper bound* [7] denoted by $SU(n)$ on number of sequential steps to solve a problem is known. A parallel algorithm solving the same problem is said to be *cost-optimal* if

$$C_{(n,p)} = O(SU(n)) \tag{3}$$

[1] For example various image processing algorithms which operate with number of image pixels per single reduction operation.

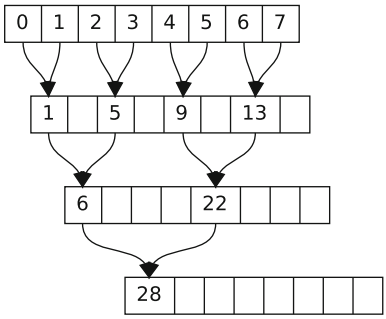


Fig. 1 EREW PRAM parallel reduction

The upper bound of the sequential reduction algorithm is $SU(n) = \Theta(n) = O(n)$ so the cost of the parallel reduction algorithm defined in Algorithm 2 is $C_{(n,p)} = p \times O(\log_2 p)$. For non-scaled parallel reduction algorithm $p = n$ which implies $C_{(n,p)} = p \times O(\log_2 p) = \Omega(n)$, hence, the algorithm is not cost-optimal since the condition defined in (3) is not met. The problem of the implementation is a lack of useful work distributed across the parallel system.

One of the possible ways how to improve the cost of the parallel algorithm is to set better *granularity*, i.e. to change a ratio between size of solved problem n and the number of processors p used for the calculation by using so called *scaling* of the algorithm [7].

Typically, decreasing number of used processors leads to better cost and efficiency of a parallel algorithm as stated in [1]. Therefore, we should modify the Algorithm 1 so the amount of the work for each processor increases. The possible modification is shown in Algorithm 2.

Assume $p' < p$ processors and the size of a problem $n > p'$.

Algorithm 2 Scaled parallel reduction

for all $i = 0$ **to** p' **do in parallel**
  $P_i : subresult \leftarrow M[i \times \frac{n}{p'}]$
  for $j = i \times \frac{n}{p'} + 1$ **to** $\frac{n}{p'}(i+1)$ **do sequentially**
    $subresult = subresult \oplus M[j]$
  end for
  $P_i : M[i] \leftarrow subresult$
end for
Calculate parallel reduction for p' items of M by using Algorithm 1
$result \leftarrow M[0]$

Parallel reduction described in Algorithm 2 and illustrated in Fig. 2 is calculated as follows: Input sequence of reduced items is split into p' sets of size $\frac{n}{p'}$ where each processor calculates the partial reduction sequentially. It is obvious that this stage produces p' partial results which are then combined into final result by using Algorithm 1. Time complexity of Algorithm 2 is

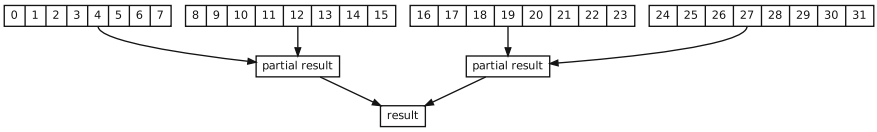


Fig. 2 Efficient scaled parallel reduction

$$T_{(n,p)} = O(\frac{n}{p'} + \log_2(p')) \tag{4}$$

Cost of the scaled parallel reduction algorithm is then

$$C_{(n,p)} = p'(\frac{n}{p'} + \log_2(p')) = n + p'\log_2(p') \tag{5}$$

In case the $n \gg p'$ the Eq. (5) can be rewritten into

$$C_{(n,p)} = O(n) = O(SU(n)) \tag{6}$$

so the algorithm can be regarded as both time- and cost-optimal.

Scaled parallel reduction of 32 items by using 4 processors is illustrated in Fig. 2.

3 Built-in Parallel Reduction Support in OpenMP

The OpenMP Application Program Interface (API) offers cross-platform shared-memory parallel programming framework for C/C++ and Fortran on all architectures, including Unix platforms and Windows NT platforms. Jointly defined by a group of major computer hardware and software vendors, OpenMP is a portable, scalable model that gives shared-memory parallel programmers a simple and flexible interface for developing parallel applications for platforms ranging from the desktop to the supercomputer [4].

The OpenMP defines set of compiler's preprocessor *directives* for parallel region definition, work-sharing constructs and per-thread synchronization. Behavior of the directives can be further tunned by so called *clauses*. In addition to the generic directives and their clauses the OpenMP contains also built-in support for parallel reduction provided by reduction clause of for directive. Let us to examine how the parallel reduction can be implemented by using these OpenMP constructs.

Listing 1 shows native OpenMP parallel reduction implementation which uses well-known *reduction* clause to calculate parallelized summation on shared variable without need of explicit inter-thread synchronization. In addition, this clause also avoids possible data race on the shared variable.

Listing 1: Built-in OpenMP parallel reduction

```
sum = 0;

#pragma omp parallel num_threads( p ) shared( sum, n, M )
{
  // calculation is done in T(n,p) = O(n/p + p)

  #pragma omp for schedule( static ) reduction( +:sum )
  for(unsigned long i = 0; i < n; ++i ) sum += M[i];
}

// implicit barrier at the end of the parallel region
```

The most of the OpenMP programmers will probably use this approach but is it really the best possible solution? As stated in GNU OpenMP Implementation notes [3], the reduction clause is implemented so it uses an array of the type of the variable, indexed by the thread's team id. The thread stores its final value into the array, and after the barrier, the master thread iterates over the array to collect the values. For better understanding what happens inside native OpenMP implementation the code presented in Listing ?? can be rewritten into the following implementation by using atomic operations and private variables as shown in Listing 2.

Listing 2: Built-in OpenMP parallel reduction deconvolution

```
sum = subsum = 0;

#pragma omp parallel num_threads( p ) shared( sum, n, M )
firstprivate( subsum )
{
  // phase I, calculated in T(n,p) = O(n/p)

  #pragma omp for schedule( static ) nowait
  for( unsigned long i = 0; i < n; ++i ) subsum += M[i];

  // phase II, calculated in T(n,p) = O(p)

  #pragma omp atomic
  sum += subsum;
}

// implicit barrier at the end of the parallel section
```

Now it is obvious, that the native implementation calculated in $T(n,p) = O(\frac{n}{p} + p)$ has got higher time complexity in contrast to the efficient implementation

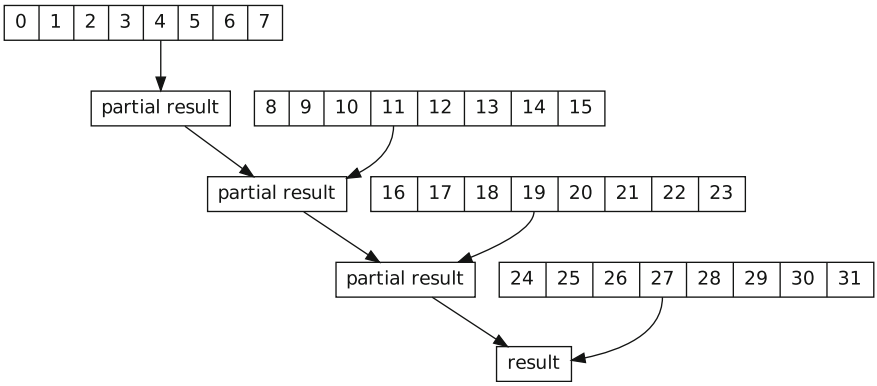


Fig. 3 Inefficient scaled parallel reduction

described in Algorithm 2 with time complexity defined in (4). The Fig. 3 illustrates how reduced operands are handled in this implementation.

Moreover, the reduction clause allows users to use just limited set of available reduction operators like $+$, $-$, *, / and $\%$ so if some custom reduction operator is needed then the user must implement the algorithm himself.

The following chapters focus on various possible implementations of optimal parallel reduction algorithm in OpenMP without usage of the reduction clause.

4 Custom Implementations

Various custom implementations of two main types of reduction algorithms are discussed in the following chapters: trivial and scaled parallel reduction algorithms. For simplicity, let us assume the number of reduced operands n is always factor of two.

4.1 Trivial Parallel Reduction Algorithm

Let us denote n to be the number of reduced operands and p to be the number of available CPUs performing the calculation where $n = p$. Also assume $M[]$ to be an array of size n containing all reduced operands. Then the trivial parallel reduction Algorithm 1 can be implemented as shown in Listing 3.

Listing 3: Custom trivial parallel reduction

```
#pragma omp parallel num_threads( p ) shared( M, p )
{
  int offset = 1;

  // calculation is done in T(n,p) = O(log(p)) with
  // BIG parallel overhead, assume n = p

  while( ( offset *= 2 ) <= p ) {
    #pragma omp for schedule(static)
    for( int pid = 0; pid < p; pid += offset )
      M[pid] += M[pid + (offset / 2)];

    // implicit barrier provided by parallel 'for' directive
  }
}

// the result is stored in M[0]

sum = M[0];
```

The algorithm consists of $\log_2(n)$ sequential phases where each phase calculates summation of adjacent operands/partial sums as described in Fig. 1. Notice that OpenMP parallel loop directive is used for distribution of the work among the available CPUs there. The work-sharing is performed in each sequential step which leads to significant parallel overhead in this implementation that cannot be neglect. Thanks to this hidden time constant the overall performance of this algorithm is comparable to the one which uses built-in reduction clause even when the theoretical time complexity is better. The comparison can be seen from benchmark results discussed in chapter "An Articial Bee Colony Algorithm for the Set Covering Problem" of this document.

The parallel overhead revealed in previous paragraph can be reduced by omitting the OpenMP parallel loop directive in favour of the work-sharing implemented by using standard C/C++ constructs. Let us discuss modifications of the previous algorithm shown in Listing 4.

Listing 4: Custom optimized trivial parallel reduction

```
#pragma omp parallel num_threads( p ) shared( M, p )
{
  int offset = 1;
  int pid = omp_get_thread_num();

  // calculation is done in T(n,p) = O(log(p)) with
  // SMALL parallel overhead, assume n = p
```

```
  while( ( offset *= 2 ) <= p ) {
    if( pid % offset == 0 ) {
      M[pid] += M[pid + (offset / 2)];
    }
    #pragma omp barrier
  }
}

// the result is stored in M[0]

sum = M[0];
```

In this implementation, all available CPUs remain active in all sequential steps but only subset of them do useful work. Indexes of enabled CPUs[2] are calculated in real-time by using modulo operator in contrast to the static work-sharing used in the Listing 3. Notice that each thread in the parallel team evaluates the condition and performs the summation independently to the other active threads so the global barrier placed below the conditional statement is needed for synchronization of subsequent parallel phases.

Benefits of this modification are clearly noticeable from benchmarks shown in chapter "An Articial Bee Colony Algorithm for the Set Covering Problem".

4.2 Scaled Parallel Reduction Algorithm

Both algorithms listed in chapter "A New Approach to Solve the Software Project Scheduling Problem Based on Max-Min Ant System" are implementations of inefficient trivial parallel reduction discussed in chapter "PPSA: A Tool for Suboptimal Control of Time Delay Systems—Revision and Open Tasks" assuming that the size of solved problem is equal to the number of assigned CPUs. Now, let us focus to time- and cost-efficient scaled parallel reduction implementation which allow users to calculate summation of $n \gg p$ operands with nearly linear speed-up.

Trivial scaled parallel reduction discussed in this chapters enhances the algorithm listed in chapter "A New Approach to Solve the Software Project Scheduling Problem Based on Max-Min Ant System" so it can be used for summation of huge number of operands with limited resources, i.e. available CPUs.

The algorithm consists of two main phases:

The first phase divides set of reduced operands into p subsets consisting of $\frac{n}{p}$ items. Each subset is reduced on single CPU in $T(n,1) = O(\frac{n}{p})$ and all subsets are processed on p CPUs in parallel so overall time complexity of this phase is $T(n,p) = O(\frac{n}{p})$.

The second phase calculates final result by using optimized parallel algorithm discussed in chapter "A New Approach to Solve the Software Project Scheduling

[2] CPUs which do some useful work.

Problem Based on Max-Min Ant System" in $T(n,p) = O(\log(p))$ from partial results provided by the first phase. Notice that private variables *subsum* are used by all threads in the parallel team for calculation of partial results instead of direct access to shared array $S[]$ used in the second phase. This modification avoids parallel overhead needed for inter-thread synchronization during concurrent write access to the shared resources. Moreover, *nowait* clause of parallel *for* directive omits unnecessary implicit barrier at the end of the parallel loop.

Global explicit barrier placed between the phases ensures that content of shared array $S[]$ used as an input in the second phase will be up-to-date after the finish of the first phase.

Overall time complexity of this implementation shown in Listing 5 is $T(n,p) = O(\frac{n}{p} + \log(p))$. As can be seen from the results published in chapter "An Articial Bee Colony Algorithm for the Set Covering Problem", this implementation guarantee best possible performance of all presented algorithms.

Listing 5: Custom scaled and optimized trivial parallel reduction

```
subsum = 0;

#pragma omp parallel num_threads( p ) shared( n, M, S )
firstprivate( subsum )
{
  int index = 1;
  int pid = omp_get_thread_num();

  // phase I, calculated in T(n,p) = O(n/p)

  // use private variable for calculation of partial sumation to avoid
  // shared memory contexts synchronization overhead

  #pragma omp for schedule( static ) nowait
  for( unsigned long i = 0; i < n; ++i ) subsum += M[i];

  S[pid] = subsum;

  #pragma omp barrier

  // phase II, calculated in T(n,p) = O(log(p))

  while( ( index *= 2 ) <= p ) {
    if( pid % index == 0 ) {
      S[pid] += S[pid + (index / 2)];
    }
    #pragma omp barrier
  }
}

// the result calculated in T(n,p) = O(n/p + log(p)) stored in S[0]

sum = S[0];
```

Table 1 Hardware configuration for benchmarking

System	SuperMicro A+ 1042G-TF Server
CPU	4 × 8-core AMD Opteron ™ 6128, 2 GHz, 32 cores in total
Chipset	AMD SR5690/SP5100
Memory	32 GB ECC, 1,333 MHz
OS	Debian Linux 6.0.5, KVM hypervisor

Table 2 Performance metrics for discussed reduction algorithms

	Sequential red.	Built-in red.	Custom red.	Optimized cust. red.
Calculation time (s)	2.6424051	0.2233472	0.2138646	0.1764686
Speed-up	–	11.8309300497	12.3555048381	14.9737976048
Efficiency	–	0.7394331281	0.7722190524	0.935862350

5 Performance Benchmarks

All benchmarks published in this chapter were measured on SuperMicro SMP server A+ 1042G-TF with following hardware configuration (Table 1):

Testing machine was using Ubuntu Linux 12.04 LTS installed inside a virtual machine managed by KVM hypervisor with 16 physical cores assigned and 16 GB RAM allocated.

5.1 Parallel Reductions Benchmark

Set of performance benchmarks was performed to evaluate a qualitative metrics of discussed scaled parallel reduction algorithms and their implementations. The testing application ran on virtual machine specified in the chapter "An Articial Bee Colony Algorithm for the Set Covering Problem". All available physical cores were assigned to OpenMP parallel team. Note that all standard compiler optimizations were disabled by using −O0 GCC compiler switch set during the building to omit any unwanted underlying source code optimizations influencing the measurement.

Results obtained from the tests are shown in Table 2. Parallel (calculation) time as well as the parallel speed-up and efficiency [1] of discusses parallel algorithms can be found there. The column named "*Built-in red.*" shows performance of scaled parallel reduction implemented by using built-in reduction clause described in Listing 1. The column named "*Custom red.*" shows performance of scaled algorithm described in Listing 3 while the last column named *"Optimized custom red."* shows performance of algorithm described in Listing 5.

As can be seen from the table the best results in meaning of the highest speed-up and efficiency and the lowest parallel time provides optimized custom algorithm listed in Listing 5. The speed-up and efficiency degradation visible from the results implies from non-negligible OpenMP implementation overhead. For more clearance the parallel times shown in the table are compared in Fig. 4.

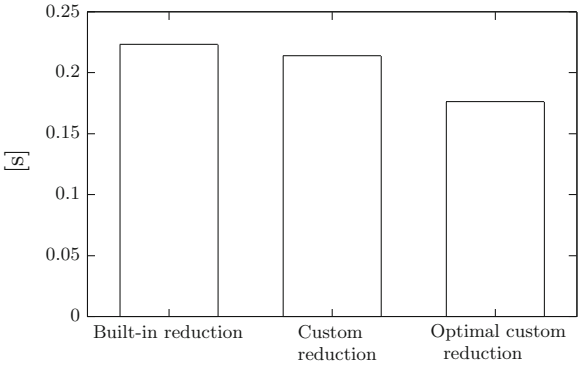


Fig. 4 Parallel times of various reduction algorithms

6 Conclusion

The paper shows that the built-in OpenMP support for parallel reduction provided by reduction clause is not neither theoretically- nor practically-optimal at least for some sorts of applications and calculations. It shows that the performance of parallel reduction algorithms can be improved significantly by using custom optimized implementations. However, it is important to note that the measured and discussed results can differ from other specific implementations and usage scenarios, mainly if other compiler-level code optimizations and reduction operators with different time complexity were used.

The paper also shows that the scalability of even trivial OpenMP algorithms is sufficient for various parallel reduction implementations.

Acknowledgments The research was supported by the European Regional Development Fund under the project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089.

References

1. Michael, J.Q.: Parallel Programming in C with MPI and OpenMP. McGraw-Hill Education, New York (2008)
2. GCC online documentation—GNU Project—Free Software Foundation. http://gcc.gnu.org/onlinedocs/
3. GNU libgomp. http://gcc.gnu.org/onlinedocs/gcc-4.8.2/libgomp/
4. OpenMP.org Website. http://openmp.org/wp/
5. Chandra, R.: Parallel programming in OpenMP. Morgan Kaufmann Publishers, San Francisco (2001)
6. Abd-El-Barr, M., El-Rewini, H.: Fundamentals of computer organization and architecture. Wiley, Hoboken (2005)
7. Tvrdik, P.: Parallel Systems and Algorithms. CVUT Press, Prague (1995)

User Testing and Trustworthy Electronic Voting System Design

Petr Silhavy, Radek Silhavy and Zdenka Prokopova

Abstract In this contribution the user interface design for trustworthy system is presented. The principle of the Electronic Voting is discussed. The research aim was to discuss a users trust and its issues, which are connected to the design process of the prototype electronic voting system.

Keywords Electronic voting · System design · User testing

1 Introduction

Discussing of the trustworthiness in the scope of the system engineering discipline takes key role in systems designing. If trustworthiness is discussed reliability, safety and security are discussed. The issue of trustworthiness is connected to the level of the user acceptance [1]. The word "system" stands for on-line systems. It means systems, which are realized on basics of Internet, or similar communication systems.

The organization of this contribution is as follows. Section 1.1 describes a basic electronic voting theory and describes electronic voting conditions. Section 2 describes the problem formulation. Section 3 describes a Study Design. Section 4 describes the selected questions, which have to be solved to achieve trustworthiness of the electronic voting. Finally Sect. 5 is the discussion.

P. Silhavy (✉) · R. Silhavy · Z. Prokopova
Faculty of Applied Informatics, Tomas Bata University in Zlin, Nam. T.G.M. 5555 76001 Zlin, Czech Republic
e-mail: psilhavy@fai.utb.cz

R. Silhavy
e-mail: rsilhavy@fai.utb.cz

Z. Prokopova
e-mail: prokopova@fai.utb.cz

R. Silhavy et al. (eds.), *Modern Trends and Techniques in Computer Science*, Advances in Intelligent Systems and Computing 285, DOI: 10.1007/978-3-319-06740-7_53,

1.1 Electronic Voting in Brief

The electronic government, the actual point of the computer-social investigation, uses several methods for improving governmental processes in Europe. These electronic methods allow the improvement in direct democracy. Probably, the most relevant solution is remote Internet voting. The remote voting solution allows participation in election process with respect to personal conditions and to the physical accessibility of polling stations, which may possibly prevent citizens from casting their votes. Therefore, remote Internet voting should effectively support voters, who are resident abroad. These voters use at present the embassy election rooms only.

There are several conditions for electronic voting systems, which were discussed several times and now are accepted as facts. The appropriate system has to follow the technical and process conditions listed below:

- Participation in the voting process is granted only for registered voters.
- Each voter has to vote only once.
- Each voter has to vote personally.
- Security and anonymity of voters and voting.
- Security for the electronic ballot box.

The first condition for electronic voting means, the voter should be registered by voting committee in the list of voters. This list is used as the basis for distribution of login information. If the voter is registered, they will be able to display the relevant list of parties and candidates.

Voters could also vote more than once, but only the last attempt will be included in the final results of the election. This possibility varies in different e-voting systems. If it is not possible to vote more than once, there should be more complicated protection for the election against manipulation and assisted voting.

The third condition—Right to vote personally—is closely connected to the previous. On the other hand this is the basic responsibility of each voter to protect his private zone for voting—in the case of the internet-based remote voting. In the "in-site" voting the system of privacy protection will be similar to the current situation.

Security and anonymity of voters and voting is probably the most important issue in the electronic voting process. The appropriate voting system should be realized in two separate parts. The first part should be responsible for authorization of the voter and the second for storing votes. Therefore the system will support anonymity. The voter should check his vote by the list of collected votes. The voter will know the unique identification of vote only. Using a cryptographic principle will protect the voting process. One of the many applicable solutions is Private Key Infrastructure. This approach deals with two pairs of keys in the first part of voting system—for authorization. In the second part of voting system—storing votes—it should deals with a public key for protection of the vote in the transport canal.

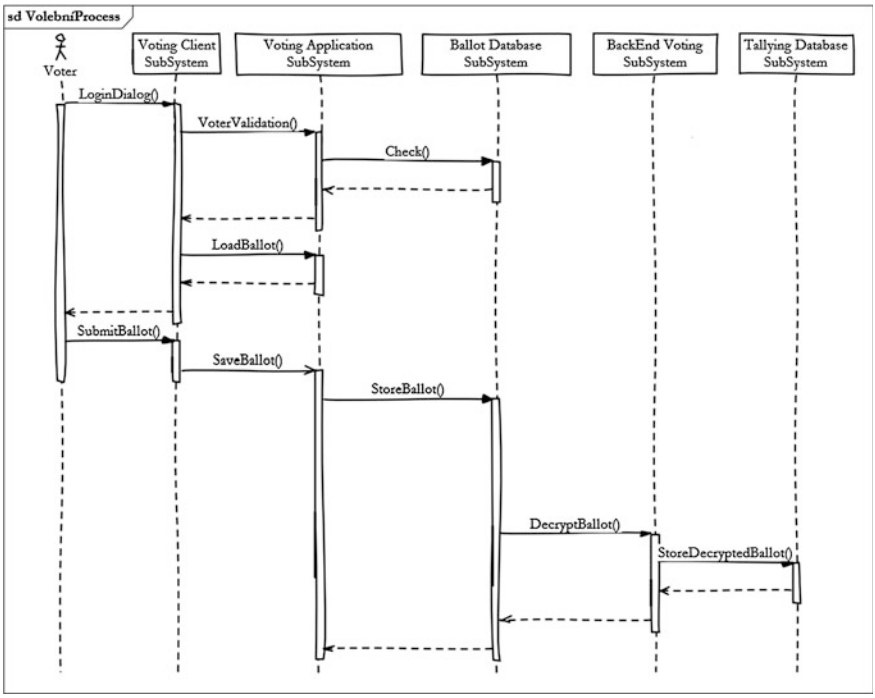


Fig. 1 Generic voting schema design [3]

The electronic ballot box should form as a database. The public key of the election committee will cipher votes in the database. Members of the committee will hold the private key, which is necessary for decrypting votes. Each member will hold only part of the key.

The sample system, which follows the defined requirements, is shown on Fig. 1 [3].

Probably, the most significant results is remote Internet voting in appropriate form, which understandable and clearly usable.

Internet voting solutions are usually divided the tree basic categories—poll site, kiosk and remote voting.

In the poll site voting, election technology is located in the election rooms. Comparing poll site voting to traditional paper-form voting, poll site brings more flexibility and usability, because voters are allow to vote from elections room up to their choice. There is no restriction to geographical locations. Poll site voting represents concept of the electronic voting. Poll site voting is effective in votes casting and tallying, because it allows certain and quicker processes.

Internet concepts allow expanding poll site voting to self-service kiosks. These kiosks should be placed in various locations. Authorities—local election committee, usually monitor elections room; kiosk should be monitored by physical attendance or by using security cameras.

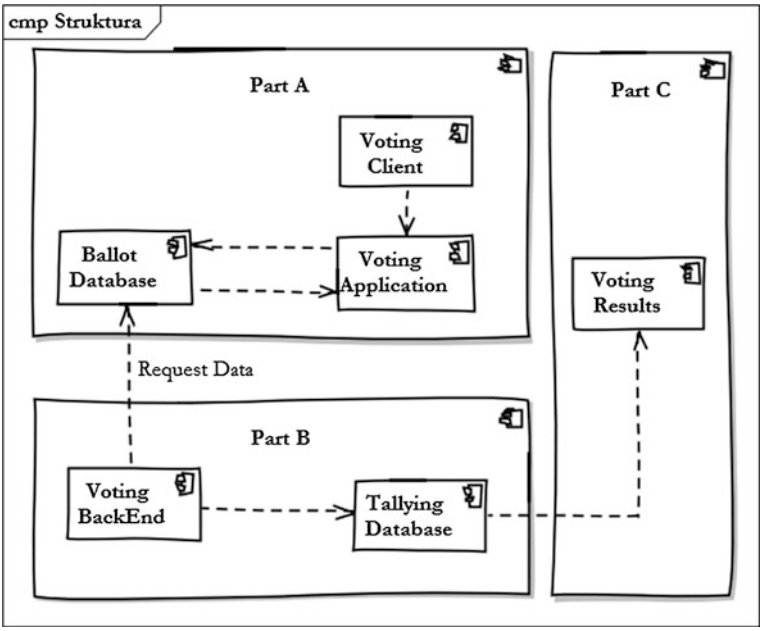


Fig. 2 Electronic voting architecture design [3]

Remote Internet voting is probably the most attracting methods of using Internet voting process. Remote voting expands remote voting schemas, which are used in some countries. These schemas compared to postal voting, offers improved casting ballots from remote locations. Voters are allows voting from home, office or other places, which are equipped by computers and Internet connections.

By investigation of these conditions and by the determination of the initial technological principles, authorities will be able to establish law to support the electronic voting system. The voting public's consensus to the electronic voting is quite important for the parliament process too.

The typical electronic voting architecture can be found in the Fig. 2. [3]. The web-based approach is useful for electronic voting systems. This technology is based on a client-server. The client-server technology has advantages in the field of support and installation.

The voting system consists four main parts [3]:

- Voting Client Subsystem (VCS).
- Voting Application Subsystem (VAS) and Ballot Database Subsystem (BDS).
- Voting Backend Subsystem (VBS) and Tallying Database Subsystem (TDS).
- Voting Results Subsystem (VRS).

In the Fig. 2 can be seen Voting System Architecture. The three separate parts are recognized [3]. Part A is used for casting votes and contains Voting Client Subsystem, Voting Application Subsystem and Voting Database Subsystem.

Voting clients represent voting terminal in elections rooms, kiosk voting or voters own computers.

Voting Application Subsystem is represented by web-based application, which contains user interface for voters, voter validation services and communication interface for Ballot Database Subsystem.

There are two most significant tasks for BDS. Votes are cast there and default ballots are generated for individual voter. Votes are cast in encrypted form, which depends on cryptographic methodology adopted for the election. For the protection against manipulation with votes in BDS HASH algorithm is implemented. HASH value is calculated irregularly based on votes, which are cast. Default ballots are generated for individual voter with respect to the election district he belongs to.

Part B represents Backend Voting Subsystem and Tallying Database Subsystem. The part B is securely connects to BDS from part A. The BVS is used my electoral committee. The BDS is responsible for auditing elections by comparing HASH based on votes and stored HASH value. The BVS deals with decryption of votes, validating of them and storing in TDS. The TDS is used for storing votes in open form. Part B is realized as web-based application and relational database server. Final part—part C—is responsible for counting final Results of the election. Part C is realized as web-based application.

Finally the last important aspect is the marketing. This classical business case is mention because there is very close link between user trust and how the system is presented and descripted to the public—future users. We have disused already, that user trust in technological system is based on social or more precisely mental aspects. It means users only believes or not, that system is reliable and then they trust to the system. Therefore the role of marketing seems to be significantly important. We do not deal with manipulation but with methods, how the system is presented to the users. The users need to be able to use the system in advance. In public election, is also important if the local authorities have appropriate level of the credibility.

2 Problem Formulation

In the socio-technical systems a human or humans which act as a users are taking important role. The system non-function requirements that are influencing the trustworthiness are safety, security and reliability. Those three basic aspects cannot be achieved by the technical design only [4]. The system architect has design the system with emphasis on process or procedural part of the system.

In the system engineering we usually follows basic recommendation about the system design which resulting in the reliable system. A reliable system does not equal to trustworthy system. The reliability refers to the system characteristics. It means, that users can use a system, which activity has no hazard state. The main task of this paper is to present the important characteristics, which take the key

role in user's final system evaluation. Furthermore success of the on-line system is dependent on users acceptance.

The electronic voting is an example of the socio-technical systems. Technologically, there is chance to achieve an appropriate level of the reliability [5] and trustworthiness. Users of such system—voters—have usually difficulties to trust e-voting systems.

The reasons can be found in a fact of black-box design. In the following chapter will evaluate such system by using experimental study by using set of users and we will present a solution based on results of such study.

3 Study Design

The system described in Sect. 1.1, were used as input for our experiment. We have evaluated the system design in controlled environment by using a group of students. There were 100 person examined in our study. In the Table 1, there can be seen, that second and third grades were interested in study as two most important groups.

For the study we deals only with students, which are informatics or non-informatics discipline. In the Table 2, you can find the summery involved disciplines.

As can be seen students, which are studied informatics related discipline are able to cooperate on such study more frequently.

3.1 Survey Design

There were nine basic questions, which were evaluated in our study. We have study each question in interval of four replies. In the Table 3, you can see list of question and scales, which were pre-prepared for each of examined user.

The structured questions were followed by open text filed, where each of examined subjects can enter individual response to presented voting schema design.

4 Results

As can be seen in Table 3 there were tree questions, which represents general attitude to electronic voting. Firstly as can be seen in Fig. 3, more than 70 % of participants believe, that electronic voting can be useful for improving general attitude to autonomy.

Table 1 Percentage of students in sample by study grade

Study years	Percentages
First grade	3
Second grade	26
Third grade	25
Fourth grade	17
Fifth grade	24
Doctoral grades	5

Table 2 Study discipline overview

Study discipline	Percentages
Informatics	70
Non-informatics	30

Table 3 List of questions used for examination

Question	Scale
System processing speed	
System controls satisfaction	
User interface design	Strongly agree
Instruction for users	Agree
Support for EV idea	Disagree
Can EV simplify participation?	Strongly disagree
Can EV improve participation?	

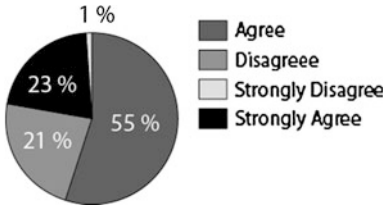


Fig. 3 Support for electronic voting

More than 97 % agree or strongly agree that electronic voting can improve participation in closed election. In the Fig. 4, you can see that only 3 % disagree.

The majority of participants believe that electronic voting can simplify voting participation and speed-up voting process. In the Fig. 5 you can see that only 2 % disagree, that electronic voting can guarantee simplification of the voting process.

In the Table 4 can be found the summary of user survey, which is focused on user experiences with proposed voting schema an its representation in prototype system.

In Fig. 6, there can be seen graphical representation of results, presented in Table 4.

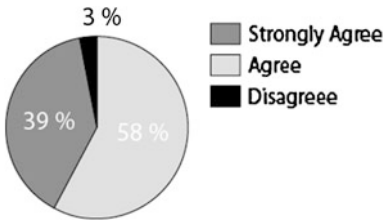


Fig. 4 Improving participation

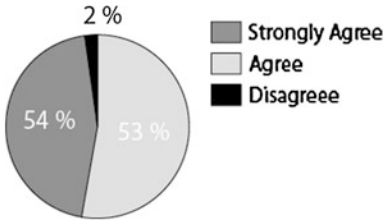


Fig. 5 Simplification of voting process

Table 4 Summary of user evaluation

Question	Strongly agree	Agree	Disagree	Strongly disagree
System processing speed	45	50	3	2
System controls satisfaction	30	65	2	1
User interface design	28	52	17	3
Instruction for users	30	60	8	2

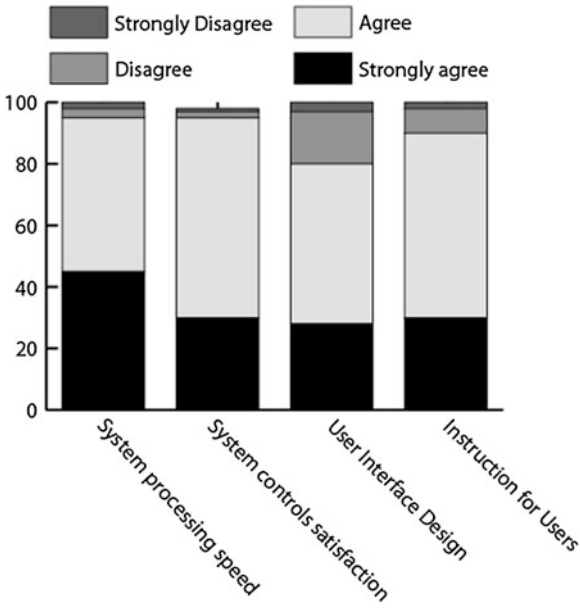


Fig. 6 Results overview

The following system was designed as results of our study. The system is based comments and user requirements, which were gathered in our study. We proposed a system for the closed election at the academic level.

5 Conclusion

The idea of the research was to discuss a users trust and its issues, which are connected to the electronic voting. The electronic voting seems to be more problematic in case of system trustworthiness.

The voting system, the electronic voting systems is the new approach. Users have no personal experience with using such system. According [6, 7] there is a difference between a trust and a confidence. If the users have no choice or not consider alternatives solutions, they have confidence to the system. In this chapter we deal with trust, because we expected, that users have choice vote by electronic voting or by well-know legacy paper ballot based system.

Firstly, users usually trust the system if they are familiar with it. Voters trust traditional voting concepts, because they are used to participate for many years. Breaking the barrier is possible by making the electronic voting optional for participation in the election. If the voters will have the opportunity to use—study the system, they will build trustworthiness.

The system complexity is reduced if the users are familiar with the system or with similar system.

People have to believe, that electronic voting is better than legacy version of the voting process. The user familiarity is based on the user interface design.

User makes mistakes. The user interface has to follow simple schema of the voting process and only limited number of the information have to be shown. The proper design of the voting system user interface is based on showing the appropriate amount of the information. The voting schema has to follow the traditional concepts of the elections.

The user satisfaction is rising if the system behavior is predictable. The users build their model or presumptions. Therefore the predictability how the system will behave is significantly important for building system trustworthiness. The second fact, which is very closed to previous discussion, is a communication between the user and system. User have to be sure, that system will be able to inform users about its state or activity. Previous thoughts are resulting in interaction design and in the principle of consistency. Consistency is a prerequisite of the predictability. Users expect, that same command or similar command will cause similar behave of the system.

Users expect, that system will use an appropriate interaction styles. In this case we will discuss ability of confirmation messages and error messages. The ballot casting in the electronic voting is a serious task. Many of users will think about the system in that way. Therefore is useful to create deep analysis of interactions. In interaction design we deals not only with the steps of the voting schema, but also

how the step should be achieved. The implantation of each step has to be non-destructive. Moreover with ability to make a back-step without data lost. The system should be communicative. It means, that each of steps will contain a confirmation message.

Secondly we will discuss a technological aspects and its impact to trustworthiness. As was shown in previous paragraphs, trust is more sociological than technological issue. This obvious fact is based [5] on the situation in which users make a certain type of the risk analysis. The mentioned analysis is subjective. Subjective analysis is usually based on user interface. The technical point deals with objective risk analysis. There is an issue, because users have to understand the system internal processes. It is impossible to for common user to understand a complex system internal processing.

In the scope of the voting system, they have been familiar with voting schema and number of non-trivial technologies. In [3], there can be found an analysis of reliable voting schema. The verification is based on mathematical approach and on empirical approach.

The key factors why users trust to the systems can be found in similarity. Users are used to trust to the similar systems. Many of the users are able to use an e-commerce system, electronic banking or electronic payments systems.

The similarity of systems is based on their core components. E-voting system and other e-processing system contains communication over the Internet, cryptography, and authentication. This fact should have a positive impact to building user trust.

The system architecture has only limited influence on the system trustworthiness. But has significant impact on user trust. Users = voters response, that trusts in the electronic voting system is based on anonymity and auditability. Users need is to check, that their ballots is counted in proper way. In means the each successful electronic voting schema have to implement a mechanisms for such control system.

Trustworthiness can be defined as user relation to the system or software solution. Firstly, users are building their option and thoughts on the experiences with the system itself. Secondly, users attitude is based on experience with the similar systems. The similarity of systems is based on their core components. E-voting system and other e-processing system contains communication over the Internet, cryptography, and authentication. This fact should have a positive impact to building user trust.

The electronic voting system design has to reflect the situation in the concrete society. There is no silver bullet solution, which is applicable everywhere. The user interface of the system should reflect the tradition of the ballot design. The basic principles of the design should reflect the core electronic voting issues—privacy, security.

This research work is not limited to potential of the electronic voting. Therefore the results and ideas should be valid for e-processing system in general.

Further research is focused on the improvement of the electronic voting, particularly in security and privacy, which seem to be important for user trust. In addition, issues connected to the cohesion among voting technology, legal principles and public attitude should be under the investigation.

References

1. Silhavy, R., Silhavy, P., Prokopova, Z.: Systematic modeling process of system behavior. Int. J. Math. Models Methods Appl. Sci. **5**(6), 1044–1051 (2011) [cit. 2012-10.08]. ISSN 1998-0140
2. Quarda, H.: Cognitive tasks behavior of intelligent autonomous mobile robots. Int. J. Math. Models Methods Appl. Sci. **5**(3), 610–619 (2011). ISSN 1998-0140
3. Silhavy, R., Silhavy, P., Prokopova, Z.: Architecture of COOPTO remote voting solution. Paper presented at the advanced techniques in computing sciences and software engineering, pp. 477–479 (2010)
4. Balla, J.: Dynamics of mounted automatic cannon on track vehicle. Int. J. Math. Models Methods Appl. Sci. **5**(1), 423–432 (2011). ISSN 1998-0140
5. Ciulanescu, M.V., Diaconu, A.: Mobile robot control using the bluetooth technology. In: Katalinic, B. (ed.) Annals of DAAAM for 2009 and Proceedings of the 20th International DAAAM Symposium, 25–28 Nov 2009, Vienna, Austria, ISSN 1726-9679, ISBN 978-3-901509-70-4, pp. 1115–1116. Published by DAAAM International Vienna, Vienna (2009)
6. Ribu, K.: Estimating Object-Oreinted Software Projects with Use Case. Unvirstiy in Oslo, Oslo (2001)
7. Silhavy, R., Silhavy, P., Prokopova, Z.: Clustered requirements in system engineering project estimation. Int. J. Math. Models Methods Appl. Sci. **5**(1), 1052–1059 (2011). ISSN 1998-0140