Advances in Intelligent Systems and Computing 470

Wojciech Zamojski
Jacek Mazurkiewicz
Jarosław Sugier
Tomasz Walkowiak
Janusz Kacprzyk *Editors*

Dependability Engineering and Complex Systems

Proceedings of the Eleventh International Conference on Dependability and Complex Systems DepCoS-RELCOMEX. June 27–July 1, 2016, Brunów, Poland

Springer

Advances in Intelligent Systems and Computing

Volume 470

Series editor

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

About this Series

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

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

Advisory Board

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

Wojciech Zamojski · Jacek Mazurkiewicz
Jarosław Sugier · Tomasz Walkowiak
Janusz Kacprzyk
Editors

Dependability Engineering and Complex Systems

Proceedings of the Eleventh International Conference on Dependability and Complex Systems DepCoS-RELCOMEX.
June 27–July 1, 2016, Brunów, Poland

Springer

Editors
Wojciech Zamojski
Department of Computer Engineering
Wrocław University of Technology
Wrocław
Poland

Tomasz Walkowiak
Department of Computer Engineering
Wrocław University of Technology
Wrocław
Poland

Jacek Mazurkiewicz
Department of Computer Engineering
Wrocław University of Technology
Wrocław
Poland

Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
Warsaw
Poland

Jarosław Sugier
Department of Computer Engineering
Wrocław University of Technology
Wrocław
Poland

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-39638-5 ISBN 978-3-319-39639-2 (eBook)
DOI 10.1007/978-3-319-39639-2

Library of Congress Control Number: 2016941311

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG Switzerland

Preface

In this volume of "Advances in Intelligent Systems and Computing," we are pleased to present proceedings of the Eleventh International Conference on Dependability and Complex Systems *DepCoS-RELCOMEX* which took place in a picturesque Brunów Palace in Poland from June 27 to July 1, 2016. It was an event in a series organized annually by Department of Computer Engineering of Wrocław University of Science and Technology since 2006 although its heritage is much older. It dates nearly 40 years back and begun with two cycles of events: RELCOMEX (1977–89) and Microcomputer Schools (1985–95) which were then organized by the Institute of Engineering Cybernetics (predecessor of the Department) under the leadership of Prof. Wojciech Zamojski, still the DepCoS chairman. In contrast to those previous events focused on classical reliability analysis, the DepCoS mission is to promote a more comprehensive approach which in the new century has earned the name *dependability*. Products of the conferences were initially published by the IEEE Computer Society (2006–09), then also by Wrocław University of Technology Publishing House (2010–12) and presently by Springer in "Advances in Intelligent Systems and Computing" Volume nos. 97 (2011), 170 (2012), 224 (2013), 286 (2014), and 365 (2015).

Design, implementation, and maintenance of contemporary complex systems have brought many new challenges to "classic" reliability theory. The complex systems are understood by us as integrated unities of technical, information, organization, software, and human (users, administrators, and management) assets, and their complexity comes not only from involved technical and organizational internal structure built upon diverse hardware and software resources but also from complexity of information processes (data processing, monitoring, management, etc.) which must be executed in their specific environment. In operation of such wide-ranging (and often also geographically distributed) systems, their resources are dynamically allocated to ongoing tasks and the rhythm of system events (incoming and/or ongoing tasks, decisions of a management subsystem, system faults, defensive system reactions and adaptations, etc.) may be considered as deterministic and/or probabilistic stream of events. Security and confidentiality issues

enforced by social context of information processing introduce further complications into the modelling and evaluation methods. Diversity of the processes being realized, their concurrency and their reliance on in-system intelligence often make construction of strict mathematical models impossible and lead to application of intelligent and soft computing methods.

Dependability is the contemporary answer to new challenges in reliability evaluation of such systems. Dependability approach in theory and engineering of complex systems (not only computer systems and networks) is based on multidisciplinary approach to system theory, technology, and maintenance of the systems working in real (and very often unfriendly) environment. Dependability concentrates on efficient realization of tasks, services, and jobs by a system considered as a unity of all technical, information, and human assets, in contrast to "classical" reliability which is more restrained to analysis of technical resources (components and structures built from them). This difference has shaped natural evolution in topical range of subsequent DepCoS conferences which can be seen over the recent years.

The program committee of the 11th International DepCoS-RELCOMEX Conference, its organizers, and the editors of these proceedings would like to gratefully acknowledge participation of all reviewers who helped to refine contents of this volume and evaluated conference submissions. Our thanks go to, in alphabetic order, Andrzej Białas, Frank Coolen, Manuel Gil Perez, Zbigniew Huzar, Vyacheslav Kharchenko, Jan Magott, Jacek Mazurkiewicz, Marek Młyńczak, Tomasz Nowakowski, Yiannis Papadopoulos, Oksana Pomorova, Krzysztof Sacha, Janusz Sosnowski, Jarosław Sugier, Victor Toporkov, Tomasz Walkowiak, Marina Yashina, Irina Yatskiv, Wojciech Zamojski, and Włodzimierz Zuberek.

Thanking all the authors who have chosen DepCoS as the publication platform for their research, we would like to express our hope that their papers will help in further developments in design and analysis of complex systems, being a valuable source material for scientists, researchers, practitioners, and students who work in these areas.

Wojciech Zamojski
Jacek Mazurkiewicz
Jarosław Sugier
Tomasz Walkowiak
Janusz Kacprzyk

Organisation Committee

Programme Committee

Wojciech Zamojski (Chairman), *Wrocław University of Science and Technology, Poland*
Salem Abdel-Badeeh, *Ain Shams University, Abbasia, Cairo, Egypt*
Ali Al-Dahoud, *Al-Zaytoonah University, Amman, Jordan*
George Anders, *University of Toronto, Canada*
Artem Adzhemov, *Technical University of Communications and Informatics, Moscow, Russia*
Włodzimierz M. Barański, *Wrocław University of Science and Technology, Poland*
Andrzej Białas, *Institute of Innovative Technologies EMAG, Katowice, Poland*
Ilona Bluemke, *Warsaw University of Technology, Poland*
Eugene Brezhniev, *National Aerospace University "KhAI," Kharkiv, Ukraine*
Alexander Buslaev, *Moscow Technical University of Communication and Informatics, Moscow, Russia*
Dariusz Caban, *Wrocław University of Science and Technology, Poland*
Krzysztof Cios, *Virginia Commonwealth University, Richmond, USA*
Frank Coolen, *Durham University, UK*
Mieczysław Drabowski, *Cracow University of Technology, Poland*
Francesco Flammini, *University of Naples "Federico II," Napoli, Italy*
Manuel Gill Perez, *University of Murcia, Spain*
Zbigniew Huzar, *Wrocław University of Science and Technology, Poland*
Igor Kabashkin, *Transport and Telecommunication Institute, Riga, Latvia*
Janusz Kacprzyk, *Polish Academy of Sciences, Warsaw, Poland*
Vyacheslav S. Kharchenko, *National Aerospace University "KhAI," Kharkiv, Ukraine*
Mieczysław M. Kokar, *Northeastern University, Boston, USA*
Krzysztof Kołowrocki, *Gdynia Maritime University, Poland*
Leszek Kotulski, *AGH University of Science and Technology, Krakow, Poland*
Henryk Krawczyk, *Gdansk University of Technology, Poland*

Alexey Lastovetsky, *University College Dublin, Ireland*
Marek Litwin, *ITS Polska, Warsaw, Poland*
Jan Magott, *Wrocław University of Science and Technology, Poland*
Istvan Majzik, *Budapest University of Technology and Economics, Hungary*
Jacek Mazurkiewicz, *Wrocław University of Science and Technology, Poland*
Marek Młyńczak, *Wroclaw University of Science and Technology, Poland*
Yiannis Papadopoulos, *Hull University, UK*
Oksana Pomorova, *Khmelnitsky National University, Ukraine*
Ewaryst Rafajłowicz, *Wrocław University of Science and Technology, Poland*
Nikolay Rogalev, *Moscow Power Engineering Institute (Technical University), Russia*
Krzysztof Sacha, *Warsaw University of Technology, Poland*
Rafał Scherer, *Częstochowa University of Technology, Poland*
Mirosław Siergiejczyk, *Warsaw University of Technology, Poland*
Czesław Smutnicki, *Wrocław University of Science and Technology, Poland*
Janusz Sosnowski, *Warsaw University of Technology, Poland*
Jarosław Sugier, *Wrocław University of Science and Technology, Poland*
Victor Toporkov, *Moscow Power Engineering Institute (Technical University), Russia*
Tomasz Walkowiak, *Wrocław University of Science and Technology, Poland*
Max Walter, *Siemens, Germany*
Bernd E. Wolfinger, *University of Hamburg, Germany*
Min Xie, *City University of Hong Kong, Hong Kong SAR, China*
Marina Yashina, *Moscow Technical University of Communication and Informatics, Russia*
Irina Yatskiv, *Transport and Telecommunication Institute, Riga, Latvia*
Jan Zarzycki, *Wrocław University of Science and Technology, Poland*
Włodzimierz Zuberek, *Memorial University, St. John's, Canada*

Organizing Committee

Wojciech Zamojski (Chairman)
Włodzimierz M. Barański
Monika Bobnis
Jacek Mazurkiewicz
Jarosław Sugier
Tomasz Walkowiak

Contents

Multiclass SVM/HMM Vowels Recognition System Towards Improving Human Computer Interaction

Ali Al-Dahoud, Mohamed Fezari, Abadi Wassila and Tahmer Al-Rawashdeh

Abstract In this paper, we present experimental results on using vowels and one syllable words (VOSW) to activate some input device such as the control of mouse pointer on the screen. This type of Human Computer Interface might be used by a category of disabled person such as upper arm and finger stroke. Our goal is to design a system for the control of mouse cursor based on voice input command, using the pronunciation of certain vowels and some syllable words which can be produced even by persons affected by chronic inflammation of the larynx and vocal fold nodules. Linear predictive coefficients (LPC) and Mel Frequency Cepstral Coefficients (MFCCs) with derivatives are selected as features. Multiclass SVM then Hidden Markov Models (HMMs) have been tested as classifiers for matching components (vowels and short words). Tests and results using different features were are presented on tables, then the two classifiers were experimented independently and results were registered. Finally a verbal user interface has been designed and experienced on web page browser.

Keywords Human computer interface · Syllable · LPC · Multiclass SVM · Binary tree

1 Introduction

Basic Human computer interfaces are useless for a category of disabled people especially those who have problems performing fine motor movement with their hand and fingers; they face difficulties when using mouse or keyboard devices, and

A. Al-Dahoud (✉) · T. Al-Rawashdeh
Faculty of IT, Al-Zaytoonah University of Jordan, Amman 11733, Jordan
e-mail: aldahoud@zuj.edu.jo

M. Fezari · A. Wassila
Faculty of Engineering, Badji Mokhtar Annaba University,
BP: 12, 23000 Annaba, Algeria
e-mail: mouradfezari@yahoo.fr

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_1

have to use alternative devices to accommodate their needs and capabilities. Many of these special devices can be very costly and therefore are not affordable to some users.

Increasing reliability and deploy-ability of the speech recognition software on the lower-end computer systems make this technology a promising alternative. However, the speech recognition systems operate on a query—response basis, where the system usually waits for the user to complete the utterance before responding. This makes the speech recognition inconvenient to use in real-time continuous tasks, including mouse pointer movement, where the minimum delay of the system is a critical feature of the feedback loop of the system. These last years, non-speech (or non-verbal) input has started to emerge. It is based on production of sounds other than speech by the user's vocal tract such as vowels, whistling and hums [1]. In non-speech input, the sound is analyzed and then certain features are extracted, such as the pitch profile, or the sound timbre, in order to solicit information from the user. We intend to explore some features and classifiers used in Automatic speech recognition (ASR).

Many voice characteristics are exploited in several works, but the most used are: energy [2, 3], pitch and vowel quality [4, 5] speech rate (number of syllables per second) and volume level [6]. However, Mel Frequency Cepstral Coefficients (MFCCs) [7–9] are used significantly in speech processing as features for automatic speech recognition and speaker identification.

2 Related Works

Some literature reviews are described in this paragraph. In the basic protocol, each vowel is associated to one direction for pointer motion [10], this technique is useful in situations where the user cannot use his or her hands for controlling applications because of permanent physical disability or temporal task-induced disability. The limitation of this technique is that it requires an unnatural way of using the voice [11, 12]. Control by generating Continuous Voice: In this interface, the user's voice works as an on/off button. When the user is continuously producing vocal sound, the system responds as if the cursor of mouse is moved or button pressed. When the user stops the sound, the system stops moving the cursor or recognizes that the button is released. For example, one can say "Volume up, aaah", and the volume of a radio set continues to increases while the "aaah" continues. The advantage of this technique compared with previous approach presented in [12] where the user would say instead "Volume up twenty" or something is that the user can continuously observes the immediate feedback during the interaction.

Harada et al. [4] presented a novel voice-based human computer interface designed to enable individuals with motor impairments to use vocal parameters for continuous control tasks.

Salem in [13] has developed an isometric tongue pointing device that can be incorporated into a variety of applications such as mouse and menu control, or robotic arm manipulation. Our design goal is to be modular, low-latency, and as computationally efficient as possible. The first of those, localized acoustic energy is used for voice activity detection, and it is normalized relatively to the current detected vowel, and is used by our mouse application to control the velocity of cursor movement. The second parameter, "pitch", is not used currently but it is left for the future use.

In [5, 14] Sporka presented a work based on Whistling on "User Interface (U3I)", based on the use of tones (in whistling or humming) where the difference between the initial pitch and current pitch determines the speed of motion.

In [7], Thiang et al., described the implementation of speech recognition system on a mobile robot for controlling movement of the robot. The methods used for speech recognition system are Linear Predictive Coding (LPC) and Artificial Neural Network (ANN). LPC method is used for extracting feature of a voice signal and ANN is used as the recognition method. Back propagation method is used to train the ANN. Experimental results show that the highest recognition rate that can be achieved by this system is 91.4 %. This result is obtained by using 25 samples per word, 1 hidden layer, 5 neurons for each hidden layer, and learning rate 0.1.

3 Feature Extraction

3.1 MFCC Feature Extraction [8, 9]

The overall process of the MFCC can be presented in the following steps:

1. After the pre-emphasis filter, the speech signal is first divided into fixed-size windows distributed uniformly along the signal.
2. The FFT (Fast Fourier Transform) of the frame is calculated. Then the energy is calculated by squaring the value of the FFT. The energy is then passed through each filter Mel. S_k is the energy of the signal at the output of the filter K, we have now m_p (number of filters) S_k parameters.
3. The logarithm of S_k is calculated.
4. Finally, the coefficients are calculated using the DCT (Discrete Cosine Transform).

3.2 Linear Predictive Components

The basic idea behind linear predictive analysis is that a specific speech sample at the current time can be approximated as a linear combination of past speech samples. Through minimizing the sum of squared differences (over a finite interval)

between the actual speech samples and linear predicted values a unique set of parameters or predictor coefficients can be determined.

LPC computation basic steps are presented in [8, 9]:

4 Classifies

4.1 Support Vector Machine (SVM)

Binary SVM, in their general form, extend an optimal linear hypothesis, in terms of an upper bound on the expected risk that can be interpreted as the geometrical margin, to non linear ones by making use of kernels k(.,.). Kernels can be interpreted as dissimilarity measures of pairs of objects in the training set X. In standard SVM formulations, the optimal hypothesis sought is of the form (1).

$$\Phi(\xi) = \sum \alpha_i k(x, x_i) \tag{1}$$

where α are the components of the unique solution of a linearly constrained quadratic programming problem, whose size is equal to the number of training patterns. The solution vector obtained is generally sparse and the non zero α's are called support vectors (SV's). Clearly, the number of SV's determines the query time which is the time it takes to predict novel observations and subsequently, is critical for some real time applications such as speech recognition tasks.

It is worth noting that in contrast to connectionist methods such as neural networks, the examples need not have a Euclidean or fixed-length representation when used in kernel methods. The training process is implicitly performed in a Reproducing Kernel Hilbert Space (RKHS) in which k(x; y) is the inner product of the images of two example x, y.

4.2 SVM Formulation and Training

Support Vector Machines (SVMs) [15] are a popular discriminative classifier described extensively in the literature. They have been successfully applied to many different applications, such as text categorization [16]; speaker verification [17], image classification [18]. SVMs are based on the intuitive concept of maximizing the margin of separation between two competing classes, where the margin is defined as the distance between the decision hyper-plane and the closest training examples. _is has been shown to be related to minimizing an upper bound on the generalization error [15]. Support Vector Classifiers [15–18] solve the following linearly constrained convex quadratic programming problem:

$$\max iW(a) = \frac{1}{2}\sum a_i a_j y_i y_j k(x_i, x_j) - \sum_{i=1}^{m} a_i \quad (2)$$

Under the constraints: $\forall_i, 0 \le a_i \le C \sum_{i=1}^{m} a_i y_i = 0$

The optimal hypothesis is:

$$f(x) = \sum_{i=1}^{m} a_i y_i k(x, x_i) + b \quad (3)$$

where the bias term b can be computed separately as in [15, 16].

Clearly, the hypothesis f depends only on the non null coefficients α corresponding patterns are called Support vectors (SV).

This feature makes SVM very attractive for high input dimensional recognition problems and for the ones where patterns can't be represented as fixed dimensional real vectors such as text, strings, DNA etc. For large scale corpora however, the quadratic programming problem becomes quickly computationally expensive, in terms of storage and CPU time.

4.3 Binary Tree for Multiclass SVM

This approach uses multiple SVMs set in a binary tree structure. In each node of the tree, a binary SVM is trained using two classes. All samples in the node are assigned to the two sub nodes derived from the current node.

This step repeats at every node until each node contains only samples from one class. That said, until the leaves of the tree.

The main problem that should be considered seriously here is how to construct the optimal tree? With the aim of partitioning correctly the training samples in two groups, in each node of the tree. In this paper we propose a genetic algorithm for constructing a binary tree structure for multiclass SVM (see Fig. 1).

4.4 HMMs Basics [6]

Over the past, Hidden Markov Models have been widely applied in several models like pattern, or speech recognition. In Fig. 2 we present an HHM model for vowel. To use a HMM, we need a training phase and a test phase. For the training stage, we usually work with the Baum-Welch algorithm to estimate the parameters (π, A, B) for the HMM. This method is based on the maximum likelihood criterion. To compute the most probable state sequence, the Viterbi algorithm is the most suitable [6].

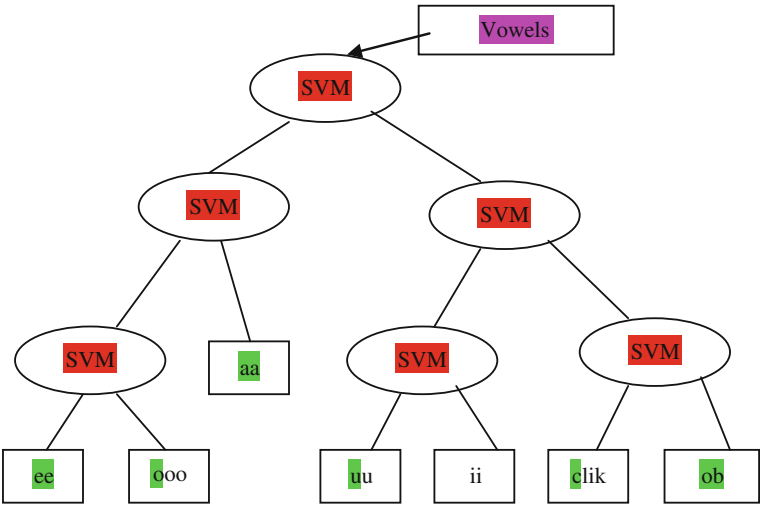


Fig. 1 Binary tree SVM for vowels

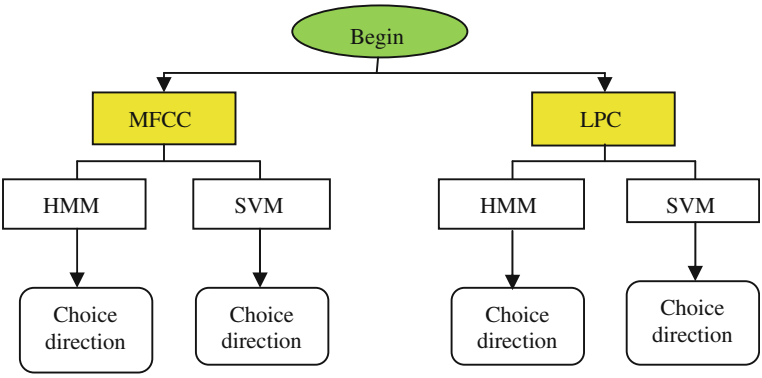


Fig. 2 Flowchart of the designed vowel recognition system

5 Description of Experiment

The experiment is designed to control the mouse cursor by using the pronunciation of certain phonemes and words, which we chose as vocabulary sustained voiles and short words: “aaah”, “ooh”, “iii”, “eeu”, “ou”, “uu”, “ik” for ‘clik’ and “ob” for ‘stop’, these two short words ‘ik’ and ‘ob’ are easy to be pronounced by person with voice pathology problems. We included some whistling tests for the future control.

Figure 2, illustrates the flowchart of he designed system. The choice of these vowels and short words is based on the following criteria: easy to pronounce, can

be pronounced persons with voice disorder, easy to model by automatic speech recognition system, based on features, the models can be discriminated from the others, easy to implement on embedded system such as DSP.

5.1 Participants Description

The participants in the experiment consist of 10 women (age 20–50 years), 10 men (age 20–60 years), and 5 children (age from 5 to 14 years) and category of persons with voice disorder from German database of the PTSD Putzer's voice in [19], 60 trials for each phoneme or word per day were performed by each user. Collection of the database is performed in a quiet room without noise. Figure 4 shows the flowchart of the application.

5.2 Features Extraction

Fundamental frequency and first and second formants were used then according to the tests and reviews, we found that the parameters more robust to noise than other parameters are the LPC coefficients and Mel Frequency Cepstral Coefficients (MFCCs).

The input signal is segmented by a window of 25 m overlapping 10 m, from each segment parameters were extracted by both methods LPC (the order of the prediction: 10) then MFCC (39 coefficients: Energy and derivative and second derivatives (3 * (12 coef MFCC +1 energy)).

5.3 Classification

For this moment, we have tested two classifier, first one has been used from previous work we have developed using ASR for robot command [18, 20], the main work were based on Hidden Markov chains (HMM) for classification phase.

For Hidden Markov models, in our system, we utilize left-to-right HMM structures with 3 states and 3 mixtures are used to model MFCCs coefficients and also the LPC coefficients.

6 Results and Discussions

For the testing phase, 30 % of recorded sounds are selected for each vowel or short word from the vocabulary.

In order to see the effect of training and making the system speaker independent, different scenarios for the tests were done, where we choose the results of recognition of three users out of database.

Some vowels and short words were correctly classified with some confusion, where a phoneme (or word) test classified as another phoneme (or word), the misclassification affected the results presented in Figs. 3 and 4. And it is clear that the confusion is higher in LPC features with SVM classifier while it is reduced using MFCC with HMM classifier.

According to the results presented above (Fig. 5), the recognition rates using MFCCs parameterization classification with SVM or HMM classification is better than: LPCs and MFCCs with SVM. So we can say that the MFCCs/HMM system is partially independent of the speaker.

Results using MFCC and HMM, on German database vowels (sounds) for persons with chronicle laryngitis that typically shows in varying harshness or breathiness. This is due to inflammation of vocal folds which reduces the ability of the vocal folds to vibrate. In the data base we selected files for the 8 speakers, we made tests and results are presented in Table 1.

We can see that the recognition rate is little bit lower than for healthy persons, we conclude that in this case other special features might be necessary to include on the application.

The work has been completed by a design of GUI (graphic user interface) to estimate the testability of the new HCI and Fig. 4 illustrates it, the GUI shows the vowel input in time spectral domain and the recognition vowel then the decision to be taken.

In Fig. 5a we present the developed GUI and in Fig. 5b, c we present screen capture of the application, in b and c we notice the direction of Mouse icon then by saying “clik” the web navigator ‘Google chrome’ is activated as shown in Fig. 5d.

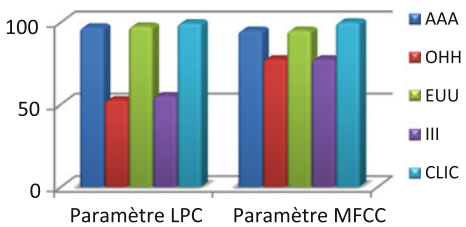


Fig. 3 Classification using LPCs, MFCC and SVM as classifier

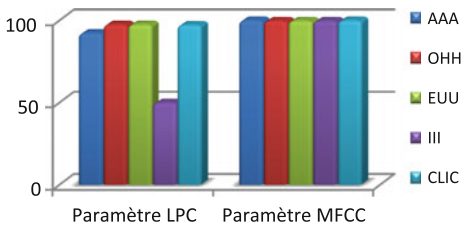


Fig. 4 Classification using LPC, MFCCs and HMM as classifier

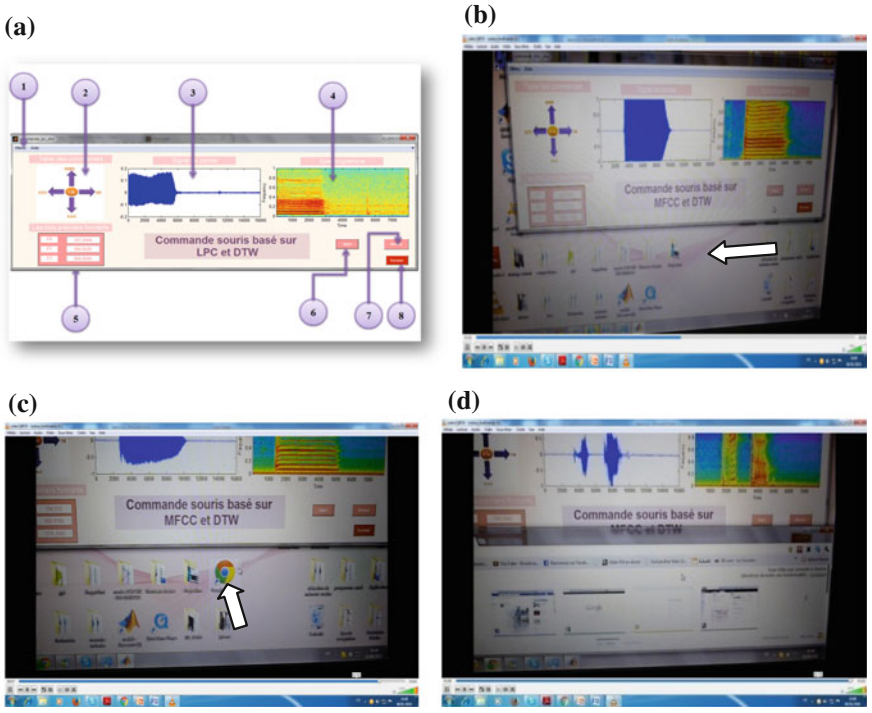


Fig. 5 GUI Scheme and screen capture of the application

Table 1 Classification using LPC and MFCC with HMM and SVM for Voweld form German DB

Vowel	LPC HMM (%)	MFCC HMM (%)	LPC SVM	MFCC SVM
aaa	59	78	45	65
ooh	45	67	37	54
eeu	49	81	39	59
iii	58	79	46	62
Clic or “eke”	55	73	42	61
Stop or “ebe”	57	78	41	59

In addition, we must consider the preprocessing for noise in future work, as well as the database training models need from the category of children. According to obtained results, we notice that the classification using HMM is better than the SVM, and the decision based on MFCC coefficients is efficient than the LPC coefficients.

7 Conclusion

Many participants mentioned that it was difficult to use the systems at the beginning but with practice, they were able to gain their skills. Participants were generally in agreement that Vowels were easier to learn than words. It was found more comfortable for voicing out vowels was a familiar activity for the users.

From experimental results, it can be concluded that MFCC features and HMM as classifier can recognize the speech signal well. Where the highest recognition rate that can be achieved in the last scenario. This result is achieved by using MFCCs and HMM. Moreover, we need to get better features to improve classification of vowel and short words pronounced from voice disabled persons; in fact this can be resolved by inserting Jitter and Shimmer as features.

We notified that the variety of signals, collected for database from different age and gender, the recording conditions and the environment, have a considerable impact in classification results.

The observation of novice users over time through a mixed of qualitative and quantitative methods shows the change in user experience as they gain more expertise in using the systems.

In future studies, it is important to ensure that mouse performance is captured. Secondly, only 25 users were involved in the experiment over a period of a week (even though each user performed 60 trials per day). So we will improve this part on database and use more trials in training phase, we would like to assess the project more on Manfred Putzer data Base since this category of population is more in need of this type of HMI.

References

1. Harada, S., Landay, J.A., Malkin, J., Li, X., Bilmes, J.A.: The vocal joystick: evaluation of voice-based cursor control techniques. In: Proceedings of the ASSETS 2006, Portland, Oregon (2006)
2. Dai, L., Goldman, R., Sears, A., Lozier, J.: Speech-based cursor control: a study of grid-based solutions. SIGACCESS Access. Comput. (77–78), 94–101 (2004)
3. Igarashi, T., Hughes, J.F.: Voice as sound: using non-verbal voice input for interactive control. In: ACM UIST (2001)
4. Harada, S., Landay, J., Malkin, J., Li, X, Bilmes, J.: The vocal joystick: evaluation of voice-based cursor control techniques. In: ASSETS'06 (2006)
5. Sporka, A.J., Kurniawan, S.H., Slavík, P.: Acoustic control of mouse pointer. Univ. Access Inf. Soc. **4**(3), 237–245 (2006)
6. Rabiner, L.: A tutorial on hidden markov models and selected applications in speech recognition. In: Proceedings of IEEE (1989)
7. Wijoyo, T.S.: Speech recognition using linear predictive coding and artificial neural network for controlling movement of mobile robot. In: International Conference on Information and Electronics Engineering IPCSIT, vol. 6, pp. 179–183 (2011)
8. Shaneh, M., Taheri, A.: Voice command recognition system based on MFCC and VQ algorithms. World Acad. Sci. Eng. Technol. **57**, 534–538 (2009)

9. Hadri, A., Boughazi, M., Fezari, M.: Improvement of arabic digits recognition rate based in the parameters choice. In: Proceedings of International Conference CISA Annaba (2008)
10. Bilmes, J.A. et al.: The vocal joystick: a voice-based human-computer interface for individuals with motor impairments. UWEE Technical Report Number UWEETR-2005-0007 (2005)
11. Abdeen, M., Moshammad, H., Yagoub, M.C.E.: An Architecture for Multi-Lingual Hands Free Desktop Control System for PC Windows, Niagara Falls. IEEE, Canada (2008)
12. Eggermont, P.P.B., LaRiccia, V.N..: Maximum Penalized Likelihood Estimation Volume I: Density Estimation. Springer, New York (2001)
13. Salem, C., ZhaS, I.: An isometric tongue pointing device. CHI 97 Electronic Publications: Technical Notes (1997)
14. Sporka, A.J., Kurniawan, S.H., Mahmud, M., Slavík, P.: Tonal control of mouse cursor (U3I): a usability study with the elderly. In: Proceedings of the HCI International 2005. Lawrence Erlbaum Associates (2005)
15. Cortes, C., Vapnik, V.: Support vector networks. Mach. Learn. pp. 273–297 (1995)
16. Burges, C.J.C.: A tutorial on support vector machines for pattern recognition. In: Data Mining and Knowledge Discovery, vol. 2, pp. 121–167 (1998)
17. Campbell, W.M., Sturim, D.E., Reynolds, D.A.: Support vector machines using gmm super vectors for speaker verification. In: IEEE Signal Processing Letters, vol. 13, pp. 308–311 (2006)
18. Chapelle, O., Haffner, P., Vapnik, V.: SVMs for Histogram-Based Image Classification (1999)
19. Putzer, M., Koreman, J.: A German Databse for a Pattern for Vacal Fold Vibration, Phonus 3. Institute of Phonetics, University of the Saarland, pp. 143–153(1997)
20. Fezari, M., Al-Dahoud, A.: An approach for: improving voice command processor based on better features and classifiers selection. In: The 13th International Arab Conference on Information Technology ACIT'2012, 10–13 Dec 2012, pp. 1–5 (2012)

Numerical Simulation of the Passenger Side Airbag Deployment in Out-of-Position

Driss Bendjaballah, Ali Bouchoucha and Mohamed Lakhdar Sahli

Abstract In recent years many accidents happened with passengers that were out of position when their airbag deployed. Therefore new safety regulations have come into effect and an airbag now has to meet several requirements that concern out of positioning. In meeting these requirements simulations of folding and deploying airbags are very useful and are widely used. The paper presents a simulation method for the deploying airbags using three materials in different working conditions. The deploying modeling of passenger side airbag is a complex and time consuming process. In these simulations the gas flow is described by the conservation laws of mass, momentum and energy. The main aim of this study is evaluate the performance of deploying of passenger side airbag using Finite Element Methods (FEM).

Keywords Airbags · Crash · Finite element simulations · Modeling · Out-of-position

1 Introduction

Nowadays, occupant safety is one of the principal objectives in the design of vehicles. Numerous innovations have appeared in recent years aimed at increasing safety in vehicles [1–3]. As is well known, airbags, like safety belts are now devices

D. Bendjaballah (✉) · A. Bouchoucha · M.L. Sahli
Mechanics Laboratory, Faculty of Technology Sciences,
Mentouri Brother University, Constantine, Algeria
e-mail: d_bendjaballah@hotmail.fr

A. Bouchoucha
e-mail: bouchoucha_ali1@yahoo.fr

M.L. Sahli
e-mail: Mohamed.sahli@femto-st.fr

M.L. Sahli
Femto-St Institute, Applied Mechanics Department,
CNRS UMR 6174, Besançon, France

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_2

designed to provide protection to the users of vehicles during crash events, minimizing the loads necessary to adapt their movement to the movement of the car [4, 5]. In general, the seat belt is designed to restrain the occupant in the vehicle and prevent the occupant from having harsh contacts with interior surfaces of the vehicles. The airbag acts to cushion any impact with vehicle structure and has positive internal pressure, which can exert distributed restraining forces over the head and face. Furthermore, the airbag can act on a wider body area including the chest and head, thus minimising the body articulations, which cause injury [6]. These safety elements can so reduce the death rates on the roads, and its protection effects have been widely approved [7, 8]. Thus, new types of airbag products are being developed to handle different collision scenarios.

In recent years, occupant protection airbags have become standard equipment on most new passenger vehicles [9–11]. The airbag cushion is composed of a woven fabric which is rapidly inflated during a car crash. The airbag dissipates the passenger's kinetic energy thereby reducing injury through biaxial stretching of the fabric bag and escaping gas through vents. Therefore, the performance of the airbag is greatly influenced by the mechanical properties of the fabric. Generally, air bags are designed to deploy in a crash that is equivalent to a vehicle crashing into a solid wall at 8 to 14 miles per hour. Air bags most often deploy when a vehicle collides with another vehicle or with a solid object like a tree. There are various types of airbags; frontal, side-impact and curtain airbags. In general, the passenger side airbags are usually larger than the driver airbags (see Fig. 1).

Extensive studies have shown that the airbag deployment in load cases consists of two occupant loading phases: a punch-out effect where the airbag bursts out of its container with the airbag and airbag module cover accelerating towards the occupant, and a second loading phase during which the airbag is taking on its deployed shape and volume (membrane-loading effect). Bankdak et al. (2002) developed an experimental airbag test system to study airbag-occupant interactions during close proximity deployment. The results provided insight for simulating the effect of inflation energy and mass flow on target response [13]. Bedard et al. (2002) found that while left-side (driver-side) impacts accounted for only 13.5 % of all crashes, the fatality rate among these crashes was 68.3 % in comparison to front impact (48.3 %), right-side impact (31.3 %), and rear impact (38.4 %). These studies underscore the importance of occupant safety during side impact collisions [14]. In the last years, current market request to reduce the time and cost airbag development. In order to achieve this result, virtual simulations play an important role since

Fig. 1 Frontal and side airbags [12]

they allow to minimize the number of experimental tests [15, 16]. Several simulation models of airbag were established [17]. It's feasible to optimize the parameters of airbag deployment using simulation technology. Experimental and numerical studies have quantified injury risks to close-proximity occupants from deploying side airbags. These studies have focused on the prevention of the most adverse effects of airbag deployment [18]. Other studies have proposed airbag characteristics to minimize particular biomechanical responses [19]. In a more recent study, Marklund and Nilsson (2003) compared deformation patterns with experimental data as well as the computational costs associated with three different airbag deployment simulation methods; they concluded that the SPH method is relatively inexpensive and produces incremental deformation patterns that compare most closely to the experimental results [20]. The process of inflation of an airbag is one of the determining factors in saving lives. The duration from the initial impact of the crash to the full inflation of an airbag is about 40 ms and during this time, the airbag goes from being in a folded state to a fully inflated state, with a high internal pressure. After achieving this state, the airbag begins to deflate, thus providing a nice cushion for the body impacting it. Ideally the person in the crash should come into contact with the airbag at this time. This study is therefore mainly focused on the numerical simulation of the passenger airbag, without taking into account the effects of a folding of airbag and the crash dummy, let's understand the one first. The main aim of this study is evaluate the performance of deploying of passenger side airbag using Finite Element Methods (FEM).

2 Finite Element Model of Airbag Simulation

2.1 Theoretical Background

Numerical simulations of airbags use very complex and techniques such as, an orthotropic model to identify the mechanical behaviors during the airbag inflation, the fluid mechanics (gas flow) to describe the inflator gas flow (pressure gradient), and improve the representation of the pressures within the airbag. To model the airbag as an orthotropic model, three material constants have to be provided. Assuming a plane stress condition, the material constitutive equations are given by [21]:

$$\begin{Bmatrix} \sigma_1 \\ \sigma_2 \\ \tau_{12} \end{Bmatrix} = \begin{pmatrix} Q_{11} & Q_{21} & 0 \\ Q_{12} & Q_{22} & 0 \\ 0 & 0 & Q_{66} \end{pmatrix} . \begin{Bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \gamma_{12} \end{Bmatrix} \quad (1)$$

where σ is the normal stress and τ is the shear stress, the subscript refers to the principal material directions, i.e. the fill and warp directions. Also ε and γ are the strain components. The material elastic constants Q_{ij} are given by the following equations:

$$\begin{aligned} Q_{11} &= \frac{E_1}{1-v_{12}.v_{21}}, \quad Q_{12} = \frac{v_{12}.E_2}{1-v_{12}.v_{21}} = \frac{v_{21}.E_2}{1-v_{12}.v_{21}} \\ Q_{22} &= \frac{E_2}{1-v_{12}.v_{21}}, \quad Q_{66} = G_{12} \end{aligned} \tag{2}$$

where E_1 and E_2 are the Young's modulus in the fill and wrap directions and G_{12} is the shear modulus of the fabric material. ν_{ij} is the Poisson ratio of the material.

The gas exerts a pressure load on the airbag causing it to expand. This expansion puts the airbag under tensile stress lowering the expansion rate. In this study, heat conduction and heat transfer is not taken into account. In the deployment of an airbag an inflator supplies high velocity gas into an airbag causing it to expand rapidly. The gas inside the airbag is assumed to be ideal, to be of constant entropy and to satisfy the equation of state:

$$p = (\gamma - 1) \cdot \rho \cdot e \tag{3}$$

Here p, ρ and e are respectively the pressure, density, specific internal energy and γ is the ratio of the heat capacities of the gas. The gas flow is described by the conservation laws for mass, momentum and energy that read:

$$\begin{aligned} &\frac{d}{dt}\int_V \rho dV + \int_A \rho(u.n)dA = 0 \Rightarrow \frac{d}{dt}\int_V \rho u_i dV + \int_A \rho u_i(u.n)dA \\ &= -\int_A p n_i dA \\ &\Rightarrow \frac{d}{dt}\int_V \rho e dV + \int_A \rho e(u.n)dA = -\int_A u_i p n_i dA \end{aligned}$$

Here V is a volume; A is the boundary of this volume, n is the normal vector along the surface A and u denotes the velocity vector in the volume. Applying Bernoulli's equation in the case of an ideal gas with constant entropy gives:

$$\frac{1}{2}u^2 + \frac{\gamma}{\gamma-1}\frac{p}{\rho} = \frac{1}{2}u_{ex}^2 + \frac{\gamma}{\gamma-1}\frac{p_{ex}}{\rho_{ex}} \tag{5}$$

Here the subscript ex denotes quantities at the throat of the tube. Furthermore u, p and ρ denote the quantities inside that part of the tube that is supplying mass. This gives

$$u_{ex}^2 = \frac{2\gamma}{\gamma-1}\left(\frac{p}{\rho} - \frac{p_{ex}}{\rho_{ex}}\right) \tag{6}$$

2.2 Materials and Boundary Conditions

The airbag system mainly consists of three parts: the airbag itself, the inflator unit and the crash sensor or diagnostic unit. Thus, to study the behavior of the airbag using FE simulations, we need to have an FE model of the airbag in the folded position. A FE model of the airbag was used to simulate the test condition as shown in Fig. 2. LS-DYNA® material model FABRIC (MAT_34) is used to simulate the airbag material. It is a variation of the layered orthotropic material model [22]. Additionally in the LS-DYNA® material model, fabric leakage can be accounted for. However, for this CAB material, the leakage is almost negligible and therefore no leakage is specified. The mechanical properties can be determined from the physical test. Typical material properties for airbag fabrics are taken as given in Chawla et al. (2004). These properties are used to simulate inflation process of airbag (see Table 1). The car dashboard is modeled as the rectangular thin plate using a MAT_RIGID material and the degrees of freedom are constrained in all the directions. It is assigned the similar properties of thermoplastic polymer for contact purposes. The porosity of the fabric is assumed zero. The Nitrogen gas is taken for inflating the airbag. Properties of nitrogen gas and initial bag conditions are shown in Table 2. The example on which we perform the study is a typical passenger side airbag. The geometric details have been measured from a commercially available airbag. The initial state of the airbag is a closed rectangular whose sides are to be finished to 482*635 mm^2, and is shown in Fig. 2.

Airbag mesh is generated in Ansys® Finite Element Software. It is consists of 2832 elements and 2875 nodes in the airbag mesh. Quadrilateral elements are used for airbag mesh. The airbag mesh is exported to Lsdyna® software (see Fig. 3). All the simulations of the airbag deployment mesh are done in this software. Contact

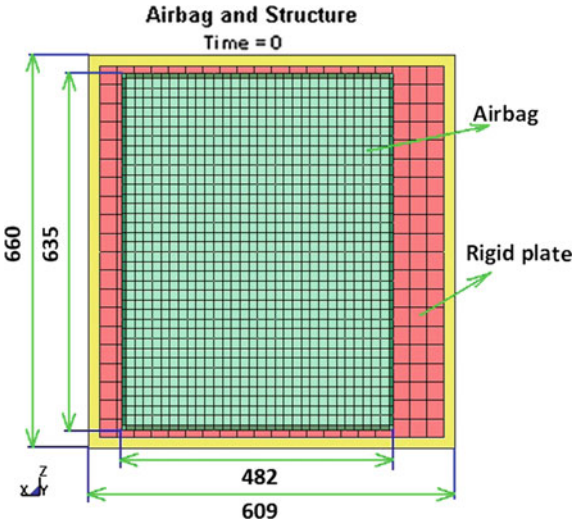


Fig. 2 The initial airbag geometry in the form of a rectangular

Table 1 Material properties of airbag and rigid plate considered in the FE simulations [23]

Materials	Airbag	Rigid plate
Density of fabric (g/cm^3)	1.02	7.84
Young's modulus (GPa)	13.8	206
Poisson's ratio	0.35	0.30
Shear modulus (GPa)	6.9	–

Table 2 Initial values used for FE simulation of the swelling of passenger airbag [24]

Pressure (Pa)	10^4
Temperature (°C)	25
Universal gas constant (kg/kmol K)	8.314
Initial pressure (Pa)	1.01×10^4
Molecular weight (kg/mol)	0.02802
Added initial volume (m^3)	3.33×10^{-4}

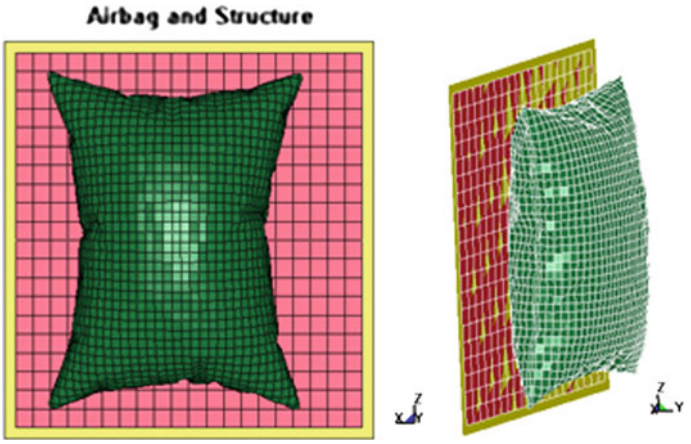


Fig. 3 Top view and isometric view of inflated airbag and rigid plate meshes

type 37 of Lsdyna® software is used for defining the contact between airbag mesh and rigid plates. This contact type is between node and surface. Airbag mesh is treated as nodes and rigid plates are taken as surface. In the simulation, the pressure generated by the gas is then uniformly applied to the internal surface of the airbag fabric.

In the simulation there will be an airbag surface and gas that fills up the inside of this surface. The airbag surface consists of flexible membrane elements that deform under tensile stresses and cannot carry compressive loads. The airbag control volume in LS-DYNA® is the airbag control volume in LS-DYNA® is modeled as an AIRBAG_SIMPLE_AIRBAG_MODEL to simulate the air test condition. A baseline model of the airbag is run using AIRBAG_LOAD_CURVE option. This gives an estimate of the volume of the airbag.

3 Numerical Results and Discussions

Figure 4 shows the different steps of the deploying airbags test simulation at different time points. Due to the fact that only 11 ms are simulated, the airbag is fully inflated as can be seen in Fig. 4. From the Figures, it can be seen that the airbag module cover opens fully, as close to reality. The increasing of volume is coming mainly from the gas flow.

Figure 5 shows the displacements result during the airbag deployment versus time values. Initially, the both surfaces of the airbag are close and parallel, then as the maximum displacement estimated is the same one with low values recorded near the corners of the structure. Then there are concentrated little by little towards the center of the airbag. The final results of the swelling contours are characterized by a symmetric displacement. The final y-y effective strain contours computed are plotted in Fig. 6. As shown this figure, a state of deployment and swelling of passenger airbag with highly non-uniform deformation is detected around the bag. In the present analysis, the localized elastic strains reach values far below the rupture strain that can be measured in the tensile test [25]. Nevertheless, according to the global material response, the constitutive model is found to be still valid at such high deformation levels. Moreover, the development of plaice can clearly be seen during the swelling of passenger airbag.

The airbag result displaces and average pressure distribution inside the airbag for four different thicknesses, 100, 150, 200 and 250 µm is depicted in Fig. 7. They represent the end of swelling of passenger airbag, the final geometry of the bag and the contour of pressure of the gas inside the bag. We can observe how not only the development of plaice of airbag, but also the distribution of pressure is quite different from one case to another. In particular, we can see how the configuration 4

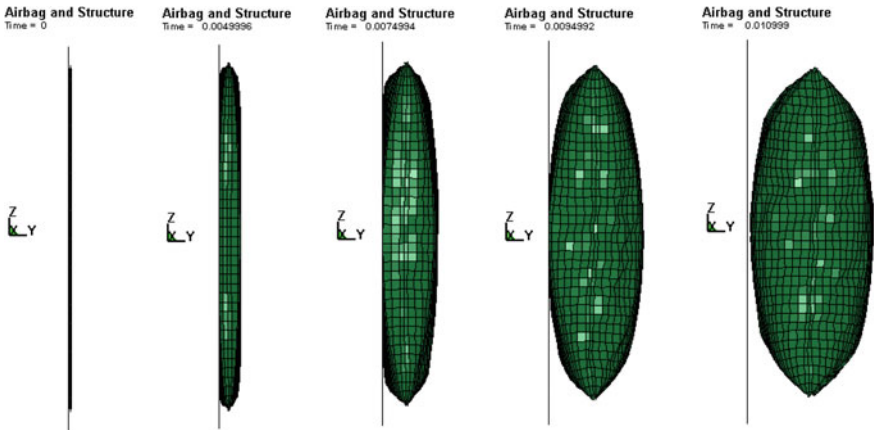


Fig. 4 Airbag deployment at t = 11 ms

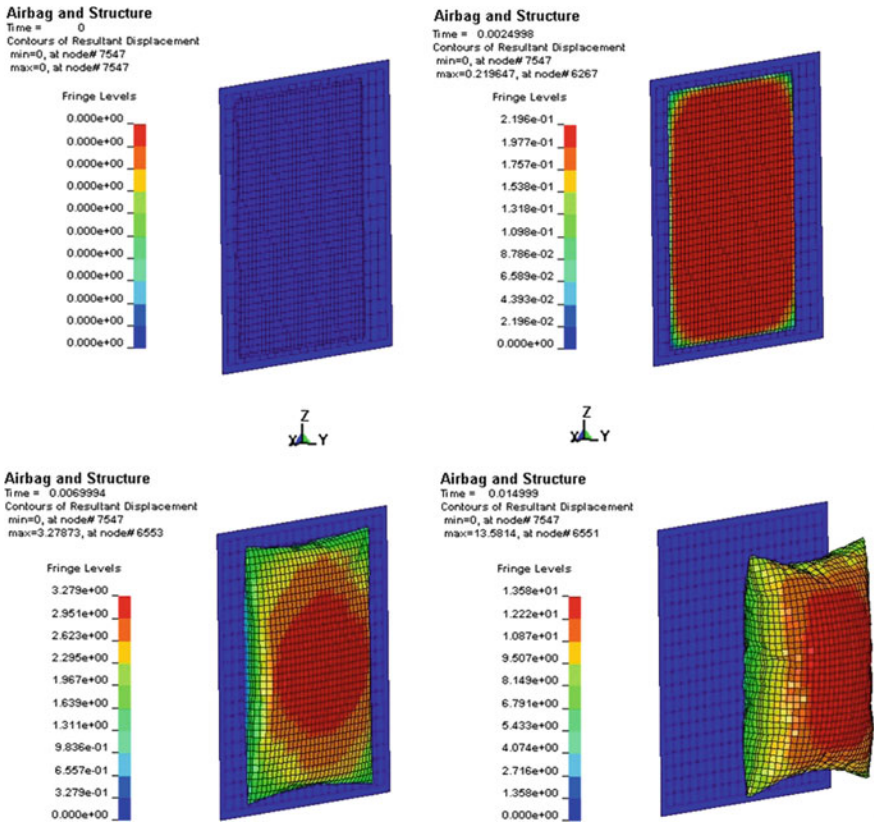


Fig. 5 Airbag result displace

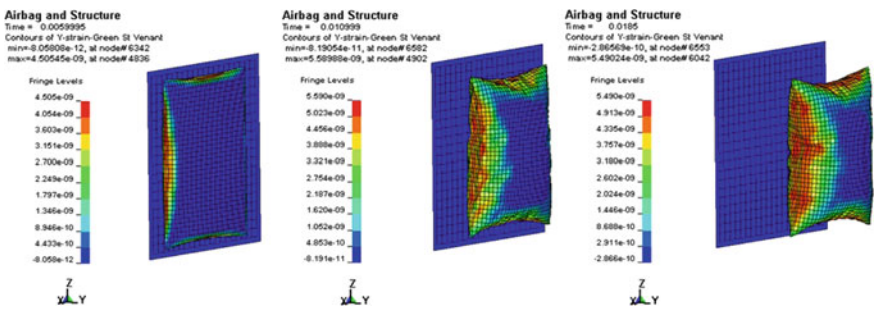


Fig. 6 Airbag result of strain y-y

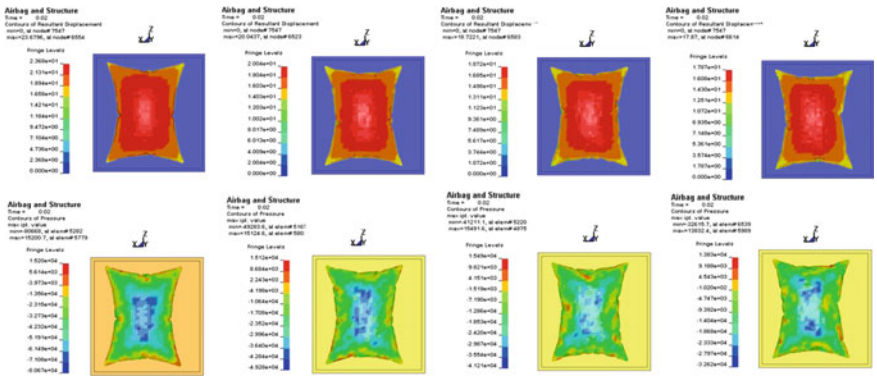


Fig. 7 Airbag result displaces and pressure using different thickness

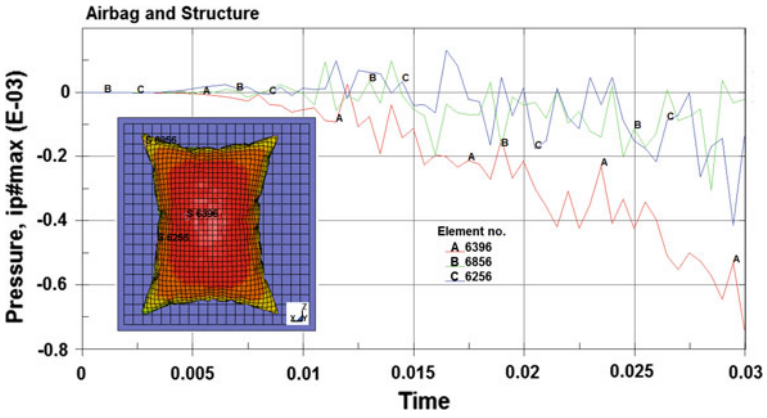


Fig. 8 Pressure evolution versus time

deploys much less aggressively than, for instance, the configuration 1. In this case, there is a quick evolution of the volume and the pressure at the same time [26].

The resulting pressure curves versus time of swelling airbag are plotted in Fig. 8 where the computed responses of different critical zones are apparent. As expected, during the swelling airbag phase, the maximum pressure values are saved at middle of the structure. Moreover, we have also seen after 30 ms, the ratio between the maximum pressure differences is around 28 %. This difference is significantly dependent of the geometric form of airbag [26]. Figure 9 shows the energy total of the system vs. time history during the deploying of the airbag. Maximum value observed at time 30 ms. They increase significantly with the gas injection pressure in bag. The total energy during the swelling airbag can be used to absorbed the collision during crash, by the cushioning effect provided by the airbag, and by the deflation of the inflated airbag, which occurs due to the holes provided in the airbag fabric.

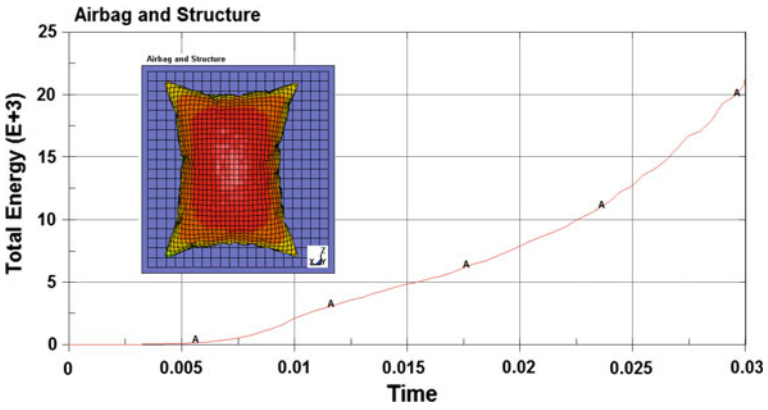


Fig. 9 Total energy versus time during the swelling airbag

4 Conclusions

The study investigated into the effect of airbag deployment in normal scenario without taking into account in this work of the presence of the occupants of various masses and statures. The results first has allowed us to shown that airbag has been modelled correctly with a proper filling of the gas flow. The numerical results shown that the method presented in this paper is also a promising method for simulating airbags but nevertheless some improvements were needed in the modelling of the cover (including the cover thicknesses and mesh boundary conditions) and the tear seam (including the tear seam geometry and the failure method). Besides that, physical tests will also conducted and simulated to assess the different scenarios. Further testing with real life airbags and comparison with experiments is required. It is also very important to model proper folding for airbag mesh for studying contact interaction of out-of-position users with an inflating airbag. These will be carried out in the near future.

References

1. Zhang, H., et al.: CAE-based side curtain airbag design. In: SAE 2004-01-0841, SAE World Congress, 8–11 March 2004, Detroit, Michigan, 8–11 March (2004)
2. Soongu, H.: A study on the modeling technique of airbag cushion fabric. In: SAE 2003-01-0512, SAE World Congress, 3–6 March 2003, Detroit, Michigan, 3–6 March (2003)
3. Canaple, B., et al.: Impact model development for the reconstruction of current motorcycle accidents. Int. J. Crash **7**, 307–320 (2002)
4. Freesmeier, J.J., Butler, P.B.: Analysis of a hybrid dual-combustion-chamber solid propellant gas generator. J. Propuls. Power **15**, 552–561 (1999)

5. Schmitt, R.G., Butler, P., Freesmeier, J.: Performance and CO production of a non-azide airbag propellant in a pre-pressurized gas generator. Combust. Sci. Technol. **122**, 305–350 (1997)
6. Gabauer, D.J., Gabler, H.C.: The effects of airbags and seatbelts on occupant injury in longitudinal barrier crashes. J. Saf. Res. **41**, 9–15 (2010)
7. Crandall, C.S., Olson, L., Sklar, D.P.: Mortality reduction with air bag and seat belt use in head-on passenger car collisions. Am. J. Epidemiol. **153**, 219–224 (2001)
8. Teru, I., Ishikawa, T.: The effect of occupant protection by controlling airbag and seatbelt. In: Proceedings of the 18th International Technical Conference on the Enhanced Safety of Vehicles. NHTSA, Nagoya, Japan (2003)
9. Braver, E.R., Kyrychenko, S.Y.: Efficacy of side airbags in reducing driver deaths in driver-side collisions. Am. J. Epidemiol. **159**, 556–564 (2004)
10. Teng, T.L., Chang, K.C., Wu, C.H.: Development and validation of side-impact crash and sled testing finite-element models. Veh. Syst. Dyn. **45**, 925–937 (2007)
11. Yoganandan, N., Pintar, F., Zhang, J., Gennarelli, T.A.: Lateral impact injuries with side airbag deployments—a descriptive study. Accid. Anal. Prev. **39**, 22–27 (2007)
12. Lim, J.-H., Park, J., Yun, Y.-W., Jeong, S., Park, G.: Design of an airbag system of a mid-sized automobile for pedestrian protection. J. Automob. Eng. 21–33 (2014)
13. Bankdak, F.A., Chan, P.C., Lu, Z.: An experimental airbag test system for the study of airbag deployment loads. Int. J. Crashworthiness **7**, 129–161 (2002)
14. Bedard, M., Guyatt, G.H., Stones, M.J., Hirdes, J.P.: The independent contribution of driver, crash and vehicle characteristics to driver fatalities. Accid. Anal. Prev. **34**, 717–727 (2002)
15. Pei, J., Yuan, S.Q., Yuan, J.P.: Numerical analysis of periodic flow unsteadiness in a single-blade centrifugal pump. Sci. China Technol. Sci. **56**, 212–221 (2013)
16. Cao, Y.H., Wu, Z.L., Huang, J.S.: Numerical simulation of aerodynamic interactions among helicopter rotor, fuselage, engine and body of revolution. Sci. China Technol. Sci. **57**, 1206–1218 (2014)
17. Wang, Y.E., Yang, C.X., Peng, K.: Airbag cushion process simulation for cargo airdrop system. J. Syst. Simul. **19**, 3176–3179 (2007)
18. Duma, S.M., Boggess, B.M., Crandall, J.R., Hurwitz, S.R., Seki, K., Aoki, T.: Upper extremity interaction with a deploying side airbag: a characterization of elbow joint loading. Accid. Anal. Prev. **35**, 417–425 (2003)
19. Haland, Y., Pipkorn, B.: A parametric study of a side airbag system to meet deflection based criteria. J. Biomech. Eng. **118**, 412–419 (1996)
20. Marklund, P.O., Nilsson, L.: Optimization of airbag inflation parameters for the minimization of out of position occupant injury. Comput. Mech. **31**, 496–504 (2003)
21. Jones R.M.: Mechanics of Composite Materials. Hemisphere Publishing Corporation (1975)
22. LS-DYNA3D User's Manual version 970
23. Chawla, A., Bhosale, P.V., Mukherjee, S.: Modeling of folding of passenger side airbag mesh. SAE Int. 1–8 (2004)
24. Deery, H., Morris, A.P., Fildes, B., Newstead, S.: Airbag technology in Australian passenger cars: preliminary results form real world crash investigations. J. Crash Prev. Inj. Control **1**, 121–128 (1199)
25. Wu, W.T., Hsieh, W., Huang, C.H., Wang, C.H.: Theoretical simulation of combustion and inflation processed of two-stage airbag inflators. Combust. Sci. Technol. **117**, 383–412 (2005)
26. Blundell, M., Mahangare, M.: Investigation of an advanced driver airbag out-of-position injury prediction with Madymo gas flow simulations. In: Madymo User Meeting (2006)

Critical Infrastructure Protection—How to Assess the Protection Efficiency

Andrzej Bialas

Abstract The paper presents a concept how to assess the effectiveness of critical infrastructure protection systems. At the beginning the main issues related to critical infrastructure protection are discussed, like resilience, interdependencies, dire phenomena caused by them, and risk management. Next, the state of the art is reviewed. It embraces frameworks, methods and tools related to infrastructures protection, especially risk management. The paper extends the researches of the EU CIRAS project beyond the risk management issue, proposing a method to assess the effectiveness of countermeasures selected for implementation. The concept is based on supplementing the risk management framework by incident management, incident statistics and effectiveness indicators presenting relevant parameters for decisions makers. To implement this concept, the CIRAS risk management software platform should be extended. Main categories of statistics and indicators dedicated for critical infrastructures are proposed. In the conclusion the concept is summarized and future works related to its validation is specified.

Keywords Critical infrastructure · Resilience · Risk management · Interdependencies · Incident management · Effectiveness indicators

1 Introduction

The paper concerns critical infrastructures (CIs) protection. CIs are large-scale infrastructures whose degradation, disruption or destruction would have a serious impact on health, safety, security or well-being of citizens or effective functioning of governments and/or economies. The examples of CIs are infrastructures providing services in the energy-, oil-, gas-, finance-, transport-, telecommunications- and health sectors.

A. Bialas (✉)
Institute of Innovative Technologies EMAG, Leopolda 31,
40-189 Katowice, Poland
e-mail: andrzej.bialas@ibemag.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_3

The processes which provide these services are based on different assets: technological, IT hardware, software, environmental, personal, and organizational. CI is identified as a very complex socio-technical system, sometimes called a system of systems. The system of systems (SoS) consists of multiple, heterogeneous, distributed, occasionally independently operating systems embedded in networks on multiple levels, which evolve over time [1].

The processes providing services are interrelated with other processes across different economy sectors. Special terms—dependencies and interdependencies—were introduced in the CI domain. Dependency defines a unidirectional relationship between two infrastructures, e.g. the telecommunications CI is dependent on the energy CI, because to function properly telecommunications needs energy. Interdependencies are much more complicated. They represent a set of different mutual and bidirectional relations existing in the set of co-operating infrastructures. The strength of coupling between particular CIs may vary.

Critical infrastructures are crucial for the functioning of a society and economy. The CIs processes may be disturbed, and their assets breached by different threats and hazards, such as: natural disasters and catastrophes, technical disasters and failures, espionage, international crime, physical- and cyber terrorism. This is deep motivation to develop critical infrastructure protection (CIP) programmes on the national or international levels. Creating and implementing these programmes is difficult due to the CIs complexity, existing interdependencies and cross-sectoral relations. Risk management is the key issue to create the CIP systems, because the identified risk is the basic factor during the selection of countermeasures which mitigate risk.

The existing risk management methods and tools, used to protect CIs, were mainly developed for business or public organizations, not especially for CIs—this is a conclusion from their review presented in Sect. 2. Critical infrastructures are much more complex than business organizations. Particular CIs do not embrace single organizations (CI operators) but their groups, and, moreover, they are connected to each other by a huge number of relationships, sometimes even unknown. Operational CIs are included in more general structures to be considered on the national or even on international levels.

The paper concerns the European CIRAS[1] project [2] related to "The Prevention, Preparedness and Consequence Management of Terrorism and other Security-related Risks Programme—CIPS". CIRAS is performed by the international consortium, including the author's organization:

- ATOS Spain SA (ATOS),
- Centre for European Security Strategies from Germany (CESS),
- Institute of Innovative Technologies EMAG from Poland (EMAG).

[1]This project has been funded with support from the European Commission. This publication reflects the views only of the author, and the European Commission cannot be held responsible for any use which may be made of the information contained therein (Grant Agreement clause).

The CIRAS method and tool are based on the FP7 ValueSec approach [3]. The countermeasures selected for implementation in critical infrastructures should satisfy the following requirements:

- be able to properly reduce the risk volume to ensure security on an accepted level;
- be cost-effective during implementation and operation and bring benefits for CI stakeholders;
- be free of social, psychological, political, legal, ethical, economical, technical, environmental, and other limitations; these intangible factors are called here "qualitative criteria".

To satisfy these requirements the CIRAS Tool was equipped with dedicated components:

- Risk Reduction Assessment (RRA) which assesses risk before and after the countermeasure implementation,
- Cost-Benefit Assessment (CBA) which assesses cost and benefits factors (monetary, tangible) in the given time horizon after the countermeasure implementation,
- Qualitative Criteria Assessment (QCA) which assesses intangible factors that may occur after the countermeasure implementation.

CIRAS focuses on the countermeasures selection, which is the central issue of risk management. This selection is understood as a single act of the decision maker and continuous, security and safety management aspects are not considered here. CIRAS does not refer to other important issues related to CIP, e.g.: resilience, incident management, effectiveness of the applied countermeasures, and CIP management aspects like: planning, implementing, monitoring, correcting, improving.

This was the motivation to work on an extension of the CIRAS methodology towards the CIP management framework, which would be able to plan, implement, monitor and maintain the CI protection on the assumed level.

The objective of the paper is to propose a method and tool to assess the effectiveness of the CI protection system. This protection is based on the applied countermeasures and on the protection management activities. The protection effectiveness rises when the number of incidents and related losses decreases. In the paper it is assumed that the effectiveness of the CI protection system can be assessed with the use of statistics of incidents and effectiveness indicators. The statistics and indicators are the basis to define correction and improvement actions in the CI protection system. The incident management is important as well. It delivers data to build statistics, to feed indicators, and to allow immediate reaction to security problems.

When the protection decreases below the acceptance level the CI protection system needs corrections and improvements, and when it considerably increases above the acceptance level, it is possible to economize the protection cost.

The results of the author's researches can be used as input to indicate directions of the CIRAS project in the future.

The paper includes the state of the art summary (Sect. 2), the concept of effectiveness assessment for countermeasures (Sect. 3), its implementation on the ready-made software platform (Sect. 4), and conclusions.

2 State of the Art

The review was focused on:

- CI-specific issues, like resilience, interdependencies and related effects,
- risk management methods and tools, especially those used in CIs,
- CIP frameworks and projects.

Critical infrastructure resilience concerns its ability to mitigate the magnitude or duration of hazardous events. The resilient CI is resistant to external and internal disturbances and can function on an acceptable efficiency level in the face of a hazardous event. It means that the CI is able to predict, absorb, react, adapt itself to critical situations, or recover after the disruptive event.

Resilience frameworks [4] are based on best practices and encompass methods and/or tools to perform the system analysis, interdependencies analysis and risk management. Building the CI resilience is a process which includes the following stages:

1. Structural analysis of the CI as a system of systems.
   It is focused on the CIs static model elaboration, i.e. most important nodes, most vulnerable nodes, dependencies and interdependencies (direct, indirect) are identified and expressed by criticality-, vulnerability-, dependency directed graphs or matrices.
2. Dynamic analysis of the CI.
   Based on the static model, the scenarios essential for the CI resilience are analyzed, like: recovering after the given event in the given time, identification of threats impacts, analyses of common failures, the system response to a failure or an incident. The event driven dynamics method is used. The qualitative approach is based on the concurrent event sequence diagrams. When the quantitative approach is applied, simulation tools are used. As a result, a set of the most dangerous risk scenarios is identified.
3. Prioritization of risk scenarios.
   The identified risk scenarios are ordered according to their harmful consequences and prepared to the risk management process in which the detailed analysis is conducted and countermeasures are selected.
4. Risk management.
   The selected and prioritized risk scenarios are analyzed, the risk value assessed, and the right countermeasures selected.

The existing frameworks are focused on the resilience building as a one-time act, not on the resilience maintenance over the time by continuous management activities.

Researches [5, 6] distinguish four basic kinds of interdependencies: physical, cyber, geographical, and logical ones. Because of interdependencies, dire effects of hazardous events propagate across the collaborating infrastructures causing CI-specific phenomena, like: cascading effects, escalating failures, common cause failures [5, 7]. Interdependencies and dependencies are expressed by matrices of relationships or by dependency graphs [4]. The interdependencies analysis is the key issue in building the CI resilience.

The first task of the CIRAS project concerned an exhaustive review of laws, standards, frameworks, methods and tools [8]. This review was based mainly on the following knowledge sources:

- the report [9] of the Institute for the Protection and Security of the Citizen, one of the EC Joint Research Centres (JRC); the report assesses and summarizes 21 existing risk management methodologies/tools on the EU and global level, identifies their gaps and prepares the ground for R&D in this field;
- the book [6]; Appendix C compares the features of about 22 commonly used risk analysis methods;
- the EURACOM report [10] features a desktop study of 11 risk assessment methodologies related to the energy sector;
- the ISO 31010 standard [11] describes about 30 risk assessment methods for different applications;
- the ENISA website [12] gives an inventory of risk management/assessment methods, mostly ICT-focused;
- projects performed on international and national levels (about 20 projects);
- frameworks used by the leading countries in this domain, e.g. [4, 10, 13–16].

From the CI protection perspective the risk management method/tool should: consider interdependencies and phenomena related to them, analyze consequences and causes of the given hazardous event, express the most important data included in the risk register understood here as the managed inventory of hazardous events, be flexible for configuration of different parameters, e.g.: likelihood, probability, frequency, consequences, their categories, scales of measures. The existing methods/tools were developed for single business organizations, not for a set of collaborating organizations, like CIs. Some methods/tools were adapted for the critical infrastructures requirements, especially for the lower level of the CI hierarchy (e.g. operator level). There is lack of risk management method/tools for the higher level (e.g. international level). The reviewed methods [8] do not address the CI specific phenomena in a satisfactory way. There is no method which would consider the cost, benefit, and intangible restrictions with respect to the CIs.

The CIs resilience- or risk management frameworks are defined on a very general level. There are no comprehensive critical infrastructure protection frameworks that would take into account all aspects important for CIs like: resilience,

interdependencies, risk management in all important perspectives (planning, implementing, maintaining, improving). Particular frameworks focus on the selected aspects, e.g., risk management [16], resilience building or interdependencies [4]. The US Dept. of Homeland Security framework [13, 14] represents a comprehensive approach to the risk-based resilience building and maintenance, including feedback loops and iterative steps. One of the risk management activities is to measure effectiveness. Metrics and evaluation procedures are used to measure progress and assess the effectiveness of the efforts to secure and strengthen the resilience of a critical infrastructure. Measures of effectiveness, indicators are broadly used in information security or IT governance standards, like ISO/IEC 27001 [17] or COBIT [18]. None of the frameworks was based on the Deming cycle [19], which is a proven solution to manage and maintain quality, security, business continuity, etc.

3 Critical Infrastructure Protection Framework Effectiveness—Concept of Assessment

The effectiveness of a CI protection system rises when damages caused by incidents and protection costs decrease. The problem is that to decrease damages, the risk should be better reduced and this rises the protection cost too. A trade-off between different factors is needed, and these factors should be identified and monitored. Sampling all these parameters in an aggregated way supports the decision makers involved in the management of critical infrastructures protection systems.

The countermeasures should be properly selected to achieve the security/safety on the accepted level. The countermeasures selection is based on the ability to reduce risk. In the developed CIRAS Tool, the cost-benefits parameters and intangible factors are taken into account too. CIRAS does not go beyond the countermeasure selection. This is considered as a one-time act. There are no operations to maintain the achieved security over time—this issue is out of this project scope.

Please note that incidents within a protected system, like a CI, are still possible, due to a few factors:

- during risk assessment a mistake or an inaccuracy may occur; please note that methods and tools used in the CI domain are rather simple, while the domain is complex,
- new threats or vulnerabilities appear, previously unknown or omitted,
- changes in CI occur and for this reason the protection system does not match the situation after changes,
- risk is ignored (this option is not recommended).

For this reason the entire protection system which embraces a set of different but coherent countermeasures should be monitored, managed and corrected to minimize incidents and their impacts. The problem is solved in security management

systems [17], IT governance systems [18] and in many other management systems related to business or public organizations, but not for critical infrastructures.

To solve this issue, the critical infrastructure and its protection system should be equipped with management facilities, especially with:

- an incident management system which should be able to identify the incident, to react, to do lessons learnt, and to derive corrections in the protection system,
- different incident assessment facilities and indicators, working like sensors, track parameters relevant to the effectiveness of the whole system and are able to monitor the effectiveness of the protection system.

To assess the effectiveness of the protection system, the number of incidents (materialized risk scenarios) and their consequences in different views should be observed. An incident management system provides these data, however, the data should be processed and presented in an aggregated way, e.g. as on-line statistics or graphs. Incidents observation and on-line reaction are close to the real-time risk management concept.

Indicators can be monitored on line to react immediately when something goes wrong, or can be analyzed in the assumed time horizons, e.g. yearly, to improve the existing protection system.

Statistics and indicators can be useful in periodical risk reassessment (a static approach). Risk prediction can be confronted with the occurred incidents (materialized risks) to elaborate more adequate predictions (and countermeasures) for the future. Information about the countermeasures past and future costs is also useful. This allows to optimize protection costs which influence the protection system effectiveness.

It is assumed that the effectiveness assessment of the CI protection system is based on different sources of information needed for decision makers and other people responsible for the CI protection. The basic sources of information showing this effectiveness are:

- incident statistics,
- indicators,
- the cost of the protection system.

The assessment results are the foundation of correction actions and continual improvement of the protection system.

4 Protection Framework Effectiveness Assessment—Implementation

To make a feasibility study of the above effectiveness assessment concept, the author proposes to use the OSCAD software platform [20]—the same as the one used in the RRA component in the CIRAS Tool.

The paper is continuation of researches focused on the OSCAD application in the CIRAS project. Current researches are focused on risk management. The paper [21] presents the requirement for the CI risk manager, [22] presents the experimentation tool based on these requirements, and [23] is focused on the validation experiment dealing with the risk management in the collaborating infrastructures. As a result the OSCAD-based RRA component is proposed which is currently adapted to and embedded into the CIRAS Tool. The experimentation risk manager is called here OSCAD-CIRAS. The paper proposes to go beyond the risk management research and to extend OSCAD-CIRAS by a new functionality to measure the effectiveness of the protection system.

The OSCAD software was originally developed as an integrated system to support:

- business continuity management according to ISO 22301 [24], to identify and mitigate different disturbances of business processes,
- information security management according to ISO/IEC 27001 [17], to identify and mitigate breaches of information assets.

OSCAD was developed for business or public organizations and it has three main functionalities:

- to perform risk management (preparedness),
- to manage incidents (reaction, recovery),
- to ensure continual improvement of the security-related management processes.

4.1 Information from Incident Management System Useful in the Effectiveness Assessment

OSCAD is equipped with a complex event/incident management system. Events are reported and evaluated. Some of them, bringing higher damages, are classified as incidents. The event circumstances and causes are specified. OSCAD helps to react when incidents occur. In this case, for serious incidents, ready-to-use emergency plans can be activated. Closed incidents are assessed again (lessons learnt) to plan corrective actions within the protection system. The incident reports are sampled in the database to produce statistics.

Figure 1 shows two simple statistics issued: events by weekdays and kinds of events (police interventions, non-classified, failures, other events). The example deals with the Railway Safety Management System [25]. Similar statistics will be created for CIs, however, to obtain reliable statistics, the data should be sampled over long time.

For critical infrastructures the examples of statistics can be created around the following:

- number of incidents of the given severity and category,
- losses from incidents of the given severity and category,

Fig. 1 Events by weekdays and kinds of events [25]. *Source* OSCAD. Prepared by the author, 2014

- number of unresolved incidents,
- total number of incidents,
- total losses caused by incidents.

The severity related to the damages caused by the event/incident can be defined similarly to the consequences severity used in the risk assessment. For example, events can be classified with the use of the enumerative scale [21]: Negligible damage, Minor damage, Major damage, Severe loss, Catastrophic.

Incidents are a subset of events of higher damages: Major damage, Severe loss, Catastrophic.

It is proposed that the categories of risk (threats) [22, 23] should comply with the categories of events/incidents. This way it is possible to compare predicted risks with incidents (materialized risks) and get extra information about the effectiveness of the protection system.

The incident statistics are a source of information to plan corrective actions in the CI protection system.

4.2 *Indicators Expressing the Effectiveness of the Protection System*

OSCAD-CIRAS can be equipped with indicators, including the user defined variables, which sample relevant values. The value of the given effectiveness indicator can be:

- entered manually by the user,
- downloaded automatically, e.g. from telematics applications, from ERP (Enterprise Resource Planning), from SCADA (Supervisory Control And Data Acquisition),
- calculated on the basis of the existing data (incident records, statistics) as aggregated values.

Indicators represent security, safety, reliability, resilience, technical and management issues. OSCAD is equipped with these mechanisms but the adequate indicators should be defined during the planned researches.

The first group of examples concerns the basic indicators or their families:

- average reaction time to an incident of the given severity and category,
- maximal reaction time to an incident of the given severity and category,
- average recovery time per incident category,
- average recovery cost per incident category,
- total recovery cost,
- number of audits, reviews, correction actions in the protection system,
- number of false alarms,
- number of incidents related to dangerous products, e.g. chemical, nuclear,
- number of deaths,
- number of persons seriously injured,
- number of incidents when RTO (Recovery Time Objective) was exceeded,
- number of precursors of incidents, near-failures,
- volume of compensation related to insurance,
- economic efficiency factors, e.g. volume of transported goods, produced energy, provided services per year,
- percentage of the state-of-the art countermeasures (modern, certified, automatic, etc.) in the protection system,
- percentage of CCTV cameras with video analysis with respect to the total number of cameras,
- number of security awareness activities, trainings for employees in the given time horizon.

The second group of indicators is defined to check the CI specific phenomena, e.g. internal and external escalation/cascading, common cause effects:

- number of incidents caused by external critical infrastructures through interdependencies,
- volume of losses due to incidents caused by external critical infrastructures through interdependencies,
- number of incidents invoked in external critical infrastructures through interdependencies,
- volume of losses due to incidents caused by external critical infrastructures through interdependencies,

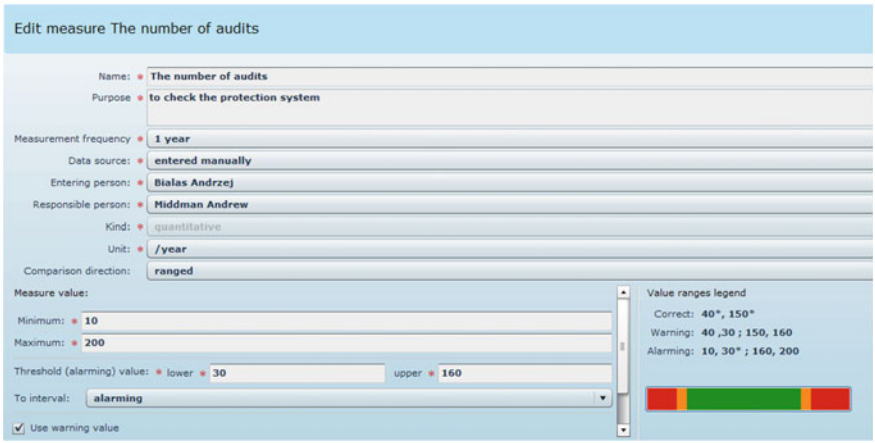


Fig. 2 Effectiveness indicator—an example. *Source* OSCAD-CIRAS. Prepared by the author, 2016

- number of internally escalated incidents,
- number of detected common cause failures.

Figure 2 presents the definition of an indicator in OSCAD-CIRAS which checks the protection system with respect to the yearly planned audits. In the example it is assumed that minimum 10 audits a year should be performed, and the total number of audits cannot exceed 200 per year. To watch the current number of audits the warning and alarm thresholds are defined. Exceeding the given threshold causes automatic generation of a task for the responsible person, and a warning or an alarm respectively. Apart from indicators which monitor the value in the range, there are indicators which show if the value is above (or below) the assumed threshold.

The third group of indicators is defined to check the cost and benefits parameters dealing with the entire CI protection system, like:

- total cost of the protection system per year,
- total investment cost related to the category of countermeasures per year,
- total operational cost related to the category of countermeasures per year,
- total benefits resulting from risk reduction.

These total indicators are calculated based on the parameters values assigned to the particular countermeasure during the OSCAD-CIRAS risk management process. These parameters, like investment cost, operation cost, future benefits, are provided for the CBA component.

The key issue is to define how often the statistics and indicators should be updated and reviewed. This feature is configurable.

It is possible to observe the QCA indicators, but this issue not discussed here. When a given incident is assessed, the QCA factors should be considered, i.e. whether the incident was caused by trespassing of the given qualitative criterion.

5 Conclusions

The paper presents a new concept and implementation of the effectiveness assessment of the critical infrastructure protection system. CI protection is based on the countermeasures which are selected according to the risk value.

The proposed method tries to assess countermeasures effectiveness in practice, observing the behavior of the protected CI. The effectiveness depends on the number of occurred incident, losses caused by them and the cost of applied countermeasures.

For this reason the OSCAD-CIRAS risk manager used in the CIRAS project is extended by an additional functionality related to:

- incident management—to gather data about incidents and to allow immediate reaction to incidents (real-time risk management aspects);
- incident statistics—to present synthetically information about incidents and damages;
- effectiveness indicators—to present more enhanced information related to incidents, their consequences, functioning of the CI and its protection system, protection cost and limitations (provided by the CBA and QCA components of the CIRAS Tool).

The proposed OSCAD-CIRAS extensions are based on the functionality existing in the software which still requires to be customized and configured. The paper shows how to do it. After that the validation experiment is planned. The validation will concern two collaborating and mutually dependent infrastructures: railway transport and energy production.

The proposed concept supports the CI protection system. The countermeasures are not only properly selected, but also monitored and managed. Synthetic information about behavior of the protected CI allows to provide corrections and improvements within the protection system.

References

1. Eusgeld, I., Nan, C., Dietz, S.: "System-of-systems" approach for interdependent critical infrastructures. Reliab. Eng. Syst. Safety **96**, 679–686 (2011)
2. Ciras project, http://cirasproject.eu/. Accessed January 2016
3. ValueSec project, www.valuesec.eu. Accessed January 2016
4. Giannopoulos, G., Filippini, R.: Risk Assessment and Resilience for Critical Infrastructures. Workshop Proceedings, 25–26 April 2012, Joint Research Centre—Institute for the Protection and Security of the Citizen. Ispra, Italy. https://www.google.pl/search?q=Risk+Assessment+and+Resilience+for+Critical+Infrastructures.+Workshop+Proceedings&ie=utf-8&oe=utf-8&gws_rd=cr&ei=s-h6VvXDMof2aJ-EqsgB. Accessed December 2015
5. Rinaldi, S.M., Peerenboom, J.P., Kelly, T.K.: Identifying, Understanding and Analyzing Critical Infrastructure Interdependencies. IEEE Control Systems Magazine. December, pp. 11–25 (2001)

6. Hokstad, P., Utne, I.B., Vatn, J. (eds.): Risk and Interdependencies in Critical Infrastructures: A Guideline for Analysis (Springer Series in Reliability Engineering). Springer, London (2012). DOI:10.1007/978-1-4471-4661-2_2
7. Rausand, M.: Risk Assessment: Theory, Methods, and Applications. Series: Statistics in Practice (Book 86). Wiley (2011)
8. Baginski, J., Bialas, A., Rogowski, D., et al.: D1.1—State of the Art of Methods and Tools, CIRAS Deliverable. Responsible: Institute of Innovative Technologies EMAG, Dissem. level: RE/CO (i.e. available for: beneficiaries, stakeholders, Europ. Commission) (2015)
9. Giannopoulos, G., Filippini, R., Schimmer, M.: Risk Assessment Methodologies for Critical Infrastructure Protection. Part I: A State of the Art. European Union (2012)
10. Deliverable D2.1: Common areas of Risk Assessment Methodologies. Euracom (2007)
11. ISO/IEC 31010:2009—Risk Management—Risk Assessment Techniques
12. ENISA: http://rm-inv.enisa.europa.eu/methods. Accessed December 2015
13. NIPP 2013: Partnering for Critical Infrastructure Security and Resilience. The US Department of Homeland Security (2013). http://www.dhs.gov/sites/default/files/publications/NIPP%202013_Partnering%20for%20Critical%20Infrastructure%20Security%20and%20Resilience_508_0.pdf. Accessed January 2016
14. Supplemental Tool: Executing A Critical Infrastructure Risk Management Approach. U.S. Department of Homeland Security, DHS Risk Lexicon (2010). http://www.dhs.gov/sites/default/files/publications/NIPP%202013%20Supplement_Executing%20a%20CI%20Risk%20Mgmt%20Approach_508.pdf. Accessed January 2016
15. Stapelberg, R.F.: Infrastructure Systems Interdependencies and Risk Informed Decision Making (RIDM): Impact Scenario Analysis of Infrastructure Risks Induced by Natural, Technological and Intentional Hazards, Systemics, Cybernetics and Informatics, vol. 6, number 5 (2013)
16. ISO 31000:2009, Risk management—Principles and guidelines
17. ISO/IEC 27001:2013 Information technology—Security techniques—Information security management systems—Requirements
18. COBIT: http://www.isaca.org/cobit/pages/default.aspx. Accessed January 2016
19. Deming cycle: https://en.wikipedia.org/wiki/PDCA. Accessed January 2016
20. OSCAD project. http://www.oscad.eu/index.php/en/. Accessed January 2016
21. Bialas, A.: Critical infrastructures risk manager—the basic requirements elaboration. In: Zamojski, W., Mazurkiewicz, J., Sugier, J., Walkowiak, T., Kacprzyk, J. (eds.) Theory and Engineering of Complex Systems and Dependability Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, June 29–July 3 2015, Brunów, Poland. Advances in Intelligent Systems and Computing, vol. 365, pp. 11–24. Springer, Cham, Heidelberg, New York, Dordrecht, London (2015). DOI:10.1007/978-3-319-19216-1_2
22. Białas, A.: Experimentation tool for critical infrastructures risk management. In: Proceedings of the 2015 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 775–780 ISBN 978-1-4673-4471-5 (Web). IEEE Catalog Number: CFP1385 N-ART (Web)
23. Białas, A.: Research on critical infrastructures risk management. In: Rostański, M., Pikiewicz, P., Buchwald, P. (eds.) Internet in the Information Society 2015—10th International Conference Proceedings, pp. 93–108. Scientific Publishing University of Dąbrowa Górnicza (2015)
24. ISO 22301:2012 Societal security—Business continuity management systems—Requirements
25. Bialas, A.: Computer support for the railway safety management system—first validation results. In: Zamojski, W., Mazurkiewicz, J., Sugier, J., Walkowiak, T., Kacprzyk, J. (eds.) Proceedings of Ninth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, June 30–July 4, 2014, Brunow, Poland. Advances in Intelligent Systems and Computing, vol. 286, pp. 81–92. Springer Cham, Heidelberg, New York, Dordrecht, London (2014). ISBN 978-3-319-07012-4. DOI:10.1007/978-3-319-07013-1

Selection of Metrics for the Defect Prediction

Ilona Bluemke and Anna Stepień

Abstract The ability to estimate if a module or a class or a method is faulty, or not, is called the defect prediction. Prediction can be used to target the improvement efforts to those modules or classes that need it the most. We investigated the classification process (deciding if an element is faulty or not) in which the set of software metrics and several data mining algorithms were used. We conducted an experiment on ten open source projects. The data concerning defects were extracted from the repository of the control version system. In this study the process of choosing appropriate metrics for the defect prediction is described. In the selection process we use unique approach by random forest.

Keywords Defect prediction · Object metrics

1 Introduction

Testing software systems is a tedious work. The distribution of defects among parts of a software system is not balanced so applying the same testing effort to all parts of a system is not the optimal approach. Therefore, it would be useful to identify fault-prone parts of the system and with such knowledge to concentrate testing effort on these parts. According to Weyuker et al. [1, 2] typically 20 % of modules contain 80 % of defects. Testers with a tool enabling for defect prediction may be able to concentrate on testing only 20 % of system modules and still will find up to 80 % of the software defects. The models and tools which can predict the modules to be faulty, or not, based on certain data, historic or the current ones, can be used to target the improvement efforts to those modules that need it the most.

The defect prediction methods were subjects of many publications e.g. [3–11] also many surveys on this subject can be found e.g. [3, 4, 7, 10]. In the defect prediction models often software or/and product metrics are used. The application

I. Bluemke (✉) · A. Stepień
Institute of Computer Science, Nowowiejska 15/19, 00-665 Warsaw, Poland
e-mail: I.Bluemke@ii.pw.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_4

of metrics to build models can assist to focus quality improvement efforts to modules that are likely to be faulty during operations, thereby cost-effectively utilizing the testing and enhance the resources. There is abundant literature on software and product metrics e.g. [5, 6, 12–21].

We performed an experiment on the defect prediction and some results presented in [22]. In the classification process (deciding if an element is faulty or not) some software metrics were used. In this study we present how the set of metrics was chosen by random forest approach. However there are many papers on finding metrics suitable for the defect prediction e.g. [5, 6, 9–11, 16–20] to our best knowledge none of them is using the random forest approach.

The paper is organized as follows: In Sect. 2 the defect prediction is briefly sketched and our prediction experiment is outlined. This includes also the presentation of investigated projects. The process of metric selection which uses random forests [23, 24] is described in Sect. 3. In Sect. 3.1 the backgrounds of random forests are given and in Sect. 3.2 steps of the selection procedure are listed as well as the values calculated for examined metrics are shown. In Sect. 3.3 our set of metrics is compared with related results. Conclusions are given in Sect. 4.

2 Defect Prediction

The defect prediction has been conducted so far by many researchers [5–11] and many prediction models has been proposed. The defect predictor builds a model based on data from old version of a project (often data from version control system are used) or current system measures.

In a defect predictor using the current system version e.g. [6, 9], different metrics are typically used. These metrics could be calculated at different levels: process, component, files, class, method. According to Catal and Diri [7]:

- the method level metrics are used in 60 % of defect predictors and
- class level in 24 % while
- process level only in 4 %.

Depending on the metric level appropriate code level can be identified as faulty. When method level metrics are used the predicted fault-prone modules are methods, if class level then classes.

In the experiment on the defect prediction we decided to use machine learning methods and metrics. We conducted the experiment with several algorithms and in [22] the results of *k*-NN algorithm and decision trees were presented. In our experiment ten open source Java projects were used. These projects are listed in Table 1. The fault prediction was made for two successive releases of each project with the usage of object metrics and machine learning algorithms. Object metrics were calculated by Ckjm tool [25] which enables to calculate 19 object oriented metrics from Java code.

Table 1 Analyzed projects

No	Project	Reference	Numb. of classes	Numb. of classes containing errors	Percentage of errors (%)
7	Apache Lucene	[28]	624	65	10.56
3	Apache Hadoop	[29]	701	59	8.40
4	Apache Tika	[30]	174	5	2.86
8	MyBatis	[31]	361	31	8.56
1	Apache Commons Lang	[32]	84	30	35.29
6	Hibernate	[33]	2123	174	8.19
5	Jsoup	[34]	45	3	6.52
9	Apache Velocity	[35]	228	5	2.18
2	Elasticsearch	[36]	2464	27	1.09
10	Zookeeper	[37]	246	37	15.04

The information about errors were obtained by specially designed and implemented application—BugMiner [26]. Given a bug pattern (a regular expression describing either a bug identifier e.g. *PROJECT-1234* or a process of fixing e.g. *fixed*, *bugfix*), the BugMiner finds in the repository of the control version system comments associated with error repairs. The *git* version control system was used [27]. BugMiner uses the metrics values calculated by *Ckjm* and generates statistics for the project, its main operation is presented in Fig. 1.

As it can be seen in Table 1 data which are used in the fault prediction process (classes containing errors) constitute small ratio of all data. Such imbalanced distribution makes a problem for the classification process as it can assign all elements to the dominant group and in the classification for the fault prediction the less numerous group is of higher interest. Solution to this problem is resampling: e.g. over-sampling the minority class.

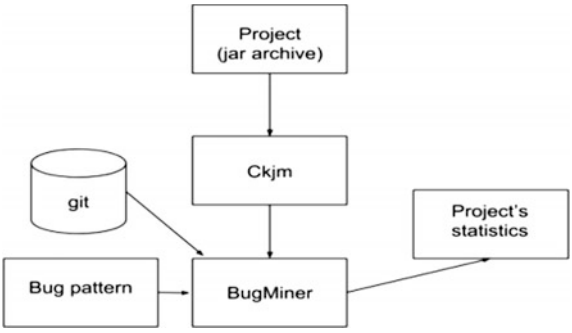


Fig. 1 Operation of the BugMiner

3 Metrics Selection

In the experiment on defect prediction we used the Ckjm [25] tool able to calculate nineteen metrics. Among them are widely known Chidamber Kemerer metrics (CK metrics) [13], QMOOD metrics proposed by: Bansiya and Davis [12], Martin [14], McCabe [16] and some Henderson-Sellers metrics [15].

We decided to use ten metrics and to choose the best ones we performed an analysis by random forests [23, 24] for all projects.

3.1 Random Forests

Random forest proposed by Breiman in 2001 [23, 24] has many classification trees. It is a combination of tree predictors such that each tree depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest.

In the original paper on random forests [23], it was shown that the forest error rate depends on two factors:

- The correlation between any two trees in the forest, increasing the correlation increases the forest error rate.
- The strength of each individual tree in the forest. A tree with a low error rate is a strong classifier. Increasing the strength of the individual trees decreases the forest error rate.

Random forest has many advantages. It runs efficiently on large data base, can handle thousands of input variables, is accurate, gives estimates of what variables are important in the classification. It also has methods for balancing error in class population unbalanced data sets.

When the training set for the current tree is drawn by sampling with replacement, about one-third of the cases are left out of the sample. This OOB (out-of-bag) data is used to get a running unbiased estimate of the classification error as trees are added to the forest. It is also used to get estimates of variable importance.

After each tree is built, all of the data are run down the tree, and proximities are computed for each pair of cases. If two cases occupy the same terminal node, their proximity is increased by one. At the end of the run, the proximities are normalized by dividing by the number of trees.

In random forests, there is no need for cross-validation or a separate test set to get an unbiased estimate of the test set error. It is estimated internally, during the run. Each tree is constructed using a different bootstrap sample from the original data. About one-third of the cases are left out of the bootstrap sample and not used in the construction of the *k*th tree. Each case left out in the construction of the *k*th tree is put down the *k*th tree to get a classification. In this way, a test set classification is obtained for each case in about one-third of the trees. At the end of the run,

take j to be the class that got most of the votes every time case n was OOB. The proportion of times that j is not equal to the true class of n averaged over all cases is the OOB error estimate.

As features used in the splitting trees we used two attributes:

1. mean decrease in accuracy and
2. mean decrease of gini index (probability that randomly chosen two samples are in the same class).

The mean decrease in accuracy (MDA) is calculated as follows. For every tree belonging to the random forest the membership of an "out of bag"—OOB sample to a class is calculated. Next the number of correctly assigned to a class samples is calculated. To define the usefulness of a variable x_i, its values are permutated in the previous selected sample (OOB). For the modified values the process of finding the membership to a class is repeated. The number of correctly assigned samples is subtracted from the same number but obtained for the original sample. The result of the subtraction is divided by the cardinality of OOB sample. The permutation is repeated for all trees in the random forest. The mean decrease in accuracy is the mean of all values obtained for all trees. Variables with high values of decrease in accuracy are significant in the prediction process.

The Mean Decrease Gini (MDG) is a measure of homogeneity of nodes and leaves in the random forest. Every time a split of a node is made on variable x_i the gini impurity criterion for the two descendent nodes is calculated. These values are less than the one for the parent node. Adding up the gini decreases for each individual variable over all trees in the forest gives variable importance. High values shows the importance of this variable in the classification process.

3.2 Metric Analysis

The Ckjm [25] tool calculates nineteens object metrics. We examined the significance of these metrics using the mean decrease in accuracy (MDA) and the mean decrease gini (MDG) for all projects used in the defect prediction experiment. Metric which was "most important" obtained the maximal number of points—nineteen, the "less important" only one point. The metric obtained the number of points accordingly to its "importance" in the classification process. In Table 2 the importance of metrics for the projects from Table 1 calculated for the mean decrease accuracy (MDA) is shown. In parenthesis the number of points assigned to the metrics is also given. In Table 3 similar values are given for the mean decrease gini. The values in tables are rounded to the first digit after the point to adjust the table to the width of the page. It can be seen that some values are equal and the number of points assigned to the metrics are not, the differences were on the second, the third position after the point.

Table 2 Importance of metrics based on MDA (projects numbered as in Table 1)

Metric	Project (numbered as in Table 1)									
	1	2	3	4	5	6	7	8	9	10
wmc	6.4(12)	15.7(11)	21.3(16)	7.9(17)	2.9(19)	24.5(12)	22.3(16)	18.2(18)	9.2(18)	12(13)
dit	1.7(4)	5.2(5)	3.1(5)	3.7(7)	−0.2(9)	7.8(5)	7.0(5)	6.3(7)	0.7(6)	3.5(3)
noc	5.2(11)	−0.3(1)	9.3(8)	0.0(2)	−2.4(1)	1.6(2)	2.4(2)	3.1(5)	−1.0(3)	−4.5(1)
cbo	4.7(10)	16.6(15)	12.3(9)	5.8(13)	−0.1(11)	29.6(18)	30.8(19)	11.9(11)	4.5(7)	9.2(8)
rfc	4.6(9)	16.0(12)	27.6(18)	7.3(15)	2.0(17)	28.8(15)	19.9(12)	16.5(16)	7.7(16)	24.5(19)
lcom	6.7(14)	19.6(18)	13.6(10)	8.1(18)	1.2(14)	21.9(11)	19.(11)	17.8(17)	5.6(12)	6.1(7)
ca	0.9(3)	11.(10)	6.8(6)	1.4(3)	−0.6(8)	21.6(10)	17.5(10)	13.8(14)	5.1(9)	10.5(12)
ce	2.2(5)	16.(14)	14.9(12)	4.6(10)	1.3(15)	28.9(16)	24.3(17)	20.4(19)	5.2(10)	16.3(15)
npm	9.5(16)	16.(13)	17.2(13)	7.4(16)	2.3(18)	31.2(19)	21.4(14)	11.9(10)	11.9(19)	9.6(10)
lcom3	2.7(7)	18.1(16)	19.3(14)	2.4(4)	0.4(12)	17.2(9)	17.0(9)	7.8(9)	7.1(15)	17.7(18)
loc	16.8(18)	20.6(19)	31.7(19)	16.2(19)	−0.2(10)	29.4(17)	25.9(18)	14.6(15)	7.8(17)	16.9(16)
dam	6.7(13)	10.4(9)	14.3(11)	3.8(8)	1.4(16)	16.4(7)	16.3(8)	5.2(6)	5.3(11)	9.4(9)
moa	0.2(1)	8.9(7)	−0.5(1)	3.3(6)	−1.7(5)	16.8(8)	7.8(6)	6.8(8)	6.7(13)	17.3(17)
mfa	3.9(8)	5.1(4)	2.1(4)	5.5(12)	0.5(13)	2.6(3)	2.7(3)	−0.5(1)	0.0(1)	4.3(5)
cam	7.1(15)	10.1(8)	20.6(15)	6.9(14)	−1.4(6)	25.5(13)	20.7(13)	13.7(13)	6.9(14)	12.8(14)
ic	0.7(2)	3.3(3)	0.5(3)	4.3(9)	−1.8(3)	2.6(4)	2.2(1)	0.9(3)	0.0(4)	4.4(6)
cbm	2.4(6)	6.1(6)	0.0(2)	2.7(5)	−2.(2)	1.1(1)	2.8(4)	2.7(4)	−1.(2)	3.6(4)
amc	19.5(19)	18.4(17)	22.5(17)	5.3(11)	−0.9(7)	25.8(14)	22.1(15)	12.7(12)	4.6(8)	10.3(11)
avg_cc	14.3(17)	1.5(2)	7.8(7)	−0.9(1)	1.9(16)	8.3(6)	11.(7)	1.9(3)	−2.7(1)	2.8(2)

Table 3 Importance of metrics based on MDG (projects numbered as in Table 1)

Metric	Project (numbered as in Table 1)									
	1	2	3	4	5	6	7	8	9	10
wmc	2.3(12)	2.8(11)	10.3(18)	0.3(9)	0.5(17)	19.1(11)	7.8(12)	3.9(13)	0.7(13)	3.2(10)
dit	0.5(3)	0.6(5)	1.2(4)	0.1(3)	0.1(4)	3.5(4)	1.6(4)	0.5(4)	0.1(3)	0.6(5)
noc	0.5(4)	0.6(4)	2.2(5)	0.0(1)	0.2(6)	6.4(5)	1.7(5)	0.8(5)	0.0(4)	0.4(3)
cbo	1.7(10)	4.4(16)	6.8(10)	0.7(16)	0.6(19)	23.8(14)	12.3(19)	4.4(16)	0.5(9)	3.8(13)
rfc	2.6(14)	4.7(17)	15.4(19)	0.5(14)	0.4(12)	33.4(18)	9.(14)	5.9(18)	0.8(16)	9.7(19)
lcom	2.4(13)	3.5(13)	8.3(14)	0.4(11)	0.3(10)	22.3(13)	6.4(9)	4.1(15)	0.7(14)	2.3(8)
ca	1.(7)	2.5(10)	4.5(9)	0.6(15)	0.4(13)	18.0(9)	6.7(10)	2.6(9)	0.3(6)	3.6(12)
ce	1.3(9)	4.3(15)	8.7(15)	0.5(13)	0.5(14)	27.8(16)	9.1(15)	10.(19)	0.5(10)	5.8(16)
npm	2.6(15)	2.3(9)	7.4(12)	0.3(8)	0.2(8)	19.2(12)	7.3(11)	2.9(10)	0.9(18)	2.3(9)
lcom3	2.1(11)	3.6(14)	6.9(11)	0.2(5)	0.2(7)	18.1(10)	8.2(13)	3.1(11)	0.6(12)	4.4(14)
loc	5.2(18)	6.6(19)	9.6(17)	1.5(19)	0.5(18)	36.3(19)	10.7(17)	4.7(17)	0.9(17)	5.9(17)
dam	1.1(8)	1.8(7)	3.8(8)	0.1(2)	0.2(5)	6.9(6)	5.9(8)	1.7(7)	0.5(8)	2.(7)
moa	0.5(5)	1.9(8)	2.5(7)	0.7(17)	0.5(15)	9.6(7)	4.8(7)	2.1(8)	0.4(7)	6.3(18)
mfa	0.2(1)	0.4(2)	0.2(1)	0.2(4)	0.1(4)	0.3(1)	0.0(1)	0.0(1)	0.0(1)	0.6(4)
cam	3.(16)	3.3(12)	7.9(13)	0.4(12)	0.3(11)	29.3(17)	9.6(16)	3.9(14)	0.6(11)	3.3(11)
ic	0.4(2)	0.1(1)	0.2(3)	0.3(6)	0.0(2)	0.6(3)	0.1(3)	0.1(2)	0.0(3)	0.2(3)
cbm	0.5(6)	0.5(3)	0.2(2)	0.4(10)	0.0(1)	0.5(2)	0.1(2)	0.1(3)	0.0(2)	0.2(1)
amc	5.9(19)	6.5(18)	9.1(16)	0.8(18)	0.5(16)	25.5(15)	11.1(18)	3.8(12)	1.1(19)	5.0(15)
avg_cc	4.0(17)	1.7(6)	2.5(6)	0.3(7)	0.(9)	10.7(8)	2.9(6)	1.1(6)	0.8(15)	1.3(6)

In Table 4 the total number of points obtained by each metric in both examinations are given. The top ten metrics, with the maximal number of points, were chosen for the defect prediction.

The most suitable ten metrics obtained after the random forests analysis are following:

1. LOC—number of lines of code,
2. AMC—Average Method Complexity [16]—the average method size for each class: the size of pure virtual methods and inherited methods are not counted, large method, which contains more code, tends to introduce more faults than a small method,
3. RFC—Response For a Class (CK metric) [13]—the cardinality of the set of all methods that can be potentially executed in the response to the arrival of a message to an object, a measure of the potential communication between the class and other classes,
4. CE—Efferent Coupling (Martin metric) [14]—the number of other classes that the class depends upon, an indicator of the dependence on externalities, efferent means outgoing,
5. CBO—Coupling Between Objects (CK metric) [13]—the number of couplings with other classes (where calling a method or using an instance variable from another class constitutes coupling),
6. CAM—Cohesion Among Methods of Class (QMOOD metrics [12])—the relatedness among methods of a class based upon the parameter list of the methods, is computed using the summation of number of different types of method parameters in every method divided by a multiplication of number of different method parameter types in whole class and number of methods
7. WMC—(CK metric [13])—the number of methods in the class (assuming unity weights for all methods)
8. LCOM—Lack of Cohesion in Methods (CK metric [13])—counts the sets of methods in a class that are not related through the sharing of some of the class fields. The lack of cohesion in methods is calculated by subtracting from the number of method pairs that do not share a field access the number of method pairs that do.
9. NPM—Number of Public Methods (QMOOD metrics [12])—counts all the methods in a class that are declared as public, the metric is known also as Class Interface Size (CIS)
10. LCOM3—Lack of Cohesion in Methods (Henderson-Sellers metric [15]).

The metrics insignificant in the classification process appeared to be:

- dit—depth of inheritance tree, noc—number of children—(CK metrics [13]),
- ca—afferent couplings—a measure of how many other classes use the specific class (Martin metric [14]),
- dam—data access metric—the ratio of the number of private (protected) attributes to the total number of attributes declared in the class QMOOD [12],

Table 4 Number of points obtained by metrics

	wmc	dit	noc	cbo	rfc	lcom	ca	ce	npm	lcom3	loc	dam	moa	mfa	cam	ic	cbm	amc	avg_
MDA	152	56	36	121	149	132	85	133	148	113	168	98	72	58	125	37	36	131	62
MDG	126	40	42	142	161	120	100	142	112	108	178	66	99	20	133	27	32	166	86
Total	278	96	78	263	310	252	185	275	260	221	346	164	171	78	258	64	68	297	148
Accept	**Y**	N	N	**Y**	**Y**	**Y**	N	**Y**	**Y**	**Y**	**Y**	N	N	N	**Y**	N	N	**Y**	N

- moa—measure of aggregation—is a count of the number of data declarations (class fields) whose types are user defined classes QMOOD [12],
- mfa—measure of functional abstraction—the ratio of the number of methods inherited by a class to the total number of methods accessible by member methods of the class QMOOD [12],
- ic—inheritance coupling—the number of parent classes to which a given class is coupled. A class is coupled to its parent class if one of its inherited methods functionally dependent on the new or redefined methods in the class [16],
- cbm—coupling between methods—the total number of new/redefined methods to which all the inherited methods are coupled [16],
- avg_cc—the arithmetic mean of the CC (Cyclomatic Complexity—Mc Cabe) value in the investigated class [21].

3.3 *Comparison with Related Work*

Considerable research has been performed on the usefulness of software metrics in the defect predictions e.g. [1–20] but usually these experiments were not statistically significant so any empirical further validation is worthwhile. The random forest we used in our experiments were not used by other researchers (as far as we know).

Isong and Ekabua recently published a systematic review of 29 papers on software metrics for fault proneness prediction [17]. In the set of considered papers were e.g.: [12, 18, 19]. Isong and Ekabua concentrated on CK metrics and LOC and analyzed the significance of metrics in relation with the fault prediction. The results of theirs analysis are shown in Fig. 2.

However our method of metric selection (random forest) is different than the methods applied by other researchers the results concerning CK set of metrics are similar. The dit, noc metrics are insignificant for defect prediction.

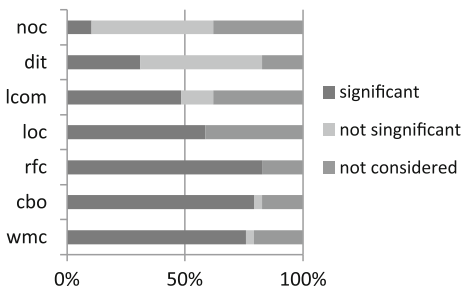


Fig. 2 Significance of CK metrics in defect prediction—based on 29 papers

4 Conclusions

However our analysis of the set of suitable metrics (by random forests) described in Sect. 3.2 was different than the analysis of e.g.: Singh et al. [19], Jureczko et al. [5, 6], Gyimothy et al. [11] and also the structure of examined projects was different, we obtained very similar set of metrics. It can be claimed that this set of metrics is very efficient and useful in the defect predictors.

The results of defect predictors could be influenced by the set of chosen metrics and specificity of projects being the subject of prediction and learning algorithms applied as well. It is important to choose appropriate set of metrics. In Sect. 3 the process of selecting metrics was described. These metrics appeared to be very successful in an experiment on the defect prediction presented in [24]. It should be also mentioned that results of prediction strongly depend on the data concerning defects. We were using the inscriptions in the repository of the version control system concerning defects, in fact some of them could be false.

References

1. Weyuker, E., Ostrand, T., Bell, R.: Adapting a fault prediction model to allow widespread usage. In: Proceedings of the International Workshop on Predictive Models in Software Engineering, Leipzig, Germany, May 12–13, 2008, PROMISE'08 (2008)
2. Weyuker, E., Ostrand, T., Bell, R.: Do too many cooks spoil the broth? Using the number of developers to enhance defect prediction models. Empirical Soft. Eng. **13**(5), 539–559 (2008)
3. D'Ambros, M., Robbes, R.: An Extensive Comparison of Bug Prediction Approaches. http://inf.unisi.ch/faculty/lanza/Downloads/DAmb2010c.pdf. Accessed January 2015
4. D'Ambros, M., Lanza, M., Robbes, R.: Evaluating defect prediction approaches: a benchmark and an extensive comparison. Empirical Softw. Eng. **17**(4–5), 531–577 (2012)
5. Jureczko, M., Spinellis, D.: Using object-oriented design metrics to predict software defects. In: Models and Methodology of System Dependability, pp. 69–81. Oficyna Wydawnicza Politechniki Wroclawskiej, Wroclaw, Poland, (2010)
6. Jureczko, M., Madeyski, L.: Towards identifying software project clusters with regard to defect prediction. In: Menzies, T., Koru, G. (eds.) PROMISE, p. 9. ACM (2010)
7. Catal, C., Diri, B.: Review: a systematic review of software fault prediction studies. Expert Syst. Appl. **36**(4), 7346–7354 (2009)
8. Moser, R., Pedrycz, W., Succi, G.: A comparative analysis of the efficiency of change metrics and static code attributes for defect prediction. In: Proceedings of the 30th International Conference on Software Engineering, ICSE '08, pp. 181–190, New York, NY, USA (2008)
9. Elish, M.O., Al-Yafei, A.H., Al-Mulhem, M.: Empirical comparison of three metrics suites for fault prediction in packages of object-oriented systems: a case study of eclipse. Adv. Eng. Softw. **42**(10), 852–859 (2011)
10. Mende, T., Koschke, R.: Revisiting the evaluation of defect prediction models. In: Proceedings of the 5th International Conference on Predictor Models in Software Engineering, PROMISE '09, pp. 7:1–7:10 (2009)
11. Gyimothy, T, Ferenc, R., Siket, I.: Empirical validation of object-oriented metrics on open source software for fault prediction. IEEE Trans. on Soft. Eng. **31**(10), 897–910 (2005)
12. Bansiya, J., Davis, C.G.: A hierarchical model for object-oriented design quality assessment. IEEE Trans. Softw. Eng. **28**(1), 4–17 (2002)

13. Chidamber, S.R., Kemerer, C.F.: A metrics suite for object oriented design. IEEE Trans. Soft. Eng. **20**(6), 476–492 (1994)
14. Martin, R.: OO Design Quality Metrics—An Analysis of Dependencies. Proceedings of Workshop Pragmatic and Theo. Directions in Object-Oriented Software Metrics, OOPSLA'94 (1994)
15. Henderson-Sellers, B.: Object-Oriented Metrics. Prentice Hall, Measures of Complexity (1996)
16. Tang, M.-H., Kao, M.-H., Chen, M.-H.: An Empirical Study on Object-Oriented Metrics. Proceedings of the Software Metrics Symposium, pp. 242–249 (1999)
17. Isong, B., Ekabua, O.: State of the art in empirical validation of software metrics for faut proneness prediction: systematic review. Int. J. Comp. Sci. Eng. Survey (IJCSES) **6**(6) (December 2015). doi:10.5121/ijcses.2015.6601
18. Goel, B., Singh, Y.: Empirical investigation of metrics for fault prediction on object-oriented software. Comput. Inf. Sci. SCI **131**, 255–265 (2008)
19. Singh, Y., Kaur, A., Malhotra, R.: Empirical validation of object-oriented metrics for predicting fault proneness models. Soft. Qual. J. **18**, 3–35 (2010)
20. Basili, V.R., Briand, L., Melo, W.L.: A validation of object-oriented metrics as quality indicators. In: UMIACS-TR-95-40 (1995)
21. McCabe, T.J.: A complexity measure. IEEE Trans. Soft. Eng. **2**(4), 308–320 (1994)
22. Bluemke, I., Stepien, A.: Experiment on defect prediction. In: Advances in Intelligent Systems and Computing, vol. 365, pp. 25–34. Springer International Publishing (2015). ISBN 978-3-319-19215-4. DOI:10.1007/978-3-319-19216-1_3
23. Random Forests—http://www.stat.berkeley.edu/ ~ breiman/RandomForests. Accessed 18 Jan 2016
24. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
25. Ckjm: http://gromit.iiar.pwr.wroc.pl/p_inf/ckjm. Accessed 22 Jan 2016
26. Stępień, A.: Fault prediction with object metrics (in polish). MSc thesis, Institute of Computer Science (2015)
27. Git: http://git-scm.com. Accessed January 2015
28. Apache Lucene: http://lucene.apache.org. Accessed January 2015
29. Apache Hadoop: http://hadoop.apache.org. Accessed January 2015
30. Apache Tika: http://tika.apache.org. Accessed January 2015
31. MyBatis.: http://mybatis.github.io/mybatis-3. Accessed January 2015
32. Apache Commons Lang. http://commons.apache.org/proper/commons-lang. Accessed January 2015
33. Hibernate: http://hibernate.org. Accessed January 2015
34. Jsoup: http://jsoup.org. Accessed January 2015
35. Apache Velocity: http://velocity.apache.org. Accessed January 2015
36. Elasticsearch: http://www.elasticsearch.org. Accessed January 2015
37. Apache Zookeeper: http://zookeeper.apache.org/. Accessed January 2015

Hybrid Generalized Additive Wavelet-Neuro-Fuzzy-System and Its Adaptive Learning

Yevgeniy Bodyanskiy, Olena Vynokurova, Iryna Pliss, Dmytro Peleshko and Yuriy Rashkevych

Abstract In the paper, a new hybrid generalized additive wavelet-neuro-fuzzy-system of computational intelligence and its learning algorithms are proposed. This system combines the advantages of neuro-fuzzy system of Takagi-Sugeno-Kang, wavelet neural networks and generalized additive models of Hastie-Tibshirani. The proposed system has universal approximation properties and learning capabilities which pertain to the neural networks and neuro-fuzzy systems; interpretability and transparency of the obtained results due to the soft computing systems; possibility of effective description of local signal and process features due to the application of systems based on wavelet transform; simplicity and speed of learning process due to generalized additive models. The proposed system can be used for solving a wide class of dynamic data mining tasks, which are connected with non-stationary, nonlinear stochastic and chaotic signals. Such a system is sufficiently simple in numerical implementation and is characterized by a high speed of learning and information processing.

Keywords Computational intelligence · Soft computing · Wavelet neuro-fuzzy networks · Hybrid intelligent systems

Y. Bodyanskiy (✉) · O. Vynokurova · I. Pliss
Kharkiv National University of Radio Electronics, 14 Nauky av.,
Kharkiv 61166, Ukraine
e-mail: yevgeniy.bodyanskiy@nure.ua

O. Vynokurova
e-mail: olena.vynokurova@nure.ua

I. Pliss
e-mail: iryna.pliss@nure.ua

D. Peleshko · Y. Rashkevych
Lviv Politechnik National University, 12 Bandera Street, Lviv 79013, Ukraine
e-mail: dpeleshko@gmail.com

Y. Rashkevych
e-mail: rashkev@polynet.lviv.ua

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_5

1 Introduction

Nowadays computational intelligence methods and especially hybrid systems of computational intelligence [1–3] are wide spread for Data Mining tasks in different areas under uncertainty, non-stationarity, nonlinearity, stochastic, chaotic conditions of the investigated objects and, first of all, in control, identification, prediction, classification, emulation etc. These systems are flexible because they combine effective approximating properties and learning abilities of artificial neural networks, transparency and interpretability of the results obtained by using neuro-fuzzy systems, the possibility of a compact description of the local features of non-stationary signals, providing wavelet neural networks and more advanced wavelet-neuro-fuzzy systems.

At the same time in the framework of such directions as Dynamic Data Mining (DDM) and Data Stream Mining (DSM) [4–6] more (if not the most) of observed systems appear either ineffective or inoperative in general. It connects with that the problems of DDM and DSM must be solved (including learning process) in on-line mode, when the data is fed to the processing sequentially, often in real time. It is clear that traditional multilayer perceptron trained based on back-propagation algorithm and requires the pre-determined training sample cannot operate in such conditions.

It is possible to implement the on-line learning process in the neural networks whose output signal depends on tuning synaptic weighs linearly, for example, radial basis function networks [RBFN], normalized radial basis function networks (NRBFN) [7, 8], generalized regression neural networks (GRNN) [9] and like them neural networks. However using of architectures that based on kernel activation functions is complicated, by so-called, course of dimensionality. Especially often such problem is appeared when using “lazy learning” [10] based on the conception “neurons on data point” [11].

Neuro-fuzzy systems have undoubted advantages over neural networks and first of all the significantly smaller number of tuning synaptic weights that allows to reduce time of learning process. Here it needs to notice the TSK-system [12] and its simplest version—Wang-Mendel system [13], ANFIS, DENFIS, SAFIS [14, 15] and etc. However, in these systems to provide the required approximating properties not only the synaptic weights but also membership functions (centres and widths) must be tuned. Furthermore, the training process of these parameters is performed using backpropagation algorithms in batch mode.

Hybrid wavelet-neuro-fuzzy systems [16], having a number of advantages, are too tedious from computational point of view, which complicates their using in real-time tasks. For solving such kind of problems, so-called, generalized additive models [17] are good. But such systems don’t operate under non-stationarity, nonlinearly and chaotic conditions.

In connection with that the development of hybrid system of computational intelligence is preferred. This system has to combine main advantages (the learning ability, the approximation and extrapolation properties, the identification of local

features of signals, the transparency and interpretability of wavelet neuro-fuzzy systems) with simplicity and learning rate of generalized additive models.

2 Hybrid Generalized Additive Wavelet-Neuro-Fuzzy System Architecture

Figure 1 shows the architecture of the proposed hybrid generalized additive wavelet-neuro-fuzzy system (HGAWNFS).

This system consists of four layers of information processing; the first and second layers are similar to the layers of TSK-neuro-fuzzy system. The only difference is that the odd wavelet membership functions "Mexican Hat", which are

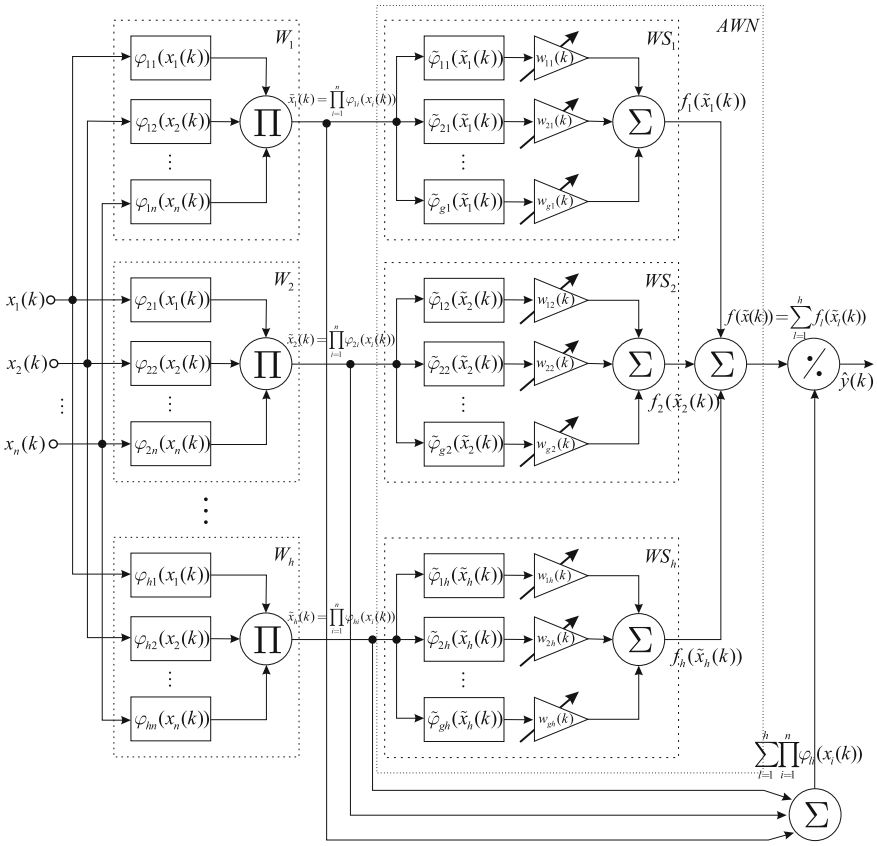


Fig. 1 Hybrid generalized additive wavelet-neuro-fuzzy system architecture

"close relative" of Gaussians, are used instead of conventional bell-shaped Gaussian membership functions in the first hidden layer

$$\varphi_{li}(x_i(k)) = (1 - \tau_{li}^2(k)) \exp(-\tau_{li}^2(k)/2) \tag{1}$$

where $x(k) = (x_1(k), \ldots, x_i(k), \ldots, x_n(k))^T - (n \times 1)$ is the vector of input signals, $k = 1, 2, \ldots$ is a current moment of time, $\tau_{li}(k) = (x_i(k) - c_{li})\sigma_{li}^{-1}$; c_{li}, σ_{li} are the centre and width of the corresponding membership function implying that $\underline{c} \le c_{li} \le \bar{c}$; $\underline{\sigma} \le \sigma_{li} \le \bar{\sigma}$; $i = 1, 2, \ldots, n$; $l = 1, 2, \ldots, h$; n is the input number; h is the membership functions number.

It is necessary to note that using the wavelet functions instead of common bell-shaped positive membership functions gives the system more flexibility [18], and using odd wavelets for the fuzzy reasoning does not contradict the ideas of fuzzy inference, because the negative values of these functions can be interpreted as non-membership levels [19].

Thus, if the input vector $x(k)$ is fed to the system input, then in the first layer the hn levels of membership functions $\varphi_{li}(x_i(k))$ are computed and in the hidden layer h vector product blocks perform the aggregation of these memberships in the form

$$\tilde{x}_l(k) = \prod_{i=1}^{n} \varphi_{li}(x_i(k)). \tag{2}$$

This means the input layers transform the information similarly to the neurons of the wavelet neural networks [20, 21], which form the multidimensional activation functions providing a scatter partitioning of the input space.

At that, therefore, in the region of input space, which remote from centres $c_l = (c_{l1}, \ldots, c_{li}, \ldots, c_{\ln})^T$ of multivariable activation functions

$$\prod_{i=1}^{n} (1 - \tau_{li}^2(k)) \exp(-\tau_{li}^2(k)/2), \tag{3}$$

the provided quality of approximation can be not high that is common disadvantage of all systems.

To provide the required approximation properties, the third layer of system is formed based on type-2 fuzzy wavelet neuron (T2FWN) [22, 23]. This neuron consists of two adaptive wavelet neurons (AWN) [24], whose prototype is a wavelet neuron of Yamakawa [25]. Wavelet neuron is different from the popular neo-fuzzy neuron [25] that uses the odd wavelet functions instead of the common triangular membership functions. The use of odd wavelet membership functions, which form the wavelet synapses $WS_1, \ldots, WS_l, \ldots, WS_h$, provides higher quality of approximation in comparison with nonlinear synapses of neo-fuzzy neurons.

In such a way the wavelet neuron performs the nonlinear mapping in the form

$$f(\tilde{x}(k)) = \sum_{l=1}^{h} f_l(\tilde{x}_l(k)) \tag{4}$$

where $\tilde{x}(k) = (\tilde{x}_1(k), \ldots, \tilde{x}_l(k), \ldots, \tilde{x}_h(k))^T$, $f(\tilde{x}(k))$—is the scalar output of wavelet neuron.

Each wavelet synapse WS_l consists of g wavelet membership functions $\tilde{\varphi}_{jl}(\tilde{x}_l)$, $j = 1, 2, \ldots, g$ (g is a wavelet membership function number in the wavelet neuron) and the same number of the tuning synaptic weights w_{jl}. Thus, the transform is implemented by each wavelet synapse WS_l in the k-th instant of time, which can be written in form

$$f_l(\tilde{x}_l(k)) = \sum_{j=1}^{g} w_{jl}(k-1)\tilde{\varphi}_{jl}(\tilde{x}_l(k)) \tag{5}$$

(here $w_{jl}(k-1)$ is the value of synaptic weights that are computed based on previous $k-1$ observations), and the general wavelet neuron performs the nonlinear mapping in the form

$$\tilde{f}(\tilde{x}(k)) = \sum_{l=1}^{h}\sum_{j=1}^{g} w_{jl}(k-1)\tilde{\varphi}_{jl}(\tilde{x}_l(k)) \tag{6}$$

i.e., in fact, this is the generalised additive model [17] that is characterised by the simplicity of computations and high approximation properties.

The output layer of system is formed by summator unit and it can be written in form

$$\sum_{l=1}^{h}\prod_{i=1}^{n}\varphi_{li}(x_i(k)) = \sum_{l=1}^{h}\tilde{x}_l(k) \tag{7}$$

and by division unit, which realizes the normalization for avoiding of "gaps" appearance in the parameters space.

In such a way the output of HGAWNFS can be written in form

$$\hat{y}(k) = \frac{\sum_{l=1}^{h}\sum_{j=1}^{g} w_{jl}(k-1)\tilde{\varphi}_{jl}(\tilde{x}_l(k))}{\sum_{l=1}^{h}\tilde{x}_l(k)} = \frac{\sum_{l=1}^{h}\sum_{j=1}^{g} w_{jl}(k-1)\tilde{\varphi}_{jl}\left(\prod_{i=1}^{n}\varphi_{li}(x_i(k))\right)}{\sum_{l=1}^{h}\prod_{i=1}^{n}\varphi_{li}(x_i(k))}$$
$$= \sum_{l=1}^{h}\sum_{j=1}^{g} w_{jl}(k-1)\frac{\tilde{\varphi}_{jl}(\tilde{x}_l(k))}{\sum_{l=1}^{h}\tilde{x}_l(k)} = \sum_{l=1}^{h}\sum_{j=1}^{g} w_{jl}(k-1)\tilde{\psi}_{jl}(\tilde{x}(k)) = w^T(k-1)\psi(\tilde{x}(k)) \tag{8}$$

where $\tilde{\psi}_{jl}(\tilde{x}(k)) = \tilde{\varphi}_{jl}(\tilde{x}_l(k))\left(\sum_{l=1}^{h}\tilde{x}_l(k)\right)^{-1} = \tilde{\varphi}_{jl}\left(\prod_{i=1}^{n}\varphi_{li}(x_i(k))\right)\left(\sum_{l=1}^{h}\prod_{i=1}^{n}\varphi_{li}(x_i(k))\right)^{-1}$, $w(k-1) = (w_{11}(k-1), w_{21}(k-1), \ldots, w_{g1}(k-1), \quad w_{12}(k-1), \ldots,$

$w_{jl}(k-1),\ldots,w_{gh}(k-1))^T$, $\tilde{\psi}(\tilde{x}(k)) = (\tilde{\psi}_{11}(\tilde{x}(k)),\tilde{\psi}_{21}(\tilde{x}(k)),\ldots,\tilde{\psi}_{jl}(\tilde{x}(k)),\ldots,\tilde{\psi}_{gh}(\tilde{x}(k)))^T$.

3 Adaptive Learning Algorithm of Hybrid Generalized Additive Wavelet-Neuro-Fuzzy System

The learning process of HGAWNFS is reduced in the simplest case to the synaptic weights tuning of wavelet neuron in the third hidden layer. For tuning of wavelet neuron its authors [25] used the gradient procedure which minimizes the learning criterion

$$E(k) = \frac{1}{2}(y(k)-\hat{y}(k))^2 = \frac{1}{2}e^2(k) = \frac{1}{2}\left(y(k) - \sum_{l=1}^{h}\sum_{j=1}^{g} w_{jl}\tilde{\psi}_{jl}(\tilde{x}(k))\right)^2 \quad (9)$$

and it can be written in form

$$\begin{aligned} w_{jl}(k) &= w_{jl}(k-1) + \eta e(k)\tilde{\psi}_{jl}(\tilde{x}(k)) = w_{jl}(k-1) + \eta(y(k)-\hat{y}(k))\tilde{\psi}_{jl}(\tilde{x}(k)) \\ &= w_{jl}(k-1) + \eta\left(y(k) - \sum_{l=1}^{h}\sum_{j=1}^{g} w_{jl}(k-1)\tilde{\psi}_{jl}(\tilde{x}(k))\right)\tilde{\psi}_{jl}(\tilde{x}(k)) \end{aligned} \quad (10)$$

where $y(k)$ is reference signal, $e(k)$ is learning error, η is fixed learning rate parameter.

For the speed acceleration of tuning process of synaptic weights under non-stationary conditions the exponential weighted recurrent least squares method can be used in form

$$\begin{cases} w(k) = w(k-1) + \frac{P(k-1)e(k)\tilde{\psi}(\tilde{x}(k))}{\beta + \tilde{\psi}^T(\tilde{x}(k))P(k-1)\tilde{\psi}(\tilde{x}(k))}, \\ P(k) = \frac{1}{\beta}\left(P(k-1) - \frac{P(k-1)\tilde{\psi}(\tilde{x}(k))\tilde{\psi}^T(\tilde{x}(k))P(k-1))}{\beta + \tilde{\psi}^T(\tilde{x}(k))P(k-1)\tilde{\psi}(\tilde{x}(k))}\right) \end{cases} \quad (11)$$

(where $0<\beta\leq 1$ is forgetting factor), which, therefore, can be numerical unstable for high tuning parameters number.

Under uncertain, stochastic or chaotic conditions, it is more effective to use the adaptive wavelet neuron (AWN) [26] instead of common wavelet neuron. In this case, we can tune not only synaptic weights, but the parameters of centres, widths and shapes.

The basis of adaptive wavelet neuron is the adaptive wavelet activation function that was proposed in [22] and can be written in form

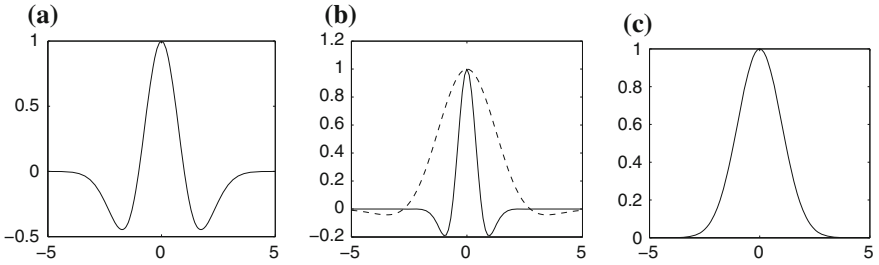


Fig. 2 Adaptive wavelet activation function: **a** $\alpha = 1, \sigma = 1$; **b** *dashed line* $\alpha = 0.3, \sigma = 1.5$; *solid line* $\alpha = 0.6, \sigma = 0.5$; **c** $\alpha = 0, \sigma = 1$

$$\tilde{\varphi}_{jl}(\tilde{x}_l(k)) = (1 - \alpha_{jl}(k)\tau_{jl}^2(k)) \exp(-\tau_{jl}^2(k)/2) \tag{12}$$

where $0 \leq \alpha_{jl} \leq 1$ is the shape parameter of adaptive wavelet function, if $\alpha_{jl} = 0$ it is conventional Gaussian, if $\alpha_{jl} = 1$ it is the wavelet "Mexican Hat", and if $0 < \alpha_{jl} < 1$ it is some hybrid activation-membership function (see Fig. 3).

Figure 2 shows the adaptive wavelet activation function with different parameters α и σ.

Basically to tune the centers, widths and shapes parameters we can use optimization of the learning criterion (9) by the gradient procedure (10), calculated the partial derivative on c_{jl}, σ_{jl}^{-1} and α_{jl}, but for the increasing speed of learning process we can use the one-step modification of Levenberg-Marquardt algorithm [27] for tuning all-parameters of each wavelet synapse simultaneously.

By introducing in the consideration $(g \times 1)$-vectors $w_l(k) = (w_{1l}(k), w_{2l}(k), \ldots, w_{gl}(k))^T$, $\tilde{\psi}_l(\tilde{x}_l(k)) = (\tilde{\psi}_{1l}(\tilde{x}_l(k)), \tilde{\psi}_{2l}(\tilde{x}_l(k)), \ldots, \tilde{\psi}_{gl}(\tilde{x}_l(k)))^T$, $c_l(k) = (c_{1l}(k), c_{2l}(k), \ldots, c_{gl}(k))^T$, $\sigma_l^{-1}(k) = (\sigma_{1l}^{-1}(k), \sigma_{2l}^{-1}(k), \ldots, \sigma_{gl}^{-1}(k))^T$, $\alpha_l(k) = (\alpha_{1l}(k), \alpha_{2l}(k), \ldots, \alpha_{gl}(k))^T$, $\tau_l(k) = (\tau_{1l}(k), \tau_{2l}(k), \ldots, \tau_{gl}(k))^T$, we can write the learning algorithm in form

$$\begin{cases} w_l(k) = w_l(k-1) + \dfrac{e(k)\tilde{\psi}_l(\tilde{x}_l(k))}{\eta^w + \left\|\tilde{\psi}_l(\tilde{x}_l(k))\right\|^2} = w_l(k-1) + \dfrac{e(k)\tilde{\psi}_l(\tilde{x}_l(k))}{\eta^w}, \\ c_l(k) = c_l(k-1) + \dfrac{e(k)\tilde{\psi}_l^c(\tilde{x}_l(k))}{\eta^c + \left\|\tilde{\psi}_l^c(\tilde{x}_l(k))\right\|^2} = c_l(k-1) + \dfrac{e(k)\tilde{\psi}_l^c(\tilde{x}_l(k))}{\eta^c}, \\ \sigma_l^{-1}(k) = \sigma_l^{-1}(k-1) + \dfrac{e(k)\tilde{\psi}_l^\sigma(\tilde{x}_l(k))}{\eta^\sigma + \left\|\tilde{\psi}_l^\sigma(\tilde{x}_l(k))\right\|^2} = \sigma_l^{-1}(k-1) + \dfrac{e(k)\tilde{\psi}_l^\sigma(\tilde{x}_l(k))}{\eta^\sigma}, \\ \alpha_l(k) = \alpha_l(k-1) + \dfrac{e(k)\tilde{\psi}_l^\alpha(\tilde{x}_l(k))}{\eta^\alpha + \left\|\tilde{\psi}_l^\alpha(\tilde{x}_l(k))\right\|^2} = \alpha_l(k-1) + \dfrac{e(k)\tilde{\psi}_l^\alpha(\tilde{x}_l(k))}{\eta^\alpha} \end{cases} \tag{13}$$

where $\tilde{\psi}_l^c(\tilde{x}_l(k)) = 2w_l(k-1)\sigma_l^{-1}(k-1)((2\alpha_l(k-1)+1)\tau_l(\tilde{x}_l(k)) - \alpha_l(k-1)\tau_l^3(\tilde{x}_l(k))) \times \exp(-\tau_l^2(\tilde{x}(k))/2)$; $\tilde{\psi}_l^\sigma(\tilde{x}_l(k)) = w_l(k-1)(\tilde{x}(k) - c_l(k-1))(\alpha_l(k-1)\tau_l^3 \times (\tilde{x}_l(k)) - (2\alpha_l(k-1)+1)\tau_l(\tilde{x}_l(k)))\exp(-\tau_l^2(\tilde{x}(k))/2)$; $\tilde{\psi}_l^\alpha(\tilde{x}_l(k)) = -w_l(k-1)\tau_l^2(\tilde{x}_l(k)) \times \exp(-\tau_l^2(\tilde{x}(k))/2)$; $\tau_l^2(\tilde{x}(k)) = \sigma_l^{-1}(k) \odot \sigma_l^{-1}(k) \odot (\tilde{x}_l(k)I_l - c_l(k-1)) \odot (\tilde{x}_l(k)I_l - c_l(k-1))$, $\odot$ is direct product symbol, I_l is $(l \times 1)$—the unit vector, η^w, $\eta^c, \eta^\sigma, \eta^\alpha$ are nonnegative momentum terms.

For increasing of filtering properties of learning procedure the denominators in recurrent equation system (13) can be modified in such way

$$\begin{cases} \eta^w(k) = \beta\eta^w(k-1) + \left\|\tilde{\psi}_l(\tilde{x}_l(k))\right\|^2, \\ \eta^c(k) = \beta\eta^c(k-1) + \left\|\tilde{\psi}_l^c(\tilde{x}_l(k))\right\|^2, \\ \eta^\sigma(k) = \beta\eta^\sigma(k-1) + \left\|\tilde{\psi}_l^\sigma(\tilde{x}_l(k))\right\|^2, \\ \eta^\alpha(k) = \beta\eta^\alpha(k-1) + \left\|\tilde{\psi}_l^\alpha(\tilde{x}_l(k))\right\|^2 \end{cases} \tag{14}$$

where $0 \le \beta \le 1$ have the same sense that in the algorithm (11).

4 Robust Learning Algorithm of Hybrid Generalized Additive Wavelet-Neuro-Fuzzy System

Although the square criterion allows to obtain the optimal evaluation when the processed signal and disturbances have Gaussian distribution, but when the distribution has, so-called, "heavy tails" (for example, Laplace and Cauchy distribution etc.) the evaluation which based on quadratic criterion can be inadequate. In this case the robust methods with M-criterion are more effective [28].

Introducing for the consideration the modified Welsh robust identification criterion in the form

$$E_R(k) = 1 - \exp(-\delta e^2(k)) \tag{15}$$

where $e(k)$ is the learning error, δ is positive parameter, which set from empirical reasoning and defined the size of nonsensitivity zone for the outliers.

Figure 3 shows the comparison the robust identification criterion with different values of parameter δ and least squares criterion.

Providing the sequence of the same transformation we can write the robust learning algorithm in form

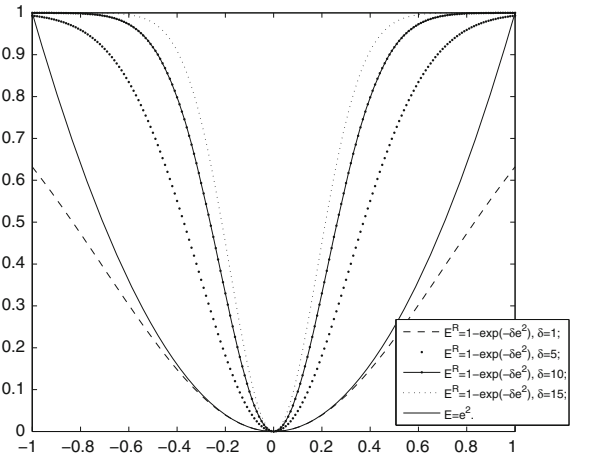


Fig. 3 Robust identification criterion (6) with different values of parameter δ

$$\begin{cases} w_l(k) = w_l(k-1) + \lambda_w \left(\delta e(k) \exp(-\delta e^2(k)) \tilde{\psi}_l(\tilde{x}_l(k)) \Big/ \eta^w \right), \\ \eta^w(k) = \beta \eta^w(k-1) + \left\| \tilde{\psi}_l(\tilde{x}_l(k)) \right\|^2, \\ c_l(k) = c_l(k-1) + \lambda_c \left(\delta e(k) \exp(-\delta e^2(k)) \tilde{\psi}_l^c(\tilde{x}_l(k)) \Big/ \eta^c \right), \\ \eta^c(k) = \beta \eta^c(k-1) + \left\| \tilde{\psi}_l^c(\tilde{x}_l(k)) \right\|^2, \end{cases} \tag{16}$$

$$\begin{cases} \sigma_l^{-1}(k) = \sigma_l^{-1}(k-1) + \lambda_\sigma \left(\delta e(k) \exp(-\delta e^2(k)) \tilde{\psi}_l^\sigma(\tilde{x}_l(k)) \Big/ \eta^\sigma \right), \\ \eta^\sigma(k) = \beta \eta^\sigma(k-1) + \left\| \tilde{\psi}_l^\sigma(\tilde{x}_l(k)) \right\|^2, \\ \alpha_l(k) = \alpha_l(k-1) + \lambda_\alpha \left(\delta e(k) \exp(-\delta e^2(k)) \tilde{\psi}_l^\alpha(\tilde{x}_l(k)) \Big/ \eta^\alpha \right), \\ \eta^\alpha(k) = \beta \eta^\alpha(k-1) + \left\| \tilde{\psi}_l^\alpha(\tilde{x}_l(k)) \right\|^2. \end{cases} \tag{17}$$

5 Conclusions

In this paper, the hybrid generalised additive wavelet-neuro-fuzzy system and its learning algorithms have been proposed. This system combines advantages of neuro-fuzzy system of Takagi-Sugeno-Kang, wavelet neural networks and generalised additive models of Hastie-Tibshirani.

The proposed hybrid system is characterised by computation simplicity, improving approximation and extrapolation properties as well as high speed of learning process. Hybrid generalized additive wavelet-neuro-fuzzy system can be used to solve a wide class of tasks in Dynamic Data Mining and Data Stream Mining, which are related to on-line processing (prediction, emulation,

segmentation, on-line fault detection etc.) of non-stationary stochastic and chaotic signals corrupted by disturbances. The computational experiments are confirmed to effectiveness of developed approach.

References

1. Rutkowski, L.: Computational Intelligence: Methods and Techniques. Springer, Berlin (2008). http://www.springer.com/us/book/9783540762874
2. Du, K.-L., Swamy, M.N.S.: Neural Networks and Statistical Learning. Springer, London (2014). http://www.springer.com/us/book/9781447155706
3. Mumford, C.L., Jain, L.C.: Computational Intelligence Collaboration, Fusion and Emergence. Springer, Berlin (2009). http://www.springer.com/us/book/9783642017988
4. Lughofer, E.: Evolving Fuzzy Systems—Methodologies, Advanced Concepts and Applications. Springer (2011). http://www.springer.com/us/book/9783642180866
5. Aggarwal, C.: Data Streams: Models and Algorithms (Advances in Database Systems). Springer (2007). http://www.springer.com/us/book/9780387287591
6. Bifet, A.: Adaptive Stream Mining: Pattern Learning and Mining from Evolving Data Streams. IOS Press, Amsterdam (2010). http://www.iospress.nl/book/adaptive-stream-mining-pattern-learning-and-mining-from-evolving-data-streams/
7. Sunil, E.V.T., Yung, CSh: Radial basis function neural network for approximation and estimation of nonlinear stochastic dynamic systems. IEEE Trans. Neural Netw. **5**, 594–603 (1994)
8. Bugmann, G.: Normalized Gaussian radial basis function networks. Neurocomputing **20**(1–3), 97–110 (1998)
9. Specht, D.F.: A general regression neural network. IEEE Trans. Neural Networks **2**(6), 568–576 (1991)
10. Nelles O.: Nonlinear System Identification. Springer, Berlin (2001). http://www.springer.com/jp/book/9783540673699
11. Zahirniak, D., Chapman, R., Rogers, S., Suter, B., Kabritsky, M., Piati, V.: Pattern recognition using radial basis function network. In: 6th Annual Aerospace Application of Artificial Intelligence Conference, pp. 249–260, Dayton, OH (1990)
12. Jang, J.-S.R., Sun, C.T., Mizutani, E.: Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence. Prentice Hall, NJ (1997). http://www.pearsonhighered.com/educator/product/NeuroFuzzy-and-Soft-Computing-A-Computational-Approach-to-Learning-and-Machine-Intelligence/9780132610667.page
13. Mendel, J.: Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice Hall, Upper-Saddle River, NJ (2001). http://www.pearsonhighered.com/educator/product/Uncertain-RuleBased-Fuzzy-Logic-Systems-Introduction-and-New-Directions/9780130409690.page
14. Kasabov, N.K., Qun, S.: DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. IEEE Trans. Fuzzy Syst. **10**(2), 144–154 (2002)
15. Rong, H.J., Sundararajan, N., Huang, G.-B., Saratchandran, P.: Sequential Adaptive Fuzzy Inference System (SAFIS) for nonlinear system identification and prediction. Fuzzy Sets Syst. **157**(9), 1260–1275 (2006)
16. Abiyev, R., Kaynak, O.: Fuzzy wavelet neural networks for identification and control of dynamic plants—a novel structure and a comparative study. IEEE Trans. Ind. Electron. **2**(55), 3133–3140 (2008)
17. Hastie, T., Tibshirani, R.: Generalized Additive Models. Chapman and Hall/CRC (1990)
18. Takagi, H., Hayashi, I.: NN-driven fuzzy reasoning. Int. J. Approx. Reason. **5**(3), 191–212 (1991)

19. Mitaim, S., Kosko, B.: What is the best shape for a fuzzy set in function approximation? In: Proceedings of the 5th IEEE International Conference on Fuzzy Systems "Fuzz-96", vol. 2, pp. 1237–1213 (1996)
20. Alexandridis, A.K., Zapranis, A.D.: Wavelet Neural Networks: With Applications in Financial Engineering, Chaos, and Classification. Wiley (2014). http://eu.wiley.com/WileyCDA/WileyTitle/productCd-1118592522.html
21. Bodyanskiy, Y., Lamonova, N., Pliss, I., Vynokurova, O.: An adaptive learning algorithm for a wavelet neural network. Expert Syst. **22**(5), 235–240 (2005)
22. Bodyanskiy, Y., Vynokurova, O.: Hybrid type-2 wavelet-neuro-fuzzy network for businesses process prediction. Bus. Inform. **21**, 9–21 (2011)
23. Bodyanskiy, Y., Pliss I., Vynokurova, O.: Type-2 fuzzy-wavelet-neuron for solving data mining problems. In.: Proceedings of the East West Fuzzy Colloquium 2012, 19th Zittau Fuzzy Colloquium. Zittau/Goerlitz: HS, pp. 96–103 (2012)
24. Bodyanskiy, Y., Kharchenko, O., Vynokurova O.: Least squares support vector machine based on wavelet-neuron. Inform. Technol. Manag. Sci. **7**, 19–24 (2014)
25. Yamakawa, T., Uchino, E., Miki, T., Kusanagi, H.: A neo-fuzzy neuron and its applications to system identification and prediction of the system behaviour. In.: Proceedings of the 2nd International Conference on Fuzzy Logic and Neural Networks "IIZUKA-92", pp. 477–483, Iizuka, Japan (1992)
26. Bodyanskiy, Y., Vynokurova, O., Kharchenko, O.: Least squares support vector machine based on wavelet-neuron. Inform. Technol. Manag. Sci. **17**, 19–24 (2014)
27. Wang, L.: Adaptive Fuzzy Systems and Control. Design and Stability Analysis. Prentice Hall, New Jersey (1994)
28. Cichocki, A., Unbehauen, R.: Neural Networks for Optimization and Signal Processing. Teubner, Stuttgart (1993)

Data Mining Algorithms in the Analysis of Security Logs from a Honeypot System

Michał Buda and Ilona Bluemke

Abstract Today many applications move to the Internet as web applications. This phenomenon causes new opportunities for attackers to take over servers or steal sensitive data such as credit card numbers, personal or corporate data. In this paper some analyses of data from a honeypot system of web application, implemented at the Institute of Computer Science, Warsaw University of Technology, are presented. The implemented honeypot has its own management software that helps to analyze the stored data. The honeypot was operating almost one year. Several data mining techniques were used to analyze the data collected by the honeypot and to detect important patterns and attacks. In this paper the results of the usage of algorithms MaxMiner and SED in the analysis of logs are presented.

Keywords Honeypot · Security · Data-mining

1 Introduction

Nowadays the security of web applications is becoming more and more important. Many applications are web-based and more and more users are keeping their data in a cloud. Paying and making money transfers by the Internet, using bank websites is common. Honeypot was introduced as a security mechanism able to detect or counteract attempts of unauthorized use of the information system. Honeypot is a specially prepared computer system with resources that imitate the real computer system. Such system can log the traffic and collect information about attacks and attackers [1].

M. Buda (✉) · I. Bluemke
The Faculty of Electronics and Information Technology,
Institute of Computer Science, WUT, ul. Nowowiejska 15/19,
00-665 Warszawa, Poland
e-mail: m.buda@ii.pw.edu.pl

I. Bluemke
e-mail: i.bluemke@ii.pw.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_6

At the Institute of Computer Science, Warsaw University of Technology in 2013 a honeypot, named WebHP, dedicated to web applications was designed and implemented [2]. The goal was to detect new vulnerabilities in web applications and new attack techniques [2]. The WebHP has its own management system—HPMS (HoneyPot Management System) which could be very useful in analyzing the inbound traffic to the WebHP. The amount of data received by honeypot is huge, so it was necessary to use the data mining approach in the analysis process.

The MaxMiner algorithm [3] which calculates maximal frequent itemsets and SED (Significant Event Discovery) [4] algorithm for the calculation of significant events [4] were used in the analysis of data collected by WebHP. The combination of both these approaches was also examined. The data stored by WebHP from 1st January 2013 to 30th June 2014 were used in the experiment.

In next section basics information concerning honeypots and some related work is presented. In the third section the theoretical background of maximal frequent itemsets and significant events are given. There is also a brief description of MaxMiner and SED algorithms. Next, the performed experiments and obtained results are described. Conclusions are given in the Sect. 5.

2 Honeypot Systems and Related Work

Honeypot is rather a technology than a specific solution. This technology is used in different aspects of information security domains. It can be used for the prevention, the detection and gathering information about Internet attacks, attackers and vulnerabilities. The value of collected data strongly depends on the interaction between malicious network traffic (cybercriminals attacks, viruses, internet worms) and the honeypot system.

The honeypot system can be implemented in various ways. It can be a computer system with unpatched "security holes" and specially prepared data, an application that simulates a service or even some records in a database. The task of the honeypot is to imitate a real system that interacts with incoming attacks.

The idea of a honeypot system was introduced by Clifford Stoll [5]. Another term—honeynet [1] is closely related to honeypot systems. It is a simple network with honeypot systems and resources that provide real systems and applications for attackers to interact with. It also collects information about attackers and attacks.

The honeytokens are also strongly related with the honeypot technology. This term was introduced by Augusto Paes de Barros in 2003 [6]. The concept of honeytokens is old and comes from such resources like books, maps, encyclopedias. The resources use fake entries or deliberately erroneous entries to detect copyright violation. In case of IT security the honeytoken is a fake digital entity that can exist in many different applications [1]. They can be of various aspects e.g. credit card number, spreadsheet, presentation, record in a database, fake user login.

There are several types of honeypot systems. They can be classified by the type of usage: production, research, honeytokens. Another classification can be made on

the basis of the level of interaction with the attackers: low, medium, high [1]. Honeypots can also be classified by the operating mode.

Research honeypots are used for gathering information about attacks and attackers without affecting the network. Collected data is used for detecting new types of attacks and vulnerabilities, to study the way in which cybercriminals attack the network and may also help to understand motives, behavior and organization of attackers. They can log any action executed in the system by an attacker [1]. Some collected datasets i.e. Kyoto 2006+ can be used by IDS (Intrusion Detection System) researchers to obtain practical, useful and accurate evaluation results [7]. Research honeypots are complex to deploy and maintain [1]. They are much more complicated than production honeypots. The amount of collected data is so huge that it can't be manually analyzed by a human. For analyzing such data, special data mining algorithms should be used.

Production honeypots are used in organizations' environments to protect important structures and data. They do not have as many functionalities as the research honeypots but are easier to build and deploy. Production honeypots often mirror a real network or services of the company. They also give less information than the research honeypots but can identify attacks patterns, provide information from which the system attack is coming and which resources are attacked [1]. Because the production honeypots do not prevent from attacks, they only gather data about them, so it is important to use additional special software, such as firewalls, IDS or IPS (Intrusion Prevention System) systems, to prevent attacks [1].

The honeypots can be also classified based on the level of interactions. The **low interaction honeypot** can only simulate services. As it is only a simulation and such honeypots do not have the operating system, they cannot be hacked. They operate as a program that do not interact with the attacker [1]. Such honeypots passively store information and do not influence the network traffic. They are easy to deploy and maintain.

Medium interaction honeypot are similar to the low interaction ones. They are also without the operating system, but simulated by them services are more complicated and give better illusion to the attacker of the real system or the network. Such behavior results in more interesting data about attackers and more complicated attacks [1]. They can dynamically customize their configuration i.e. create port listeners according to incoming TCP connections. They can also automatically download malware for the later analysis.

High interaction honeypots are the most complicated honeypot systems. It is hard to design, develop and maintain such a system. Because they involve operating system on which they work, they can be acquired and used by attackers [1]. It means that they should be constantly monitored to ensure that they have not become dangerous. Very good interaction with attackers results with the huge amount of valuable data about different threats.

Historically honeypots act as **servers**. They listen on a connection and when it occurs they interact with attackers and collect information about attacks. Such honeypots are classified as server honeypots. The server based approach is not able

to detect client-side attacks so a new type of honeypot was needed–the client honeypot [8].

Client honeypots crawl the network, interact with servers and classify them basing on their malicious nature. In contrary to server honeypots they simulate client-side software like a web browser but do not expose any services that could be hacked. Another difference is that the client honeypots are active and constantly interact with remote servers to be attacked. Next difference is that all connections to traditional honeypots are considered as suspicious network traffic. Client honeypots have to identify if a server is malicious or not [8].

In the literature there are some examples of the usage of data mining methods to the analyze of data gathered by a honeypot system. Ming-Yang Su in [9] states that attacks on some protocols or ports are more popular. Author also proposes own modification of WINEPI algorithm to calculate episodes in collected data. In this algorithm all obtained episodes are pruned by procedure RCEP (Redundant/Correlated Episode Pruning). Used data come from attacks on port TCP 445 which can be used by SMB (Serial Message Block protocol) in Microsoft Windows Network services [10].

Min Qin and Kai Hwang in [11] present that original approach introduced in [12] by Manilla, Toivonen, Verkamo generates many FER (Frequently Episodes Rules) which are complicated and redundant. They proposed new pruning technique. This technique uses new base-support mining scheme. In this scheme algorithm starts with the highest *minimum support* threshold to find episodes related to high frequency axis values [11]. Then the minimum support threshold is iteratively lowered by half until a very small threshold is reached. In each iteration new candidate FER with at least one new axis value is linked [11].

3 Theoretical Background

Data mining was used in experiments to find attack patterns in dataset obtained from honeypot logs. To find the patterns, two data mining algorithms: MaxMiner [3] and SED (Significant Event Discovery) [4] were chosen. Below, the algorithms and the theory of maximal frequent itemsets and significant events is briefly presented. Maximal frequent itemsets are calculated by the MaxMiner, and significant events are calculated by the SED algorithm.

3.1 MaxMiner Algorithm

MaxMiner [3] algorithm is used for the effective detection of maximal frequent itemsets in a dataset. Dataset, which the algorithm is using, contains all transactions according to the given range.

Frequent itemset is a subset of elements from different transactions, that occurs in greater number of transaction than the minimal support parameter, and which all subsets are also frequent. Elements of a frequent itemset are not all transactions, but just parts of them.

The minimal support is a parameter of this algorithm. Support of a set is the number of transaction, which contain all its elements. Minimal support may be defined as the number of transactions or the percentage of them in a dataset.

Maximal frequent itemset is a set, which do not have superset, which also is frequent. Because all frequent sets are subsets of maximal frequent set, the result achieved by MaxMiner represents all frequent sets present in the dataset.

3.2 Maximal Frequent Itemset

Let $\boldsymbol{I} = \{i_1, i_2, \ldots, i_n\}$ denote the set of all possible attributes in a database, which are often called items. Let D denote the set of transactions called the database. Transaction T is a subset of items from I that occur together in one operation. Each transaction has a unique identifier denoted as TID. The frequent set definition uses the idea of a support measure. Support of the set X with items from I is the number of transactions from a database D which contain all items from X [13]. It is said that the set X is the *frequent itemset* if and only if the support of X is greater than the given threshold parameter *minSup*. Frequent sets have a simple and interesting property. If any set X is frequent then all of its subsets are also frequent and have the support not less than the support of set X [14].

Maximal frequent set is defined as a frequent set in which all proper supersets are not frequent. In another words if such frequent set will be extended by any of item from I, then this new set will not be frequent.

For the calculation of maximal frequent itemsets the MaxMiner algorithm can be used. This algorithm [3] uses hashtree data structure to efficiently calculate maximal frequent itemsets.

3.3 SED Algorithm

SED (Significant Event Discovery) [4] algorithm is used for detecting significant events. *Significant event* is an event, for which in all time moments belonging to a period where it occurs, there exists a subsequence in which the event is frequent. The frequency is referred to as the minimal support parameter. Moreover, an event is *strongly significant* if the length of a period is not shorter than the window length. In SED algorithm, the method of sliding window is used. It allows to obtain better efficiency than calculating frequency of events in subsequence each time. SED is using two properties of significant periods:

1. after connecting two overlapping strongly significant periods the significant period is achieved,
2. each strongly significant period is also significant.

3.4 Significant Event

Let introduce some definitions that are important for understanding the significant event pattern. Suppose E is an alphabet of event identities. Data stream is defined as a tuple $\left(t_b^{DS}, t_e^{DS}, DS\right)$ where t_b^{DS} is a natural number which represents the time of the first event in the data stream and t_e^{DS} is a natural number which represents the time of the last event in the data stream. DS is a sequence of pairs (event and its time of occurrence) ordered by time from the oldest to the newest one. Formally it can be described by formula (1) [4]

$$S = \langle (e_1, t_1), (e_2, t_2), \ldots, (e_n, t_n) \rangle \tag{1}$$

where $e_i \in E$ and $t_b^{DS} \leq t_i \leq t_e^{DS}$ and $t_i \leq t_{i+1}$.

Substream of stream ***DS*** is defined as a 3-tuple (t_b^S, t_e^S, S) where t_b^S and t_e^S are respectively start and end time of substream ***S***. Numbers t_b^S and t_e^S are natural and pass condition (2)

$$t_b^{DS} \leq t_b^S \leq t_e^S \leq t_e^{DS} \tag{2}$$

S is defined as formula (3). Length of a substream is defined as a difference between t_e^S and t_b^S.

$$\left\{(e,t) \in DS : t_b^{DS} \leq t \leq t_e^{DS}\right\} \tag{3}$$

Support of an event e in the substream ***S*** (denoted as $eSup(S, e)$) is defined as the power of set of pairs (e, t) in ***S***. Event e is frequent in substream ***S*** if

$$eSup(S, e) > minESup$$

where *minESup* is a given parameter.

Period ***P*** is defined as the set of natural numbers that are not less than t_b and not greater than t_e. Event e is significant in the period ***P*** if for each t in ***P*** there exist substream ***S*** with length ***Wl*** and ***S*** contains time moment t in which e is frequent.

Event e is **strongly significant** in the period ***P*** if length of ***P*** is not lower than the length of window (denoted as ***Wl***) and event e is frequent in each substream that has length ***Wl*** covered by period ***P***.

4 Experiments and Results

At the Institute of Computer Science Warsaw University of Technology the honeypot WebHP [2] was implemented and used in experiments which have started at 1st January 2013. The system is still working. WebHP collects huge amount of data and it was necessary to implement the data mining module to analyze the stored data. This module is described in [15].

Below some results obtained by MaxMiner algorithm, SED algorithm and a combination of both these algorithms SMFIC (Significant Maximal Frequent Itemsets Calculator) are presented. Experiments were performed on data gathered from 1st January 2013 to the 30th of June 2014 by WebHP system.

4.1 Results of MaxMiner Algorithm

WebHP collects data in following format. Each HTTP header is stored with the value and the metadata as a timestamp in a single database record. It was necessary to transform this data to the format appropriate for MaxMiner algorithm [3]. Data has been converted to an array with records containing the transaction *id* and list of items. An item is an encoded pair: HTTP header and it is integer value.

After preprocessing data calculations were made. The algorithm was running on a part of datasets and was triggered by a crontab. Five configurations for the calculation of maximal frequent itemsets were established in the crontab [15]:

- every hour on data from the last hour,
- every 6 hours on data from last 6 hours,
- every day at midnight on data from the last whole day,
- every Sunday on data from last 2 weeks,
- on the first day of each month on data from the last 31 days.

The reason of such approach was that the calculation on huge database will take very long time, so it should not be performed often. The value of minimal support was set to 4, which made the calculations sensitive for occasional connections in a long period of time. In a short period of time the result should contain connections which occurred relatively often.

A script that calculated statistics data was also written [15]. This script was calculating the number of maximal frequent itemsets that occurred in examined periods: 1, 6, 24-h and others. The statistics were using only maximal frequent itemsets that have not occurred in shorter period, included in the current one. Also the information about the first and the last detection of maximal frequent itemset was stored. The last detection was updated if a new occurrence has been detected.

In the discussed time period there were observed 300 000 transactions, in which 34339 maximal frequent itemsets were detected. It is a significant number, but if

transactions related with spam attack on guestbook (one subpage of WebHP) and robots.txt file will be skipped, only 6685 other itemsets remain.

4.2 Results of SED Algorithm

The SED algorithm [4] uses several parameters: the window length, the minimal support and the minimal length of the significant event. Window length represents length of the currently analyzed substream. Minimal support represents minimal number of events in a window, when an event can be called frequent in a substream. The minimal length is the length of a minimal substream, in which an event was detected as significant.

The window length was set to 1, 6, 12, 24 h and 7, 14, 30, 60, 90 days. Values for the minimal support were chosen as 4, 8, 16, 32, 64, 128 appearances of an event. The minimal length was set to 1, 12, 24 h and 5, 7, 10, 14, 30 days [15].

As each connection to the honeypot system may contain a lot of headers, it was necessary to select the most important ones to define an event. In the first group of experiments [15] were chosen following fields: IP address, HTTP method and URL address. The second group of experiments has event defined as a pair IP address and URL. In both groups unique events were encoded into integers [15]. During the experiments 5 941 917 significant events were detected and 1520 of them were unique.

4.3 Results of SMFIC Algorithm

Experimentally the SED algorithm was executed on data obtained from the run of the MaxMiner algorithm [15]. This data contains maximal frequent itemset as an event, and as the time of an event, the start of the period for which this maximal frequent itemset was calculated, was assigned. The results of SMFIC, called significant maximal frequent itemsets, were itemsets marked by SED as significant events.

The window lengths was set to: 1, 24 h, 7 and 30 days. Value of the minimal support was set to 4 and 8 occurrences. Minimal length of significant event was set to 1, 24 h, 7 and 30 days. Calculations were performed for all configurations of these parameters.

Similarly as in the SED algorithm, the most common events were related with the same activities. The SMFIC was able to discover some events (as maximal frequent itemsets) that were not detected by SED. The SMFIC detected 268 286 significant maximal frequent sets, in which 12 932 were unique [15].

4.4 Activities Detected by all Algorithms

Some of the attack attempts were discovered by all algorithms used in the experiments. In this group are: spam attack on a simple guest book, phpMyAdmin scanning, searching vulnerabilities in PHP-CGI, joomla and CFIDE, w00wt00t scanning. The activities of bots (Internet robots) were also detected.

Attack on a simple guest book, which is a subpage of WebHP, generated the majority of patterns. This attack manifested itself by adding spam to the guestbook and was observed during all considered periods of time. Transactions came from different IP addresses. At first, attackers were using both GET and POST methods. After some time only POST method was used.

Second most common activity was phpMyAdmin scanning. Attackers were trying to check if the application is installed on the server. In SED, it could be recognized i.e. as event ('173.212.***.***', 'POST', '/phpMyAdmin/scripts/setup.php') [15]. In MaxMiner and SMFIC, the activity could be recognized by REQUEST_METHOD and SCRIPT_NAME headers. They contained values such as POST and/app/pma/setup.php. Different values of the header SCRIPT_NAME are showing that the attackers were checking different paths where the application can be installed. In MaxMiner and SMFIC, some patterns contain header [POST] [RAW POST] with coded content. After decoding this content it can be seen, that the attacker was trying to attach a remote file using remote FTP server. An example of decoded content is given below:

```
action = lay_navigation&eoltype = unix&token = d70b81...&
configuration = a:{i:;:0:"  PMA_Congig":{s:"source";s:(:
"ftp://37.59.XX.YY/pub/43.php";}}
```

The honeypot system detected also many attempts to install malicious software. Attackers were using encoded queries. ASCII characters were written in hexadecimal notation. The difference was in the version of PHP and CGI. An example of discovered event is: ('143.107.***.**', 'GET', '/cgi-bin/php5?%2D%64+%61…').

All algorithms recognized patterns according to web crawlers activities. Crawlers regularly scan the Internet and later analyze obtained data. Examples of crawlers are bots belonging to leading search engines suppliers, like Google, Bing or Yahoo. There are also crawlers that search for vulnerabilities in applications, like ZmEu or Morfeus. Most of them can be recognized after HTTP_USER_AGENT and HTTP_FROM headers. In example, internet bot ZmEu can be recognized by HTTP_USER_AGENT header, which has "ZmEu" value. The bot is searching for vulnerabilities and "holes" in phpMyAdmin, which enable to break into the database system.

4.5 Activities Detected not by All Algorithms

Some of the activities were recognized not by all algorithms. To this group belong HNAP attack, tmUnblock.cgi vulnerability, ipv4scan bot activity, including passwd file, attempts to soapCaller.bs file and Horde framework [15].

HNAP attack was detected by MaxMiner and SMFIC algorithm. It is related with the network traffic connected with a failure in some routers software. It allowed logging to administration panel of the router. The attack was recognized by REDIRECT_URL and PHP_AUTH_USER headers, which contained values such like "HNAP1" and "admin". In some patterns it were also observed REQUEST_ METHOD headers with unprintable bytes. In SMFIC results, the longest significant maximal frequent itemset has taken 144 days. It was detected for window length 30 days, minimal support 4 and minimal length 1 h. All these activities were detected between October 2013 and March 2014.

Another attack related with routers was tmUnblock.cgi file. It is an executable file, which oldest versions had many vulnerabilities. It is used by TheMoon worm. Unique significant maximal frequent itemset related with tmUnblock has taken from end of February to the beginning of March 2014.

Scanning performed by ipv4scan bot was detected by MaxMiner and SED algorithm. There is no information about bad habits of this bot and what it exactly does but it was observed that sometimes printers connected to the Internet print a paper with HTTP request from this bot. In SED, this attack is represented by event ('93.174.**.**', 'GET', 'http://ipv4scan.com/hello/check.txt'). All events were detected between February and June 2014 and all of them were made from one IP address.

Scanning related with the/etc/passwd file was detected by MaxMiner and SMFIC. It is a file that contains information about user in Unix systems. Example scanning had a value ABTPV_BLOQUE_CENTRAL in QUERY_STRING and REDIRECT_ QUERY_STRING headers. It allows the attacker to attach any local file. Most of maximal frequent sets connected with this file were discovered in August 2014.

5 Conclusions

MaxMiner, SED and SMFIC algorithms were used in analyzing data collected by WebHP honeypot. All these algorithms significantly decreased the amount of data that has to be analyzed by a human. Detected patterns of attacks are easy to analyze by WebHP/HPMS users. Several attacks patterns, such as phpMyAdmin, PHP-CGI, w00tw00t were discovered by all of the presented algorithms. As some of attacks e.g. HNAP1, tmUnblock.cgi, ipv4scan were discovered by some of them, it can be stated that there is no "ideal" algorithm and several ones should be applied. Furthermore, the quality of obtained results depends on algorithms parameters and

input definition. Sometimes it is difficult to set appropriate parameters values and for example results of the SED algorithm may depend on which items are chosen for the event definition. Despite of this inconveniences data mining algorithms are very useful in analyzing data stored by honeypots.

References

1. Mokube, I., Adams, M.: Honeypots: Concepts, Approaches, and Challenges. doi:10.1145/1233341.1233399
2. Buda, M.: Implementation and Integration of WEB Application Honeypot System Together with Advanced System for Monitoring and Preliminary data Analysis. Ba diploma, Institute of Computer Science, Warsaw University of Technology, Warsaw (2013) (in polish)
3. Bayard, R.J.: Efficiently mining long patterns from databases. In: SIGMOD '98 Proceedings of the 1998 ACM SIGMOD International Conference on Management of Data, pp. 85–93 (1998). doi:10.1145/276304.276313
4. Cabaj, K.: The new approach to the knowledge discovery in data streams. Ph.D. thesis, Institute of Computer Science, Warsaw University of Technology, Warsaw (2009) (in polish)
5. Biedermann, S., Mink, M., Katzenbeisser, S.: Fast dynamic extracted Honeypots in cloud computing. In: CCSW '12 Proceedings of the 2012 ACM Workshop on Cloud Computing Security Workshop, pp. 13–18 (2012). doi:10.1145/2381913.2381916
6. Honeytokens: The Other Honeypot (2016) http://www.symantec.com/connect/articles/honeyto-kens-other-honeypot. Accessed January 2016
7. Song, J., Takakura, H., Okabe, Y., Eto, M., Inoue, D., Nakao, K.: Statistical analysis of honeypot data and building of Kyoto 2006 + dataset for NIDS evaluation. In: BADGERS '11 Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security, pp. 29–36 (2006). doi:10.1145/1978672.1978676
8. Honeynet Project, (2016). https://www.honeynet.org/node/158. Accessed January 2016
9. Su, M.Y.: Applying Episode Mining and Pruning to Identify Malicious Online Attacks. doi:10.1016/j.compeleceng.2015.08.015
10. Server Message Block, (2016). https://en.wikipedia.org/wiki/Server_Message_Block. Accessed January 2016
11. Qin, M., Hwang, K.: Frequent episode rules for Internet anomaly detection. In: Third IEEE International Symposium on Network Computing and Applications, 2004, NCA 2004. Proceedings (2004). doi:10.1109/NCA.2004.1347773
12. Mannila, H., Toivonen, H., Verkamo, I.: Discovery of frequent episodes in event sequences. In: Data Mining and Knowledge Discovery, vol. 1, Issue 3 (1997). doi:10.1023/A:1009748302351
13. Agarwal, R., Aggarwal, C., Prasad, V.V.V.: A Tree Projection Algorithm for Generation of Frequent Item Sets. In: IBM T. J. Watson Research Center, Yorktown Heights, New York, 10598. doi:10.1006/jpdc.2000.1693
14. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques. Elsevier. ISBN-13: 978-0-12-088407-0 (2005)
15. Buda, M.: The Application of Data Mining Methods to Analyze Data from Honeypot Systems. M.Sc. diploma, Institute of Computer Science, Warsaw University of Technology, Warsaw (2015) (in polish)

About Synergy of Flows on Flower

Alexander P. Buslaev, Alexander G. Tatashev and Marina V. Yashina

Abstract Classical traffic flow theory was broadly separated into two branches: fluid-dynamical approach and and car-following micromodelling, and most important and difficult problems are network modelling of traffic flow. There exist many works of experimental and computer simulation types, but exact results fot saturated traffic flow on networks are appeared not often. We consider a movement of particles on network of special type as a set of contours with a common node. In 2009 we introduced a flower as network type for transport model, where the dynamical system defined by a system of differential equations on flower was developed. In this paper we study, for the system of connected contours, problem of search the conditions, such that the system becomes synergy mode independence on initial particles configuration at finite time, i.e. all particles move without delay for next time. It is proved, besides all the other results, that the search of synergy conditions is reduced to the investigation of the existence of solutions of linear Diophantine equations with two variables.

Keywords Dynamical system · Synergy · Network of flower type · Diophantine equations

1 Introduction

Research on traffic flow modelling has been developed during long time. At the beginning of 30th last centure Greenshilds [1], had began experimental study and created two types of traffic models: the follow-the-leader model and analog of

A.P. Buslaev (✉)
Moscow State Automobile and Road Technical University (MADI) and MTUCI,
125319 Leningradskii pr. 64, Moscow, Russia
e-mail: apal2006@yandex.ru

A.G. Tatashev · M.V. Yashina
Moscow Technical University of Communications and Informatics (MTUCI) and MADI,
8-a Aviamotornaya street, 111124 Moscow, Russia
e-mail: yash-marina@yandex.ru

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_7

fluid-dynamical model ("density-velocity-intensity"). Then these approaches have been broadly developed by Lighthill and Whitham [2], Drew [3] and so on. But at the second part of XX centure in connected with global automobilization growth it has been clear that obtained approaches can not satisfactorily describe of saturated traffic on complex network of megapolis.

At the beginning of 90th last centure Nagel and Schreckenberg [4, 5] introduced simulation model based on cellular automata for modelling of saturated traffic. Then this method has been widely spreaded as agent based modelling. Concerning exact results it can be noted that success has been more poor. In this content we would like to accent works of Blank [6, 7]. But and now the network case of this problem is open, and the exact results are appearing seldom, therefore they are more important.

We consider the movement of particles on "flower", i.e. the system of contours with one *a common point named a node.* Contour is a closed sequence of cells with clockwise numbering. At the current time there is *no more than one particle* in each cell. The particle in discrete times is moved on a contour sequentially counter-clockwise with the condition that the located in front cell is vacant. The number of particles on each contour is given. No particle more than one particle can simultaneously pass through the node located at the common point of contours. When two or more particles of different contours attempt simultaneous to cross the node, there appears *a conflict (competition)*, which is resolved by a given deterministic or stochastic rule. The transport model on the supporter type "flower" has been introduced and studied in [8, 9]. The considered system is a synthesis of flow system on two contours with particles. The model of isolated movement on one contour was developed in [6]. There it was obtained, in particular, that, if flow density on the contour is not more than half, then the flow velocity becomes possibly maximal in a finite time.

In this paper we study for the system of connected contours *problem of search the conditions, such that the system becomes synergy mode independence on initial particles configuration in finite time, i.e. all particles move without delay for next time*. It is proved, besides all the other results, that the search of synergy conditions is reduced to investigation of existence of solutions of *linear Diophantine equations* with two variables [10].

2 System Description

We assume the system contains L *contours* $C_1, \ldots, C_L$. Every contour C_i, $i = 1, \ldots, L$ has N_i cells numbered $(i, 0), \ldots, (i, N_{i-1})$. On the contour C_i the M_i particles move, where $(M_i < N_i)$, $i = 1, \ldots, L$. At any time $T = 0, 1, 2, \ldots$, each particle is in one of the cells. In every cell it can not be more than one particle.

All contours have a common point, this point is called *node.* The cells are numbered in reverse order the direction of particles motion (modulo N_i on contour

C_i, $i = 1, \ldots, L$). A particle on contour C_i passes through the common node moving between cells $(i, 0)$ and $(i, N_i - 1)$, $i = 1, \ldots, L$.

If at the time T a particle on contour C_i is located in (i, j)-th cell, $(i, j-1)$-th cell is vacant and the particle should not pass through a common node, then the particle at time $T + 1$ will be in $(i, j-1)$-th cell, $j = 1, 2, \ldots, N_{i-1}$, $i = 1, \ldots, L$.

With this numbering the $(i, 0)$-th cell is before the node, and particle located in (i, j)-th cell will be in the $(i, 0)$-th cell for the j displacements. If the forward cell is not vacant, then the particle remains at the same cell and does nit move.

If two or more particles tend to move in the corresponding vacant cells passing through a common node, then the *a conflict (competition)* takes place. In conflict case, only one of the competing particles moves according with given rule of the conflict resolution, for example, *each competing particle wins competition with equal probability.*

We call *the system state at T time* as vector

$$(k_{11}, \ldots, k_{1M_1}; \ldots; k_{L1} \ldots, k_{LM_L}),$$

where $k_{i1}, \ldots, k_{iM_i} (k_{i1} < \cdots < k_{iM_i})$ are the numbers of cells, in which particles of contour C_i, are located $i = 1, \ldots, L$ at this time. We assume that

(a) *the system is in synergy mode* with certain time point, if all the particles move on each time from that moment;
(b) the system state at this time moment *creates a synergy mode;*
(c) *the system has a total property of synergy,* if for any admissible states the system becomes synergy mode for the number of steps with finite expectation.

3 Synergy Mode of Flows on a Flower

We denote by $d(i_1, i_2) = GCD(N_{i_1}, N_{i_2})$ the greatest common divisor of numbers N_{i_1} and N_{i_2}, $i_1 \neq i_2$, $1 \leq i_1 \leq i_2 \leq L$.

Theorem 1 *System state*

$$(k_{11}, \ldots, k_{1M_1}, \ldots, k_{L1}, \ldots, k_{LM_L})$$

creates synergy mode if and only if,

(1) $\forall i, 1 \leq i \leq L, \forall j, 1 \leq j \leq N_i$

$$|k_{i,j+1} - k_{i,j}| > 1; \tag{1}$$

(2) $\forall i_1, i_2, 1 \leq i_1 < i_2 \leq L, \forall j_1, j_2, 1 \leq j_1 \leq M_{i_1}, 1 \leq j_2 \leq M_{i_2}$, *a number* $k_{i_1 j_1} - k_{i_2 j_2}$ *is not divided into* $d(i_1, i_2)$.

Proof If on any contour there are particles in neighboring cells, then one of the particles can not move in the current step, and, thus, the system does not generate a synergy mode. Assume that none contour has the particles in neighboring cells. We will show that this state creates synergy mode, if and only if, under no value i_1, i_2, j_1, j_2 $(i_1 \neq i_2,\ 1 \leq j_1 \leq M_{i_1},\ 1 \leq j_2 \leq M_{i_2})$ there is no non-negative integers numbers x, y, been a solution of the following equation

$$N_{i_1} x - N_{i_2} y = k_{i_1 j_1} - k_{i_2 j_2}. \tag{2}$$

Indeed, if for some $i_1 \neq i_2$ $(1 \leq j_1 \leq M_i,\ 1 \leq j_2 \leq M_j)$ there exists a solution (x_0, y_0) of Eq. (2), then the conflict will be no later than over $N_i x_0 + N_j y_0$ steps. If for all $i_1,\ i_2,\ j_1,\ j_2$ $(1 \leq j_1 \leq M_{i_1},\ 1 \leq j_2 \leq M_{i_2})$ the Eq. (2) does have solution, then the movement delays of the particles will not be, i.e. this state generates the synergy mode.

According to the theory of linear Diophantine equations with two variables, [10], the Eq. (2) has an integer solution if and only if, the number $k_{i_1 j_1} - k_{i_2 j_2}$ is divided into $d(i_1, i_2)$.

Theorem 1 is proved. □

Corollary 1 *If for some i_1, i_2 $(1 \leq i_1 < i_2 \leq L)$ the numbers N_{i_1}, N_{i_2} are co-prime, then the state generating synergy does not exist.*

Indeed, if for some i, j, $1 \leq i, j \leq L$ the numbers N_i, N_j, are relatively prime, then the condition (2) of Theorem 1 can not satisfy.

We formulate a theorem on the necessary conditions of the existence of at least one state generating a synergy mode. We denote by r_i *the density* of particles on the contour C_i Then $r_i = \frac{M_i}{N_i}$, $i = 1, \ldots, L$.

Theorem 2 (necessary conditions) *Necessary condition for the existence of state, generating synergy mode, is to fulfill the following relations*

(1) $\forall i,\ 1 \leq i \leq L$

$$r_i \leq \frac{1}{2} \tag{3}$$

(2) *the following inequality holds*

$$r_1 + r_2 + \cdots + r_L \leq 1. \tag{4}$$

Proof

(1) In accordance with the theorem condition, at any time the number of vacant cells on the contour C_i are smaller than number of particles. It means that all the particles of this contour can not simultaneously move as the particle moves to vacant cell. Hence the second statement of Theorem 2 is followed.

(2) Assume that the system is able to synergy. We denote by $D(1,\ldots,L) = LCM(N_1,\ldots,N_L)$ the least common multiple numbers $N_1,\ldots,N_L$. Then, at each time interval with duration D there exist r_iD steps, when the particle of contour C_i passes through the node. Since through the node no more than one particle can pass simultaneously, then the following inequality should be performed

$$\sum_{i=1}^{L} r_i D \le D.$$

It is equivalent to the inequality (4). Theorem 2 is proved.

We give an example that shows that the implementation of the necessary conditions are not a sufficient condition of existence of the state generating synergy. □

Example 1 Let $L = 2$, $N_1 = 6$, $N_2 = 3$, $M_1 = 3$, $M_2 = 1$. Then

$$r_1 + r_2 = \frac{M_1}{N_1} + \frac{M_2}{N_2} = \frac{1}{2} + \frac{1}{3} = \frac{5}{6} < 1,$$

i.e. the necessary condition of synergy is fulfilled, that given by the third statement of Theorem 2. However, as will be shown, synergy state does not exist.

Suppose for definiteness that first particle on contour C_1 is located in cell (1, 1). Then, if the state $(k_{11}, k_{12}, k_{13}; k_2)$ is a state of synergy, then $k_{11} = 1$, $k_{12} = 3$, $k_{13} = 5$. According to Theorem 1 for the synergy state the differences $k_{11} - k_2$, $k_{12} - k_2$, $k_{13} - k_2$ should not be divided into GCD (3, 6) = 3. But for any of the states $(1, 3, 5; 1), \ldots, (1, 3, 5; 6)$ this condition is not satisfied. Thus, really the state generating synergy does not exist.

Let $d(1,\ldots,L) = GCD(N_1,\ldots,NL)$ be the greatest common divisor of numbers $N_1,\ldots,N_L$. The following theorem gives a sufficient condition for the existence of state generating synergy mode for the case when each contour has one particle.

Theorem 3 (sufficient conditions) *Let $M_1 = \cdots = M_L = 1$. A sufficient condition for the existence of a state generating synergy is to perform at least one of the following conditions*

(1) *Suppose $M_1 = M_2 = \cdots = M_L = 1$ and*

$$d = GCD(N_1,\ldots,N_L) \ge L; \tag{5}$$

(2) *Numbers $N_1, N_2, \ldots, N_L$ are divided into L and $\forall\, i = 1,\ldots,L$, the following inequality is true*

$$R_i \le \frac{1}{L}. \tag{6}$$

(3) If $N_1 = \cdots = N_L$ *and the inequalities $r_i \le \frac{1}{L}$ are true for all $i = 1,\ldots,L$, then there exists at least one state of synergy.*

Proof

(1) If the inequality (6) is satisfied, then the state generating synergy is, for example, a state when the particle of contour C_i is in (i, k_i)-th cell, and the remainder of the division of k_i into $GCD(N_1, \ldots, N_L)$ is equal to i, $i = 1, \ldots, L$. But then, according to Theorem 1, this state is a state of synergy. The first proposition of Theorem 3 is proved.
(2) A state, when the particles of contour C_i are located in the cells with numbers, giving at division into L the remainder i, $(i = 1, \ldots, L)$, is a state of synergy. The second proposition of Theorem 3 is proved.
(3) The synergy state is the state, when particles of C_i-th contour are located in the cells with numbers divided by L give the remainder i, $i = 1, \ldots, L$. Theorem 3 is proved.

□

4 About Synergies on a Flower of Figure Eight Type

The following theorem gives synergy conditions in the case when the number of contours is equal to 2, i.e. "figure eight".

Theorem 4 *Let* $l = 2$.

(1) If $M_1 = M_2 = 1$, *then the system for any initial state becomes a synergy state for finite time, if and only if, the numbers* N_1 *and* N_2 *are not coprime.*
(2) *If numbers* N_1, N_2 *are even, and* $N_1 = N_2$. *Then necessary and sufficient condition, for the existence of at one state generating synergy, is to fulfill the following inequalities* $r_1 \leq \frac{1}{2}$, $r_2 \leq \frac{1}{2}$.

Proof

(1) If the numbers N_1 and N_2 are coprime, then, according to Theorem 1, the system can not become the synergy state. Let G be the set of states of the system (k_1, k_2), when $k_1 - k_2$ are divided into $GCD(N_1, N_2)$. If the system is in a state doing not belong to this set, then the system is in a synergy state. If the system is not in a state belonging to the set G, then not later than by $N_1 N_2$ steps a conflict of particles will be. After this conflict the system will be in a state that does not belong to the set G, and thus, the system is in synergy state. The first proposition of Theorem 4 is proved.
(2) In accordance with the second proportion of Theorem 3, the inequalities $r_1 \leq \frac{1}{2}$, $r_2 \leq \frac{1}{2}$ are necessary conditions of synergy. If these inequalities are satisfied, then the state of synergy is a state of the system when all particles of one contour are in cells with even-numbered numbers, and the particles of another contour are in odd-numbered cells. Theorem 4 is proved.

□

5 On Index Dial of States for Flower

We introduce the concept of *index dial of system state*. It will be used in the formulation of necessary and sufficient conditions of existence at least one state generating synergy mode.

Let $D = LCM(N_1, \ldots, NL)$ be the least common multiple of $N_1, \ldots, N_L$. Suppose that a system state

$$(k_{11}, \ldots, k_{1,M_1}, \ldots, k_{11}, \ldots, k_{1,M_L})$$

is given. We describe an algorithm for constructing the index dial corresponding to this state.

Step 1 We assigned the value 0 to all vector components of index dial D.

Step 2 For all $l = 0, 1, \ldots, \frac{D}{N_1} - 1$ we substitute coordinates with numbers $k_{11} + lN_1, \ldots, k_{1,M_1} + l_1 N_1$ to value 1.

Step 3 Suppose $i = i + 1$.

Step 4 For all $l = 0, 1, \ldots, \frac{D}{N_i} - 1$ we replace coordinates with numbers

$$k_{i1} + lN_i, \ldots, k_{i,M_i} + lN_i$$

either to value 1, if this coordinate had the value 0 or to $L + 1$, if this coordinate had the value not equal to 0.

Step 5 If $i < L$ and none of coordinates has value $L + 1$ then we increase the value of i to 1 and return to step 4. If $i = L$, or, if at least one coordinate value is equal to $L + 1$, then the algorithm stops.

In cyclic vector the *neighboring* coordinate with the number $i + 1$ is considered neighboring cell with number i, $i = 1, \ldots, D - 1$. We assume the cell with numbers 1 and D are neighboring to each other.

The following theorem gives necessary and sufficient condition of existence at least one state generating synergy of the initial system.

Theorem 5 *For all L,* $N_1, \ldots, N_L$, $M_1, \ldots, M_L$ *the state generating synergy exists if and only if in the case, if a set of index dials contains at least one element, such that the values of all coordinates of the vector are no more than L, and neighboring coordinates are not equal.*

Proof If an index dial with the value $L + 1$ corresponds to the system state, then no later than through D steps, the conflict will be in the system. If at current time, an index dial with two neighboring coordinates having the same value i $(1 \leq i \leq L)$ corresponds to the system state, then on the contour C_i there is a particle which can not move in the current time. Thus, the index dial, satisfying the condition of Theorem 5, corresponds to the state generating synergy mode. □

Assume the contrary, that the index dial, satisfying the condition of Theorem 5, corresponds to the initial system. Then conflict does not take place in current time

and in future, since a vacant cell will be ahead of any particle on each contour. Hence, the system is in synergy mode. Theorem 5 is proved.

6 Comments and Hypotheses

Let number of contours L, numbers of cells $N_1, \ldots, N_L$, and numbers of particles $M_1, \ldots, M_L$ be given.

Hypothesis 1 If there exists an initial condition with an admissible state of the index dial, then the system has a property of total synergy.

Hypothesis 2 If $N_1 = \cdots = N_L = 1$, then a necessary and sufficient condition for total synergy is the following

$$r_1 + \cdots r_L \leq 1.$$

Let us prove the theorems, confirming the hypothesis of sufficient conditions for existence of the total synergy property for the system.

Theorem A *Let $L = 2$, $N_1 = N_2$, $M_1 \leq \frac{N_1}{2}$ $M_2 = 1$, and the rule of conflict resolution is given when the particle on the contour C_1, wins the competition with probability $\alpha_1 > 0$. Then the system has a total property of synergy.*

Proof We will prove that if, from a certain moment, the particle on the contour C_2 not will win a competition at each time, then the system becomes a state of synergy at finite time. It will follow that the system becomes to a synergy state at time with a finite mathematical expectation, i.e., has the total synergy property, as, with positive probability, a particle on contour C_2 will always will lose a competition before system become to a synergy state.

Suppose that, starting from the moment T_0, a particle on contour C_2 always loses competition. Then after a moment of T_0 a particle on contour C_2 could not break the particle movement on contour C_1. Thus, particles are move as if the contour C_1 is single.

According to the results in [8], where the particle movement on a contour it considered, due to the condition $M_1 \leq \frac{N_1}{2}$ after a certain time T_1, a vacant cell will be in front of each particle on contour C_1. Therefore, the particles on contour C_1 will move at each step. After some time T_1 no more than one conflict can take place. After the moment the conflicts will not take place and the particle on contour C_2 also moves at each steps. Theorem A is proved.

The following theorem is a generalization of Theorem A. □

Theorem B *Let* $N_1 = \cdots = NL = N$, $L \leq \frac{N}{2}$, $M_1 \leq \frac{N}{2}$, $M_2 = \cdots = M_L = 1$, *and the rule of conflict resolution is given, such that each competition with a positive probability is wined by particle that having lower number of contour. Then the system has total synergy property.*

Proof Let us prove that, if, after a certain time, each competition will be wined by the particle with less number of contour, then the system will become a synergy state in finite time. Then it will follow that the system has a total property, as according with the theorem condition with positive probability a competition will always be wined by a particle with the lowest number of contours until the system become to a synergy state. Suppose that, starting from the moment T_0, a competition is wined by a particle with a lower number. Then, after a moment T_0, particles on contours could not break of particles movement on contour C_1. Thus, the particles on contour C_1 are moving as if the contour was C_1 will be single.$C_2, \ldots, CL$

In accordance with the results in [8], from some time T_1, there is a vacant call in front of each particle on contour C_1. Thus, all particles will move on the contour every step.

Let us prove that, for any given $i = 1, 2, \ldots, L$, there exists time T_i, such that after T_i all particles contours $C_1, \ldots, C_i$ move each step. If $i = 1$ this statement is trivial. Suppose that it holds for $i = k$, $1 \leq k \leq L - 1$ and let us prove that this should be a true statement for $i = k + 1$. Starting from the time T_k, all particles on contours $C_1, \ldots, C_k$ move at each step.

Particle on contour C_{k+1} does not move on the steps, when it loses a competition to particles on contours $C_1, \ldots, C_k$. And a particle on contour C_{k+1} can not be in conflict with the same particle one of these contours more than once.

Indeed, to be again in conflict with the same particle one on the contours $C_1, \ldots, C_k$, the particle on contour C_{k+1} should lose N conflicts to different particles. But there are only $\frac{N}{2} + L - 1 < N$ particles other than particles on contour C_{k+1}. Therefore, there exists a time $T_{k+1} \geq T_k$ such after that all particles on contours $C_1, \ldots, C_{k+1}$ move at each step. Theorem B is proved.

The following hypothese is the result of computer simulation study developed by student Pavel Sokolov.

Hypothesis 3 If the numbers $N_1, N_2, \ldots, N_L$ is divided into L and for $\forall\, i = 1, \ldots, L$ the inequality holds

$$r_i \leq \frac{1}{L},$$

then the system has the property of total synergies.

References

1. Greenshields, B.D., et al:. The photographic method of studying traffic behavior. In: Proceedings of the US Highway Research Board, vol. 13 (1934)
2. Lighthill, M.J., Whitham, G.B.: On kinematic waves: II. Theory of traffic flow on long crowded roads. Proc. R. Soc. Lond. Ser. A **229**, 281–345 (1955)
3. Drew, D.R.: Traffic Flow Theory and Control, 467 pp. McGraw-Hill, New York (1968)
4. Nagel, K., Schreckenberg, M.: A cellular automation model for freeway traffic. J. Phys. I. **2**, 2221–2229 (1992)
5. Nagel, K.: Particle hopping models and traffic flow theory. Phys. Rev. E. **53**(5), 4655 (1996)
6. Blank, M. L.: Exact analysis of dynamical systems arising in models of traffic flow. Russ. Math. Surv. **55**(3), 167–168. http://dx.doi.org/10.1070/RM2000v055n03ABEH000295 (2000)
7. Blank, M.L.: Ergodic properties of a simple deterministic traffic flow model. J. Stat. Phys. **111** (3–4), 903–930 (2003)
8. Buslaev, A.P., Yashina, M.V.: About flows on a traffic flower with control. In: The World Congress in Computer Science, Computer, Engineering and Applied Computing (WORLDCOMP'09), LAS Vegas, Nevada USA (July 13–16, 2009) in Proceedings of the 2009 International Conference on Modelling, Simulation and Visualization (MSV'09). Arabnia, H.R., Deligiannidis, L. (eds.) CSREAS Press, pp. 254–257 (2009)
9. Buslaev, A.P.: Traffic flower with *n* petals. J. Appl. Funct. Anal. (JAFA) **5**(1), 85–99 (2010)
10. Sierpinski, W.F.: On solution of equations in integer numbers. Publ. Inostr. Lit., Moscow (1961) (In Russian) (O rozwiazywaniu rownan w liczbach calkowitych, 1956)

Estimation of Travel Time in the City Based on Intelligent Transportation System Traffic Data with the Use of Neural Networks

Piotr Ciskowski, Adrianna Janik, Marek Bazan, Krzysztof Halawa, Tomasz Janiczek and Andrzej Rusiecki

Abstract The paper presents a method of travel time estimation by neural nets based on traffic data collected by cameras of the Intelligent Transportation System in the city of Wrocław, Poland. The methodology is explained of using traffic intensity data as neural net inputs and of using car plate number recognition system to provide target training data. The advantages of the suggested solution are pointed out. The results of preliminary research are presented for several travel routes and neural net architectures.

Keywords Intelligent transportation system · Neural nets · Travel time estimation

1 Introduction

Intelligent Transportation Systems (ITS) and advanced traffic information systems (ATIS) have become a standard in today's urban areas. Among many functionalities provided by them, traffic characteristics measurement, analysis and prediction are the main ones. While many of them are important in terms of urban traffic maintenance, the travel time information is a feature of great value for everyday road users.

A wide survey on traffic prediction methods, covering a broad range of techniques may be found in [1]. Prediction methods found in literature and implemented in various ATIS systems worldwide include simple techniques as instantaneous travel times, historical averaging or clustering of traffic patterns. More advanced prediction methods employ parametric models, such as micro-, macro- and mesoscopic simulation models, as well as different algorithms of time series prediction,

P. Ciskowski (✉) · A. Janik · M. Bazan · K. Halawa · A. Rusiecki
Faculty of Electronics, Department of Computer Engineering,
Wrocław University of Science and Technology, Wrocław, Poland
e-mail: piotr.ciskowski@pwr.edu.pl

T. Janiczek
Faculty of Electronics, Department of Control Systems and Mechatronics,
Wrocław University of Science and Technology, Wrocław, Poland

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_8

such as linear regression, Kalman filtering or ARIMA models. Examples of applying neural networks for traffic prediction may be found in [2–5]. Neural prediction solutions found in literature are usually focused on time series prediction, or use traffic models to provide training data for the net. A solution using neural nets and car plate recognition system for travel time prediction is presented in [5], although it was applied on an interurban highway, where the travel character is quite different from urban area, and different kind of inputs were used, adding travel times on sub-links to traffic volume and speed data.

In subsequent sections we present the Intelligent Transportation System in Wrocław, one of the major cities in Poland. The authors' research group introduces novel traffic modelling and optimization methods in ITS Wrocław. Two new methods of calculation the OD (origination-destination) matrix are presented in [6], while in [7] we suggest a method of using ensembles of neural nets and data from different parts of the city. This paper focuses on travel time prediction in the urban area. The methodology of travel time prediction by neural nets and car plate recognition data is explained later on and the results of preliminary research are presented.

2 Travel Time Estimation in ITS Wrocław

The Intelligent Transportation System in Wrocław, referred to as ITS Wrocław hereafter, is an integrated environment consisting of measurement, communication, database and data analysis infrastructure. The system is implemented on Microsoft SQL Server as the data engine.

Apart from datacenters used for data processing and the communication infrastructure, the system uses sensors for collecting data, e.g. induction loops or video cameras, as well as display devices, installed on bus/tram stops or above traffic lanes, for data presentation. Several web services and mobile applications have been deployed to provide more functionality for the citizens.

A few types of cameras are used to measure various aspects of the traffic:

- video surveillance cameras—used to capture video sequences, e.g. as evidence in case of traffic incidents,
- video detection cameras—used for traffic flow measurements,
- ARTR cameras—used for plate number recognition,
- VIM cameras.

From the above mentioned, two types of cameras are important in the scope of the presented research: video detection and ARTR.

There are 355 video detection cameras installed on 67 intersections. Each one monitors one or two road lanes, each lane assigned to one or two traffic streams (e.g. straight and turn). These cameras detect each vehicle appearing on a given lane and store the following data in the dedicated data table: camera id, detection and store time, the length, speed and type of the detected vehicle, the lane number.

Video detection cameras spot only the arrival of a vehicle. They record general, above mentioned data on each individual vehicle. The vehicles cannot be identified, although based on these data, one can calculate such traffic characteristics as: the intensity and density of traffic flow, or average speed for each traffic stream on every monitored intersection.

The ARTR cameras are used to capture and recognize car plate numbers. There are 38 cameras, installed on 19 locations along the main traffic routes in the city. When a certain car plate number is recorded at both ends of a route (and usually one or two cameras along the route for verification), the vehicle's travel time is calculated and stored in a dedicated data table.

The camera location is presented in Fig. 1. Many yellow pins overlap indicating two or more video detection cameras installed close to each other at the same intersection. The red pins are fewer as the number of ARTR cameras is much smaller.

The ARTR cameras are used to calculate the travel time along 5 main routes in the city—in both directions, given 2 variants for each route. The current travel times

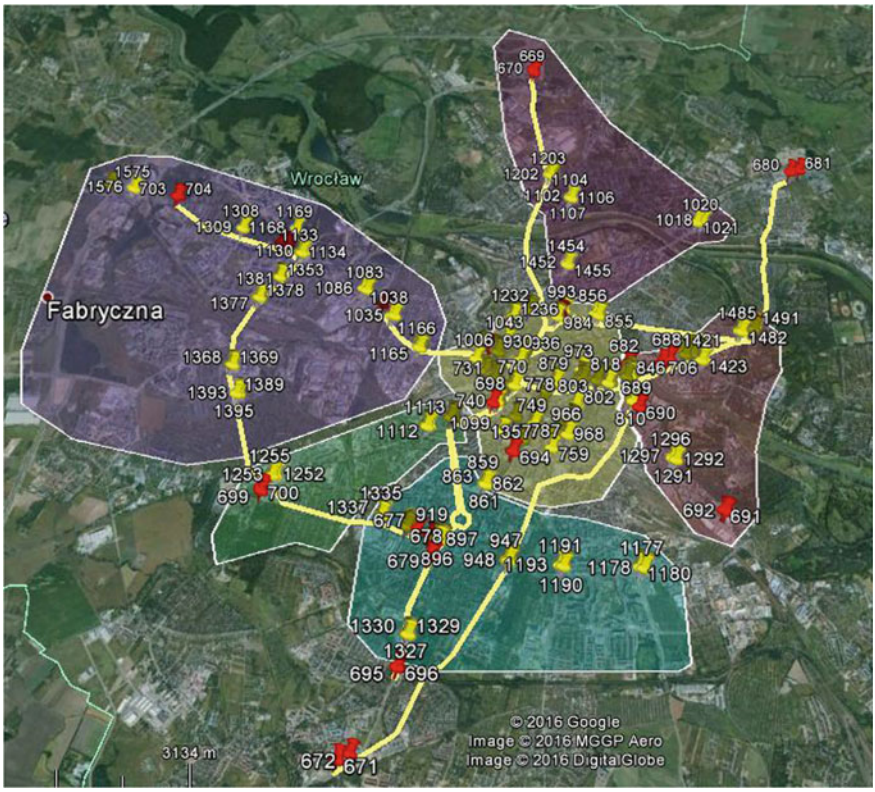


Fig. 1 Video detection (*yellow*) and ARTR (*red*) cameras location in Wrocław, city zones and major routes are also indicated, image: Google Earth

Fig. 2 An example of a variable message board in the city of Wrocław. Two travel times are displayed for the direction: City centre, via: Karkonoska St., and via: Ślężna St.

are displayed on 13 variable message boards, installed over the roads, see Fig. 2. Travel times for both variants of each route are displayed to suggest drivers the optimal one. The boards are situated ahead of the appropriate forks in the road. They are used to display other traffic alerts and information, as well.

2.1 Travel Time Calculation, Based on Car Plate Number Recognition

The travel time information system for drivers, already implemented in ITS Wrocław, is based on real travel times recorded by ARTR cameras. Each time a car (identified by its plate number) is recorded at the beginning and at the end of one of the routes defined in the system, the travel time for that route is updated. Although several procedures are used for car's route verification (e.g. one or two additional ARTR cameras along the route verify that the driver did not choose a detour), some disadvantages of that solution may be pointed out.

The routes defined in the system are relatively long, from 4 to 8 km, some of them covering the whole city area along main East-West or North-South transit directions. Therefore only a fraction of cars travelling along that routes is captured both at the beginning and at the end, and then considered for travel time calculation.

The number of measurements provided by ARTR cameras in travel time data tables is much smaller than the number of records in data tables filled by video detection cameras. For the preliminary research described later on, we used a snapshot of ITS data tables, covering 2 weeks in May 2014. During that period of time the number of cars travelling the entire routes ranged from 100 to 500, depending on the route. The measurements are asynchronous—scattered unevenly over the 2 week long recording time. As a consequence, travel time information stored in the system is quite often based on either a small sample of measurements, or on outdated records.

As the number of video detection cameras is much greater than ARTR cameras, they record data continuously all the time, and each one captures all the vehicles passing a certain lane, they provided approximately 16 million records during the same 2 weeks. That gives a number of approximately 45 thousand measurements per single camera in the same time, recorded constantly without any interruptions (apart from those caused by camera failure). Each route defined in ITS Wrocław is covered by 2–8 video detection cameras. The amount of data that may be used to estimate travel time is therefore much larger than the number of measurements on which it is calculated nowadays. Moreover the information is provided constantly as a continuous and steady stream of data.

Another disadvantage of the current system is the strict definition of travel routes. They are chosen as main transit arteries for vehicles traveling through the city. As the authors may confirm, many citizens would choose other paths for the origin-destination patterns defined in ITS Wrocław. Therefore, a more flexible information system is needed, with more alternative paths defined for each route. Another functionality expected by the drivers, is the estimation of travel time for any combination of origin and destination points, not confined to the 13 routes defined nowadays.

2.2 *Travel Time Estimation by Neural Nets, Based on Traffic Intensity and Average Speed*

The simplest way to improve the travel time information system would be to mount additional ARTR cameras on more routes than installed today. That solution causes some financial costs due to hardware purchase, installation, maintenance and data system reconfiguration. Flexibility of the system is still limited by the number of installed cameras and the routes resulting from camera locations.

Substitution of travel time calculation, implemented nowadays, with travel time estimation, based on the data provided by video detection cameras, would increase the accuracy of the system and allow to introduce new travel routes. It does not entail any additional costs, as video detection cameras are already installed and provide data. ARTR cameras would still be used as a source of training data—the real travel times, assumed to be correct after verification and filtering.

The idea of using camera data to train neural nets estimate the travel time, is illustrated in Fig. 3, while a sample route with both types of cameras is presented in Fig. 4.

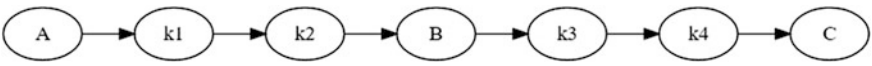


Fig. 3 The idea of travel time estimation system. *A, B, C* ARTR cameras, *k1–k4* video detection cameras

Fig. 4 An example of a travel route (Kochanowskiego–Lotnicza) used for travel time estimation. *Red pins* ARTR cameras, *yellow pins* video detection cameras, image: Google Earth

We train a separate neural net for each travel route. The data from ARTR cameras (marked as A and C in Fig. 3 and as red pins in Fig. 4) at both ends of the route are used to calculate the real travel times of individual vehicles, and provide the desired output for neural net training. One or two ARTR cameras along the route (camera B in Fig. 3, two cameras in Fig. 4) are used to verify each vehicle's path. Only the vehicles that appeared consecutively in points A-B-C are considered for travel time calculation. The system presented in the paper used this method of vehicle's path verification, as it is already implemented in ITS Wrocław. However we allow other methods in the future, e.g. monitoring the Bluetooth devices installed in vehicles, which is also performed in ITS Wrocław, and would provide more accurate route verification, not limited to the routes defined in the system.

2.3 Data Preprocessing

As mentioned in the previous section, a separate neural net is used for each route to estimate the travel time. Therefore all the nets have one continuous output, giving the estimated time. The number of inputs is twice the number of video detection cameras along the route for which the net is designed. The inputs are supplied with traffic intensity and average speed of vehicles, calculated for the last 15 min. As a result of training, the neural net supplied with traffic intensity and average speed data, will estimate the travel time for a given route.

A preliminary research has been performed to validate the validity of the described concept. It was based on real measurement data, provided in data table files by ITS Wrocław, covering 2 weeks in May 2014. Training data were prepared for the nets using a combination of database search and aggregation operations.

The data table with ARTR cameras records was searched for the vehicles traveling along that route, using verification methods described earlier (additional cameras along the route). Travel time for each vehicle was calculated and filtered for gross errors. Each travel time was taken as a desired output for one training sample. Then, for each travel time, the data tables with video detection camera records were searched for measurements from all the cameras along the defined route, covering 15 min before the vehicle began its move at the origin. For each camera, the number of recorded vehicles was aggregated in the given time window, giving the traffic intensity, while the vehicles' speed was averaged over the same period.

Figures 5 and 6 illustrate the input data for one of the nets used in the research, used to estimate travel time for the route marked in Fig. 4. The net was supplied with 16 inputs (intensity and average speed), provided by 8 cameras along the route. Figure 7 presents the desired outputs for the neural net, that is travel times for

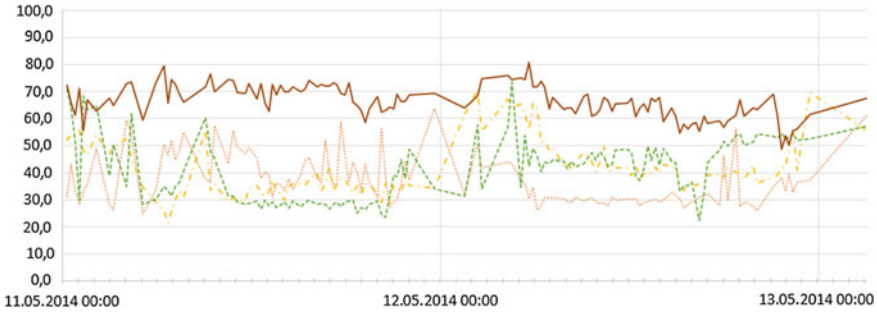


Fig. 5 Input data for the net-vehicles' average speed travel route: Hallera-Lotnicza, first 2 days, each series represents data from one video detection camera, 4 of 8 cameras on the route are presented for clarity

Fig. 6 Input data for the net-traffic flow intensity travel route: Hallera-Lotnicza, first 2 days, each series represents data from one video detection camera, 4 of 8 cameras on the route are presented for clarity

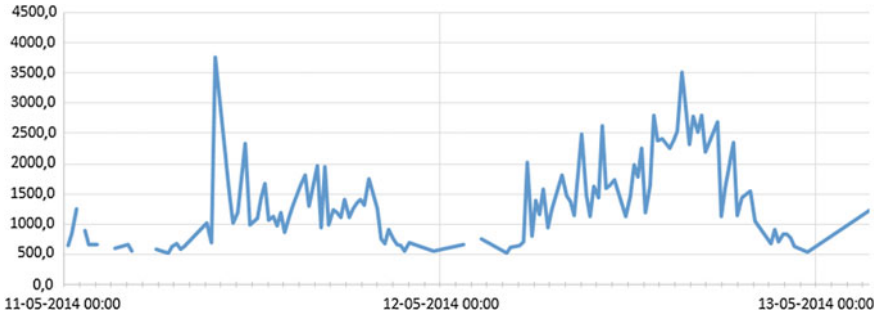


Fig. 7 Desired output of the net-travel time (in s) travel route: Hallera-Lotnicza, first 2 days

that route during the first two days. One may notice a clear dependence of travel time on the time of day.

The nature of the data supplied on neural net's inputs is illustrated in Fig. 8, where the distributions of average speed and traffic intensity for one of the cameras along Kochanowskiego-Lotnicza route are presented. As the route chosen for this experiment is one of the city's main arteries, the vehicles' average speed is relatively high, considering urban conditions. The distribution of traffic intensity reaches the peak value of about 250 cars per hour, referring to the congestion normally observed during afternoon hours. One may notice that due to the effective control by traffic light microregulation, the congestion does not progress to jam conditions, as the average speed does not drop below 25 km/h.

In addition to input data analysis, Fig. 9 presents the distribution of the desired outputs of the neural net, collected in the training set. The data provided by ITS measurements are a subject to statistical analysis in order to filter either too short, or too long travel times. As the route analyzed is about 7 km long, the travel time of below 2.5 min would result in values exceeding the city speed limit. On the other hand a 3 h travel time does also indicate a vehicle obviously spotted during two separate journeys and not filtered out by internal ITS algorithms.

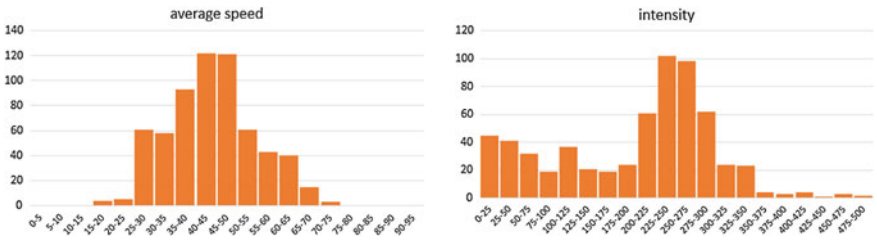


Fig. 8 Average speed and traffic flow intensity distribution for one of the cameras, travel route: Hallera-Lotnicza, 2 weeks

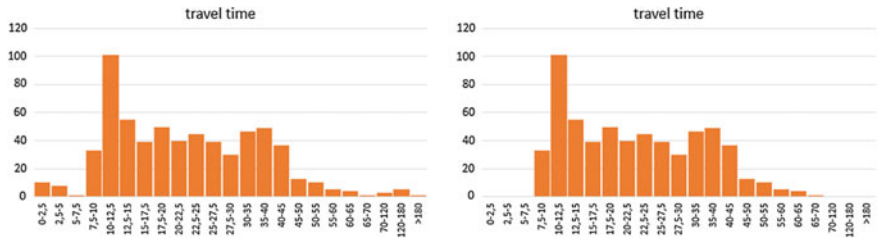


Fig. 9 Travel time distribution before and after data filtering, travel route: Hallera-Lotnicza, 2 weeks

2.4 Results

Preliminary research of the suggested methodology has been conducted, based on real data covering 2 weeks in May 2014. Training data for several routes was extracted from ARTR data tables, although only a few of them were considered for the test. As presented in Table 1, for most routes the amount of data provided by ARTR cameras is not sufficient to construct a reasonable training data set. For example, let us consider route 4, for which only 97 vehicles traveling the entire route were captured during 2 weeks. That number of training examples is far too low for a neural net with 12–23 inputs (supplied with data provided by 11 video detection cameras). On the other hand, for some other routes defined in the ITS Wrocław only 1 or 2 cameras could be matched along the route, providing not enough information on traffic conditions to estimate travel time. Route 1 was chosen for the preliminary research presented in this paper, due to the number of training examples.

A dedicated application has been implemented in ASP.NET C# and Python 3.4 environment, in order to extract data from ITS tables and to prepare training

Table 1 Training data

	Route	Length (km)	Number of video detection cameras	Number of examples
1	Hallera–Lotnicza	7.5	8	598
2	Lotnicza–Hallera	7.5	1	336
3	Kochanowskiego–Legnicka	6	8	486
4	Legnicka–Kochanowskiego	5.7	11	97
5	Krzywoustego–Krasińskiego	1.1	5	187
6	pl. Grunwaldzki–al. Karkonoska	3.1	2	264
7	Powstańców Śląskich–Żmigrodzka	4	4	412
8	Żmigrodzka–Powstańców Śląskich	7.2	9	104

Table 2 Training results

Inputs	Structure	MSE	Correlation coefficient R
Time + traffic intensity	9-3-1	0.02165	0.696
Time + traffic intensity	9-5-1	0.02277	0.751
Time + average speed	9-3-1	0.02658	0.763
Time + traffic intensity + average speed	17-3-1	0.01738	0.786
Time + traffic intensity + average speed	17-5-1	0.01973	0.761

datasets. Data search, aggregation and filtering was conducted according to the methods described in Sect. 2.3.

We used MATLAB environment to simulate and train the neural nets. Multi-layer perceptrons with one hidden layer of 10, 5 and 3 neurons were trained using the Levenberg-Marquardt training algorithm, an efficient and stable gradient method. The tanh activation function was used for the hidden layer neurons, while output neurons used linear activation function. The inputs and outputs of the nets were scaled to 0–1.

The recorded travel times of vehicles were used as the desired outputs. All the nets were supplied with time of the day on the first input. Other inputs included only the traffic intensity (cars/hour) on all intersections along the route, only the average speed (km/h), and the combination of both. As the data was provided by 8 cameras along the route, the number of nets' inputs was 9 or 17.

The results of training (averaged over 10 experiments, in each case) are presented in Table 2. The mean square error and the output-target correlation coefficient were calculated on a testing set, consisting of 15 % of examples not used for training and validation. The training usually took about 10 training epochs and was terminated before overfitting occurred. All the nets reached low values of mean square error. Increasing the input data vector by using both information on traffic intensity and average speed improved the results.

3 Conclusions

The paper presents a method of travel time estimation, performed by neural nets. The suggested technique does not predict the current travel time, based on past values, but estimates it using current data about the traffic on all the intersections along the analyzed route. Due to much larger amount of data provided by continuous traffic condition measurements, the trained neural net may provide more fluent and accurate estimations than the techniques used today, based on individual vehicles' travel time measurements.

Preliminary research, based on a small training set, proves that neural nets are useful in travel time estimation, although these results are still not satisfactory for a commercial use. The main reason is the weakness of training data, considering both

the amount of data, its distribution over time and accuracy. The training set used in this research was collected as a data table dump and covered only 2 weeks of measurements. That data set did not allow to extract a satisfactory number of training examples, as only a small fraction of the captured vehicles travelled the whole routes.

Further research on a much bigger data set is needed. Therefore a new training data acquisition system is being developed, allowing direct access to Wrocław ITS data tables, both historical and current, as well as redefinition of routes, so that more training examples are provided for shorter routes.

While car plate number recognition system is a suitable tool for providing target training data for neural nets, the methods of verification the travel route of each individual car still need improvement to avoid errors and undue variance in training data. Other types of neural nets, e.g. RBF nets or context-dependent neural nets, taking more advantage of the dependency of the travel time on time of the day, should also be considered.

Acknowledgments This work was partially supported from grant no S50242 at Wrocław University of Science and Technology.

References

1. van Hinsbergen, C.P.I.J., van Lint, J.W.C., Sanders, F.M.: Short term traffic prediction models. In: 14th World Congress on Intelligent Transport Systems, ITS 2007 11/2007
2. Park, D., Rilett, L.R.: Forecasting freeway link travel times with a multilayer feedforward neural network. Comput. Aided Civil Infrastruct. Eng. **14**, 357–367 (1999)
3. Kisgyorgy, L., Rilett, L.R.: Travel time prediction by advanced neural network. Periodica Polytechnica Ser. Civil Engineering **46**(1), 15–32 (2002)
4. Mark, C.D., Sadek, A.W., Rizzo, D.: Predicting experienced travel time with neural networks: a PARAMICS simulation study. In: Proceedings of the 7th International IEEE Conference on Intelligent Transportation Systems (2004)
5. Innamaa, S.: Short-term prediction of travel time using neural networks on an interurban highway. Transportation **32**, 649–669 (2005)
6. Bazan, M., Ciskowski, P., Halawa, K., Janiczek, T., Kozaczewski, P., Rusiecki, A., Szymański, A.: Two Methods of Calculation of the Origination Destination Matrix of an Urban Area, Raport W04/P-007/15. Wrocław University of Technology (2015)
7. Halawa, K., Bazan, M., Ciskowski, P., Janiczek, T., Kozaczewski, P., Rusiecki, A.: Road traffic predictions across Mayor City intersections using multilayer perceptrons and data from multiple intersections located in various places. IET Intel. Transport Syst. (submitted)

Evaluation of Deletion Mutation Operators in Mutation Testing of C# Programs

Anna Derezińska

Abstract Deletion mutation operators were endorsed to be beneficial in effective evaluation of tests for C and Java in comparison to traditional structural mutation operators. They were also supposed to generate fewer equivalent mutants. In this work we present evaluation of deletion mutation operators in the context of structural and object-oriented operators used in first and second order mutation testing of C# programs. Deletion operators could surpass standard mutation operators in the lower number of generated mutants and decreased mutation testing time. Experiments on C# programs were conducted on statement and operator deletion mutation operators. Considering mutation time, it could be profitable to replace other structural mutation operators with the deletion ones, although there are no distinctive results in reduction of the number of equivalent mutants.

Keywords Mutation testing · Deletion mutation operators · Mutation cost reduction · C# · Visual studio

1 Introduction

Mutation testing approach can assist in evaluation of a test set quality as well as in designing of effective test cases [1]. Its main idea relays on injection of small modifications into a program under tests. A modified program, so-called *mutant*, is run against a set of tests. If applying of tests allow us to differentiate a mutant behavior from the original program, a mutant is said to be *killed* by the tests. The tests are recognized as being able to detect program faults of the kind introduced. There are two reasons for a mutant not being killed, either the test set is not

A. Derezińska (✉)
Institute of Computer Science, Warsaw University of Technology,
Nowowiejska 15/19, 00-665 Warsaw, Poland
e-mail: A.Derezinska@ii.pw.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_9

sufficient and should be extended, or the mutant is *equivalent* and cannot be killed by any test. A wide utilization of mutation testing is hampered by expenses counted in terms of mutant number, executed tests, and also an effort devoted to identification of equivalent mutants [1–4].

On of solutions promoted to be beneficial within those problems is usage of deletion mutation operators [5–9]. They are based on deleting parts of a program code, while the standard structural operators deal mostly with the substituting of the corresponding code elements, e.g. mathematic operators.

A problem of program equivalence is undecidable and, therefore, exact complete identification of equivalent mutants cannot be realized in an automated way [10]. There are different approaches that assist in partial detection of such mutants, for example based on constraint solving, program slicing, or data-flow pattern techniques. But currently many of equivalent mutants still undergo a manual inspection, due to a lack of an efficient support in mutation testing tools. Recently, it has been reported about a case study in which automated application of Trivial Compiler Equivalence (TCE) technique to C programs detected 30 % of equivalent mutants in benchmark programs and 7 % in a set of real programs [3].

Equivalent mutants can lower the accuracy of mutation testing results and increase cost of a mutation testing process. In general, there are two main directions how to cope with the problem (i) avoid situations (e.g. mutation operators) that lead to generation of many equivalent mutants and (ii) examine mutants not being killed in order to determine their equivalence. Evaluation of deletion operators belong to the second approach, in which we try to avoid operators that create many equivalent mutants, or constraint their specifications.

In this paper, deletion mutation operators are studied in new application areas. Two operators were adopted to C#, implemented at the .NET intermediate language level and experimentally evaluated. Second order mutation testing [11] was also applied for the first time to deletion operators, and to other operators of C#. Moreover, sets of object-oriented mutation operators were surveyed in the context of deletion operators.

The general outcome of experiments confirmed ability of substituting and reducing a set of standard mutation operators in C# programs. Lower number of mutants with a comparable, or slightly declined mutation results were obtained in case of the first and second order mutation testing. However, no benefits are visible as far as equivalent mutants are concerned. Besides, it was shown that is no point in substituting OO operators that already cover different, specialized deletion concepts.

In the next Section different approaches to deletion mutation operators will be reviewed. Section 3 provides description of experimental set-up and discusses results of conducted experiments. Related work and some conclusions finish the paper.

2 Deletion Mutation Operators

Among deletion mutation operators the mostly used is a Statement Deletion Operator (SDL). The main idea behind SDL is deleting a single statement from a program. The modified program should remain correctly compliable, what is a prerequisite of applying any mutation operator. In general, the total number of SDL mutants is proportional to the number of statements in the program under test.

2.1 Statement Deletion and Other Structural Deletion Mutation Operators

Although the SDL idea is straightforward different variations were considered for various programming languages, and implemented in mutation tools. Statement deletion operator was introduced in Mothra for the Fortran language [12], which has no complex control structures. Preliminary positive results reported in [5] were conducted on C programs with a single statement deletion operator.

A definition of SDL that takes into account different control structures of Java, such as all possible cases, Boolean conditions, inner statements and nested control structures, was discussed in [6]. Therefore this SDL operator was more sophisticated than those defined for C or Ada.

Statement deletion operator called SSDL was implemented for C programs in the Proteum tool [7]. It systematically removes each statement block, and each individual statement inside each statement block. It delets each statement and all inner statements. Apart from SSDL, three other deletion operators were implemented in Proteum, namely operator deletion (OODL), variable deletion (VVDL), and constant deletion (CCDL). The OODL operator removes each arithmetic, relational, logical, bitwise and shift operator. In expressions, an operator is removed together with an operand in order to remain compliable. Variable (VVDL) and constant (CCDL) deletion operators remove all occurrences of variable references or constant references from every expression, accordingly. Removing a variable or a constant in an expression can also cause deleting an associated operator.

Two selected deletion mutation operators (SSDL and OODL) were adapted for C# programs [13]. They were designed to represent corresponding changes of the C# source code in the Intermediate Language of .NET and implemented in the VisualMutator tool [14]. This SSDL operator deletes only single assignments and does not removes declarations (e.g. *int v = value;* is mutated to *int v;*). Exceptionally, assignments in constructors are not removed. During code visiting, appropriate statements are marked as being mutable. Further in the mutation

process, they are substituted with empty statements. Converting expressions, the OODL operator works on the following operators: +, −, *, / , %, &&, ||, &, |, ^, <, >, >=, <=, ==, and !=. In case of binary operators, deleting of operands from both sides are taken into account. For example, for a statement *x = y + z;* two mutants will be created: *x = y;* and *x = z*. VVDL and CCDL operators were not implemented in VisualMutator, as they seemed to imitate more artificial faults, and overlap with the OODL operator to some extent.

2.2 *Deletion Mutation Operators Versus Object-Oriented Operators*

Deletion operators were presented as being new in relation to other structural mutation operators [5–7]. However this idea was successfully applied in the object-oriented (OO) operators, i.e. class level operators and other operators dealing with specific features of object-oriented languages. In Table 1 object-oriented deletion mutation operators are listed. In the third column operators proposed for Java [15] and implemented in the MuJava tool [16] are marked. Other Java tools have no or only scarce OO operators, as for example one deletion JTD operator in Judy [17]. In the subsequent columns deletion operators designed for C# language are marked [18], and those implemented in CREAM [19] or VisualMutator [10] tools. The last column deals with the OO operators adopted for C++ [20].

In evaluation of OO operators for Java, some of them created many equivalent mutants, such as deletion OO operators: JSD (91 %), IOD (43 %), PCD (57 %) [21]. Those numbers are very high, although they are sensitive to an experimental case study.

Mutants of CDD and CCA operators proposed for C++ language are counted as "potentially" equivalent [20]. In experiments, the percentage of equivalent mutants was high (57 % or 100 % of all mutants). Experiments conducted on other OO deletion operators, gave various results but the equivalent mutants could not be neglected. For example, there were over 50 % of equivalent mutants for OMD. In other cases, only few mutants for ISD and IOD were created, but all were equivalent.

In [22], a mutation operator estimation was given, based on experiments with various C# programs and 13 thousands of mutants. However, delete OO operators have in general no explicit classification about generating equivalent mutants by an operator.

Table 1 Object-oriented deletion mutation operators

		Operator	Java [15] and MuJava [16]	C# [18]	CREAM [19]	VisualMutator [10]	C++ [20]
1	IHD	Hiding variable deletion	v	v	v		v
2	IOD	Overriding method deletion	v	v	v		v
3	ISK/ISD	Base keyword deletion	v	v	v	v	v
4	IPC	Explicit call of a parent's constructor deletion	v	v	v		v
5	OMD	Overloading method deletion	v	v			v
6	JTD/CTD	This keyword deletion	v	v	v		v
7	JID/ CID	Member variable initialization deletion	v	v	v	v	v
8	PCD	Type cast operator deletion	v				v
9	JSD	Static modifier deletion	v				
10	JDC	Java-supported default constructor deletion	v				
11	EHR	Exception handler removal		v	v	v	v
12	OPD	Overriding property deletion		v			
13	OID	Overriding indexer deletion		v			
14	CCA	Copy constructor and assignment operator overloading deletion					v
15	CDD	Destructor method deletion					v

3 Evaluation

New delete mutation operators applied to C# programs were evaluated and compared with other standard and selected object-oriented operators in a case study. In this section a tool support, experimental set-up, results and their discussion will be presented.

3.1 Mutation Tool

The experiments were conducted using VisualMutator, a .NET mutation testing tool for C# programs [14]. VisualMutator (VM in short) was designed as an add-in for Visual Studio to verify quality of a test suite in an active C# solution.

An important feature of VM is insertion of mutation in the intermediate code (CIL) of .NET. This solution provides a faster mutant generation, because no program recompilation is necessary as in the case of parser-based approaches. On the other hand, application of others than simple structural mutation operators can be limited.

The previous version the tool (VM 2.0) supported selected structural mutation operators and 12 object-oriented ones [23]. Standard operators cover functionality of selective mutation [24] in order to avoid mutant redundancy. Among others, operators were chosen considering avoidance of a great number of equivalent mutant generation. Moreover an unambiguous reproduction of a C# code change at CIL level was also taken into account. The current version of the tool (VM 2.1) [13] was extended with two deletion mutation operators: SSDL and OODL presented in Sect. 2.

Another enhancement of the VM 2.1 is ability of higher order mutation testing [11]. Currently we can choose between first and second order mutation.

3.2 Experimental Set-up

In experiments two C# programs were used. The first program implemented the mathematical Prim's algorithm—a greedy algorithm that finds a minimum spanning tree for a weighted undirected graph. This combinatorial program was applied above all in investigation of mutation time and time-out limits. Then, it was primarily used in evaluation of standard operators, including delete ones, as in a one-class program generation of OO operators was limited.

The main object of the case study was the second program, which combined various library routines employed in program development. It comprised a number of text manipulation methods (e.g. dividing text in paragraphs, searching for words, counting how many times certain word appeared etc.), and mathematical functions

Table 2 Object programs

Program	Number of classes	Number of methods	LOC	Number of test classes	Number of test methods	LOC of tests	LOC total
Prim	1	1	100	1	1	55	155
Helper	10	52	454	5	24	251	705

(e.g. operations on fractions, integration, simple probability counting etc.). Numerous functionalities and origins of the routines made this so-called Helper program suitable for evaluation of diverse tests. A comprehensive class structure with inheritances, delegations, and overloads gave an opportunity to use various kinds of mutation operators.

The basic measures of the Prim and Helper programs are given in Table 2. All experiments were conducted on a PC with Intel Core 2 Duo CPU, 2.34 GHz, 4 GB Ram, with 64-bit Windows 7 operating system.

3.3 Experiment Results

Results of the first order mutation testing of the programs are given in Tables 3 and 4, accordingly. In the second column symbolic mutation operators are listed. Standard mutation operators stand for: AOR—Arithmetic Operator Replacement, SOR—Shift Operator Replacement, LCR—Logical Connector Replacement, LOR—Logical Operator Replacement, and ROR—Relational Operator Replacement.

From among object-oriented operators the following were used in experiments: DEH—Method Delegated for Event Handling Change, DMC—Delegated Method Change, JID—Field Initialization Deletion, JTD—This Keyword Deletion, PRV—Reference Assignment with Other Compatible Type, EHR—Exception Handling

Table 3 Mutation testing results of the Prim program

Operator type	Operator	Mutant number	Killed mutants	Equivalent mutants		Mutation score (%)
Structural	AOR	48	48	0	0	100
	ROR	112	95	10	9 %	93
	Total	160	143	10	6 %	95
Deletion structural	OODL	49	47	1	2 %	98
	SSDL	26	22	1	4 %	88
	Total	75	69	2	3 %	93
Object-oriented	JID	1	1	0	0	100
	JTD	1	1	0	0	100
	PRV	24	22	1	4 %	95
	Total	26	24	1	4 %	96

Table 4 Mutation testing results of the Helper program

Operator type	Operator	Mutant number	Killed mutants	Equivalent mutants		Mutation score (%)
Structural	AOR	232	217	10	4 %	98
	SOR	2	1	0	0	50
	LOR	20	16	2	10 %	89
	ROR	189	153	12	6 %	86
	Total	443	344	24	5 %	82
Deletion structural	OODL	161	120	14	9 %	82
	SSDL	52	43	2	4 %	86
	Total	213	163	16	8 %	83
Object-oriented	JID	11	10	0	0	91
	JTD	11	10	0	0	91
	PRV	39	34	2	5 %	92
	EHR	4	3	0	0	75
	EXS	7	1	2	29 %	20
	EAM	9	8	0	0	89
	MCI	5	5	0	0	100
	Total	86	71	4	5 %	88

Removal, EXS—Exception Swallowing, ISD—Base Keyword Deletion, EAM—Accessor Method Change, EMM—Modifier Method Change, and MCI—Member Call from another Inherited Class. If for a certain operator no mutants of a program were created, this operator is omitted in the tables.

In the subsequent columns numbers of all created and killed mutants are given. Among live mutants some equivalent mutants were manually detected. There are shown a direct number of equivalent mutants and a relative number calculated in the relation to all created mutants. The last column contains mutation score equal to the ratio of the number of killed mutants over the number of non-equivalent mutants.

3.4 Discussion

Considering standard mutation operators it should be noted that their number was already limited in the experiments. VisualMutator implements only a subset of many possible operators, including above all operators distinguished as selective ones [24]. Therefore, not so many redundant mutants, i.e. mutants that are killed by the same tests, are created [25]. However in both programs, the number of mutants generated by the new deletion operators is below a half of all structural mutants. Mutation scores of delete mutants are very similar to those of other standard ones.

A number of mutants cannot be overestimated as a cost factor. It has different impact in accordance to a mutation testing process. One of real cost measures is a mutation testing time including mutant creation and testing. In the experiments on the Prim program, total mutation time for all standard operators was about 473 s while for two deletion operators 232 s. It was a halve of time, so the relation was similar to the number of mutants. However in case of Helper, a bigger and more representative program, the corresponding times were equal to 1266 and 1020 s, accordingly. Mutant testing is realized in VM by many parallel threads. Therefore, the mutation time does not depend linearly on the mutant number.

Another important cost factor is existence of equivalent mutants. Since they are not processed in an automated way, cost of equivalent mutant detection can even surpass other mutation testing efforts. In the Helper program, number of equivalent mutants recognized for the OODL operator was the highest. Fraction of generated mutants that turned out to be equivalent is also higher for all deletion operators (8 %) than for other standard or object-oriented ones (5 % in both cases).

Object-oriented operators create, from definition, a lower number of mutants than the structural operators. The OO operators deal with diverse specific language features, and not generating mutants by some of them is a natural way of an operator set adjustment to a program requirements. In the previous studies on C#, mutation testing score for typical test sets delivered with open source projects was lower than for the structural mutants. It pointed to insufficient tests [19, 26]. In the presented experiments, mutation score of OO mutants is comparable to, or even a few % higher than, structural mutants. This fact can be caused by a higher quality of the test sets prepared for the programs and used in experiments.

While performing the second order mutation, the number of created mutants was, according to expectations, about a half lower in each operator group than in the first order mutation. The number of equivalent mutants was also lower. For example, 2 equivalence mutants of the Helper program were detected for standard operators, 8 for delete operators (OODL 6 and SSDL 2), and 5 for object-oriented operators. However, due to the low absolute values, no general rule among the groups can be observed.

Summing up, application of deletion mutation operators can be beneficial in comparison to other structural operators, as having less mutants the similar mutation results are obtained. However, experiment results do not endorse the hypothesis about a low number of equivalent mutants provided by deletion mutation operators.

The most of threats to validly concern a problem of generalization of results, as the experiments were conducted on a limited number of programs. Another issue is manual recognizing of equivalent mutants. Such a procedure cannot be treated as perfect, because we cannot proof that all equivalent mutants were detected and that any non-equivalent mutant was not counted to be equivalent one. However, after a careful examination and for a not very big number of mutants considered, we can assume that the presented results are correct.

4 Related Work

Statement Deletion (SDL) mutation operator was implemented in the MuJava tool and used in experiments on Java [6]. Mutants were created with SDL and other method level operators. The SDL mutation with 92 % mean mutation score did not surpass selective mutation in the terms of mutant killing ability. But it was a good alternative, giving over 81 % saving in the number of mutants compared to selective mutants. It was also observed that SDL gave poorer results (mutation score 85 % or less than 80 %), in case of testing classes having a lot of arithmetic or logical operations, or statements with complicated operations such as bit shifting. It was expected to have no equivalent mutants from SDL. In fact the number of equivalent mutants was low—only 3.74 % of the SDL mutants in comparison to 9.18 % of the selective mutants.

However, there was done an additional statistical meta-analysis of ICST'2013 papers [27]. Among others, experimental results from the paper about statement deletion mutation operator applied in Java [6] were examined towards their statistical significance. Statistical evaluation confirmed that the statement deletion operator produced significantly less mutants. On the other hand, mutation score for deletion operator was significantly worse than a MuJava test set with mutation score 1.

Experiments on four deletion operators were conducted on C programs [7, 8]. SSDL operator does not perform well for small program (mutation score below 80 %), but this result is not statistically important for a low number of mutants. For bigger programs average mutation score was about 93 %. The similar results were given for the VVDL operator. OODL resulted in the highest average mutation score (95 %). The worst results were obtained for the CCDL operator (80 %). Considering effectiveness, combination of SSDL and OODL was recommended following these experiments. The numbers of equivalent mutants generated for the particular operators were low on average (about 2 % for SSDL and OODL, 0.5 % for VVDL, and 1 % for CCDL). Low numbers of equivalent mutants were claimed to be valid in general. There were presented situations when an equivalent mutant can be created by SSDL due to a redundant code, but they were considered to be exceptional.

In [9] four structural deletion mutation operators were used in generation of test data reducing a mutation cost and data size, but no quantitative data were given.

5 Conclusions

The paper recalls the idea of deletion mutation operators in the structural and object-oriented mutation testing. It shows the benefits of the deletion operators applied in C# code in comparison to standard structural operators in terms of number of generated first and second order mutants. However, the improvement is

not as important considering mutation time. Moreover, ability of equivalent mutant creation is not significantly lower, neither in structural operators nor for object-oriented ones.

Acknowledgments I am very thankful to my student A. Bełz for extending the VisualMutator tool and performing mutation testing experiments.

References

1. Jia, Y., Harman, M.: An analysis and survey of the development of mutation testing. IEEE Trans. Softw. Eng. **37**(5), 649–678 (2011)
2. Usaola, M.P., Mateo, P.R.: Mutation testing cost reduction techniques: a survey. IEEE Softw. **27**(3), 80–86 (2010)
3. Papadakis, M., Jia, Y., Harman, M., Le Traon, Y.: Trivial compiler equivalence: a large scale empirical study of a simple, fast and effective equivalent mutant detection technique. In: 37th IEEE International Conference on Software Engineering, pp. 936–946, IEEE/ACM (2015)
4. Madeyski, L., Orzeszyna, W., Torkar, R., Józala, M.: Overcoming the equivalent mutant problem: a systematic literature review and a comparative experiment of second order mutation. IEEE Trans. Softw. Eng. **40**(1), 23–42 (2014)
5. Untch, R.: On reduced neighborhood mutation analysis using a single mutagenic operator. In: ACM Southeast Regional Conference, pp. 19–21 (2009)
6. Deng, L., Offutt, J., Li, N.: Empirical evaluation of the statement deletion mutation operator. In: 6th IEEE International Conference on Software Testing, Verification and Validation (ICST 2013), pp. 84–93, IEEE (2013)
7. Delmaro, M.E., Offut, J., Ammann, P.: Designing deletion mutation operators. In: IEEE International Conference on Software Testing, Verification, and Validation (ICST), pp. 11–20, IEEE (2014)
8. Delmaro, M.E., Deng, L., Durelli, V.H.S., Offut, J.: Experimental evaluation of SDL and one-op mutation for C. In: IEEE International Conference on Software Testing, Verification, and Validation (ICST), pp. 203–212, IEEE (2014)
9. Rani, S., Suri, B.: An approach for test data generation based on genetic algorithm and delete mutation operators. In: 2nd International Conference on Advances in Computing and Communication Engineering, pp. 714–718, IEEE (2015)
10. Budd, T.A., Angluin, D.: Two notions of correctness and their relation to testing. Acta Informatica **18**(1), 31–45 (1982)
11. Jia, Y., Harman, M.: Higher order mutation testing. Inf. Softw. Technol. **51**, 1379–1393 (2009)
12. DeMillo, R.A., Offutt, J.: Constraint-based automatic test data generation. IEEE Trans. Software Eng. **17**(9), 900–910 (1991)
13. Bełz, A.: Improving visual mutator—a visual studio add-in implementing mutation testing for C#. Bechelor thesis, Warsaw University of Technology (2016)
14. VisualMutator. http://visualmutator.github.io/web/
15. Ma, Y-S., Kwon, Y-R., Offutt, J.: Inter-class mutation operators for Java, In: Proc. of the 13th International Symposium on Software Reliability Engineering, ISSRE'02, pp. 352–363, IEEE Computer Society (2002)
16. Ma, Y-S., Offut, J.: Description of class mutation operators for Java. muJava Home Page. https://cs.gmu.edu/∼offutt/mujava/. Accessed Jan 2016
17. Madeyski, L., Radyk, N.: Judy—a mutation testing tool for Java. IET Softw. **4**(1), 32–42 (2010)

18. Derezińska, A.: Advanced mutation operators applicable in C# programs. In: Sacha, K. (ed.) IFIP vol. 227, Software Engineering Techniques: Design for Quality, pp. 283–288. Springer, Boston (2006)
19. Derezińska, A., Rudnik, M.: Quality evaluation of object-oriented and standard mutation operators applied to C# programs. In: Furia, C.A., Nanz, S. (eds.) Tools Europe 2012. LNCS, vol. 7304, pp. 42–57. Springer, Berlin (2012)
20. Delgado-Perrez, P., Medina-Bulo, I., Domingues-Jimenez, J.J., Garcia-Dominguez, A., Palomo-Lozano, F.: Class mutation operators for C++ object-oriented systems. Ann. Telecommun. **70**(3), 137–148 (2015) (Springer)
21. Hu, J., Li, N., Offutt, J.: An analysis of OO mutation operators. In: 4th International Conference on Software Testing, Verification and Validation Workshops, pp. 334–341 (2011)
22. Derezińska, A.: Classification of Advanced Mutation Operators of C# Language, Borzemski, L. et al. (Eds.) Information Systems Architecture and Technology, New Developments in Web-Age Information Systems, pp. 261–271. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław Poland (2010)
23. Derezińska, A. Trzpil, P: Mutation testing process combined with test-driven development in . NET environment. In: Zamojski, W., Mazurkiewicz, J., Sugier, J., Walkowiak, T., Kacprzyk, J. (eds.) DepCoS-RELCOMEX 2015. AISC, vol. 365, pp. 131–140. Springer, Heidelberg (2015)
24. Offutt, A.J., Lee, A., Rothermel, G., Untch, R.H., Zapf, C.: An experimental determination of sufficient mutant operators. ACM T. Softw. Eng. Methodol. **5**(2), 99–118 (1996)
25. Ammann, P., Delmaro, M. E., Offut, J.: Establishing theoretical minimal sets of mutants. In: IEEE International Conference on Software Testing, Verification, and Validation (ICST), pp. 21–30, IEEE (2014)
26. Derezińska, A., Szustek, A.: Object-Oriented testing capabilities and performance evaluation of the C# mutation system. In: Szmuc, T., Szpyrka, M., Zendulka, J. (eds.) CEE-SET 2009. LNCS, vol. 7054, pp. 229–242 (2012)
27. Hays, M., Hayes, J.H., Bathke, A.C.: Validation of software testing experiments. In: IEEE International Conference on Software Testing, Verification, and Validation (ICST), pp. 333–342, IEEE (2014)

Tracing Life Cycle of Software Bugs

Bartosz Dobrzyński and Janusz Sosnowski

Abstract The paper deals with the problem of monitoring software development and maintenance processes. In particular, we concentrate on data reported in software bug repositories. These data characterize the progress and effectiveness of the above mentioned processes. To analyze these data we have developed a special program which extracts information from typical bug repositories (e.g. Bugzilla) and generates various useful statistics. The analysis methodology is based on problem handling graphs (PHGs) and various metrics. The usefulness of this approach has been illustrated for some real open source and commercial projects. We present the measurement methodology and interpretation of the obtained results, which confirmed their practical significance.

Keywords Software reliability · Monitoring software development and maintenance · Software bug life cycle

1 Introduction

Developing complex software projects we face the problem of efficient development and maintenance processes. For this purpose various special tools (commercial and open source) are available [7, 8, 12, 13]. They are targeted at reporting the progress of these processes and the encountered program bugs. Papers in the literature devoted to these issues are mostly targeted at so called software reliability growth models (SRGMs) [1, 14], evaluation and improvement of development or maintenance processes (e.g. [9–11, 15, 16]). In fact, they base on some coarse grained data on detected bugs and their corrections. Having analyzed bug repositories for many

B. Dobrzyński (✉) · J. Sosnowski
Institute of Computer Science, Warsaw University of Technology,
Nowowiejska 15/19, 00-665 Warsaw, Poland
e-mail: Dobrzynski.b@gmail.com

J. Sosnowski
e-mail: J.Sosnowski@ii.pw.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_10

open source and some commercial projects we have found a lot of detailed data which can provide us with a better view on the considered processes as well as on the appearing problems [2]. In particular, we can generate so called problem handling graphs (PHG) introduced in our Institute [3, 4]. To facilitate generation of these graphs and their analysis we have developed a special tool—BugAnalysis. The analysis is targeted at structural graph properties and time statistics in correlation with various problem handling paths and states. This approach significantly extends classical analysis reported in the literature.

The developed tool has been provided with an API interface for popular bug repository Bugzilla (comparable with a large class of such tools). Nevertheless, it can be easily adapted to other repositories. An important capability of this tool is presenting and processing graphs taking into account various problem aspects. It provides also useful statistical measures. This approach has been verified for a benchmark of repository data related to various real projects.

In Sect. 2 we characterize data of bug repositories and outline the problem of their analysis. Section 3 describes the developed tool for extracting and analyzing repository data. Section 4 presents and explains results obtained for a sample of open source and commercial projects. Final conclusions are comprised in Sect. 5.

2 Bug and Issue Tracking Systems

In complex systems an important issue is tracking development and maintenance processes. For this purpose various tools are available. They are specified as project management, test management, bug tracking, issue tracking systems, etc. (compare [8, 18]). In this study we concentrate on tracking problem life cycle. The problem is some generalized notation for software bugs, configuration, user, hardware errors and other encountered anomalies. Such problems can be reported in data repositories of the above mentioned tools. The comprised information, creation and collection of reports, methods of accesses may differ upon the tools, their configuration and usage policies. These differences are not important in our approach, hence we base on relatively representative and universal Bugzilla system [17] widely used in practice. In the sequel we outline some basic features of this system.

Problem tracking systems provide capability of reporting and monitoring various data on detection or solution (handling) of problems by various actors (project mangers, developers, testers, users, etc.) which have different access and editing rights. The type and format of reported data can be configured taking into account default specifications and possible individual customization. Similarly, problem handling processes can be correlated with a default or customized workflow. The most important data are listed below:

Problem unique ID which can be supplemented with a name
Problem priority (assigned by someone authorized to do this, e.g. project manager) related to handling importance and problem severity. Typically, several severity

levels are distinguished ranging from blocking the system or application to those of lower significance (e.g. minor cosmetic issues). Additionally, problem complexity can be also given (related to needed fixing effort).
Problem description: detailed description of the anomalous behavior, list of steps to reproduce the detected problem. This can be supplemented with various attachments, e.g. print screens, code segments (available as additional files or URL pointers). Sometimes, it is useful to add short summaries and keywords describing the considered problem (this may simplify searches and queries). Moreover, some localization data can be added indicating code revision, module or component affected by the problem.
Problem progress history: it describes the life cycle for each bug related to sequence of handling processes specified as states. It comprises entry and exit times related to the involved states. Depending upon the state and assumed handling scheme the problem can move to another state or can be deleted (rejected).
Actors involved in problem handling (unique personal IDs)
Exchanged information between actors dealings with the specified problem or problem classes.

Sometimes specific additional information is added to registered problems, e.g. estimated time and deadline for resolving the problem, flags marking problems with specific features, target version in which the problem should be fixed, problem dependencies (a problem cannot be fixed unless other problems are resolved). This can be combined with sending alerts or emails to specified recipients.

The specified data is available in databases of the above mentioned tools and can be exported in various formats for subsequent analysis. Nevertheless, these tools provide some searching and query mechanisms for manual tracing individual problems or give some general statistics, e.g. related backlog of problems in a specified state.

Tracing life cycle of problems relates to time tracking of traversed states within the handling processes starting from the registration of the problem and terminating at its solution. In this process it is useful to base on the introduced PHG graphs. The complexity of these graphs depends upon the number of states and edges. Having analyzed bug repositories for many commercial and open source projects we have found that the number of states ranges from about 10 to less than 30. They are usually defined within development companies or projects. For illustration we give a list of states used in open sources projects (e.g. Tomact, Eclipse): NEW, ASSIGNED, RESOLVED_LATER, RESOLVED_MOVED, RESOLVED_FIXED, RESOLVED_INVALID, RESOLVED_WONTFIX, RESOLVED_DUPLICATE, RESOLVED_WORKSFORME, REOPENED, NEEDINFO, VERIFIED, CLOSED.

For comparison states used in commercial projects within one company were as follows: NEW, ANALYSING, REJECTION_CLOSED, REJECTION_SUSPENDED, WORK_IN_PROGRES, REJECTED, WAITING_FOR_RETEST, RETESTING, REOPENED, FIXED, SUSPENDED, CLOSED. Another example with 26 states is given in [3].

The names of states are self-explanatory, however they will be commented in Sect. 3 in relevance to PHG graphs and paths. It is worth noting that the initial state is NEW and the final one (terminal) should be CLOSED, however quite often the last state of the problem is the one with prefix RESOLVED. States NEEDINFO and REOPENED refer to problems which need delivery of additional information and reprocessing, respectively.

An important issue is to trace the states and paths of problem handling, the number of related problems, time properties, various statistics in relevance to problem priority or sources, methods of resolution, etc. We can identify typical or dominating situations, find bottlenecks, search for abnormal problem processing and development or maintenance deficiencies. In particular, we can identify correctly resolved and verified problems, percentage of duplicated (the same problems announced by different actors), rejected or invalid problems due to user misunderstanding, negligible inconsistencies, etc. Such advanced analysis needs some specialized tool, for this purpose we have developed a system targeted at graph oriented analysis described in Sect. 3. It combines database, statistical and visualization techniques.

3 PHG Analysis

Evaluation of project progress and problem handling activities needs more sophisticated processing of bug repositories. We have achieved this with developed BugAnalsis tool. It communicates directly with bug tracking tool and creates its own database with special functionalities. In particular, it derives and visualizes PHG graphs assuring direct access to timing specifications of each problem. Furthermore, it provides various statistics on states and paths taking into account the number of processed problems, their properties, handling times, etc. (compare Sect. 4).

The BugAnalysis has been designed as java standalone application. In the first layer the graphical user interface (GUI) was implemented. It allows the user to select data source for generation of PHG graph. The second layer represents application logic and data model, which provides methods to interact with the generated graph. The third layer contains data connectors with Bugzilla and text files. Implementation of Bugzilla connector was based on J2Bguzilla library [19], which provides connection to Bugzilla API. Library has been extended to ensure bug history exportation through handling webservice Bug::History.

BugAnalysis tool has been tested using the “open” Bugzilla repositories, in which API interface was turned on. To ensure that our application does not generate too much load for the Bugzilla repository with the same requests, we decided to store exported data in temporary text files with a specific structure. The temporary files can be later used as data source.

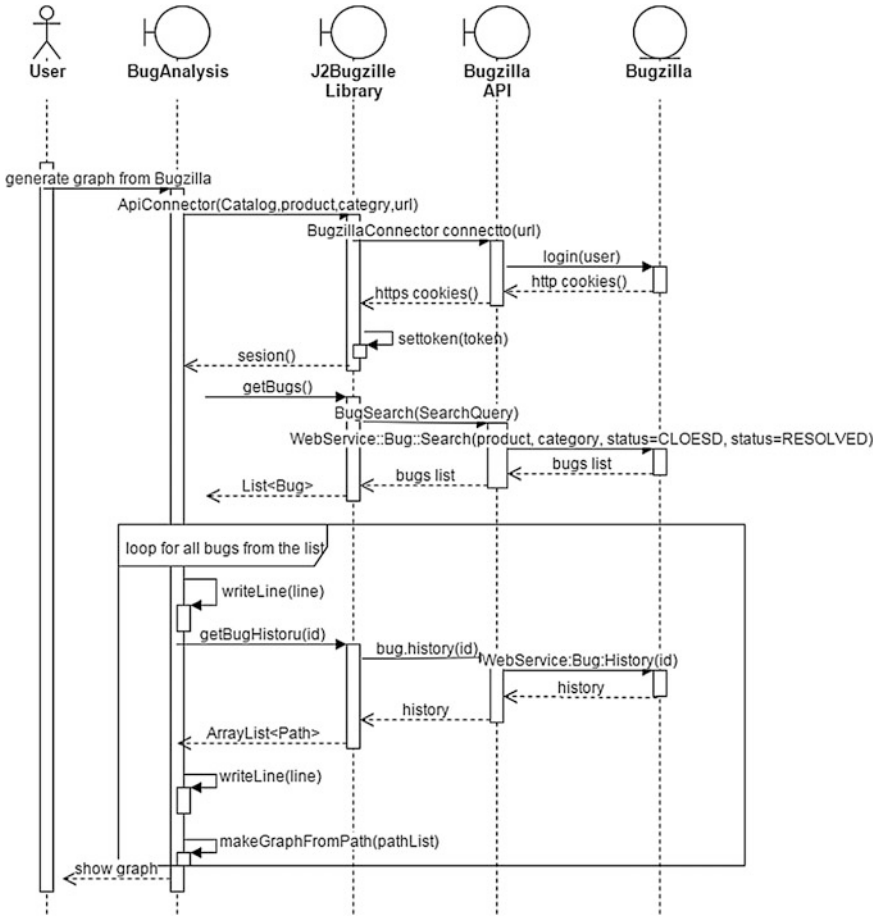


Fig. 1 Sequence diagram of generating PHG graph from Bugzilla

In Fig. 1 we present an UML sequence diagram of generating a PHG graph from Bugzilla repository. This process can be divided into the following steps:

1. The application initiates a session for connection to the URL specified by the user. This step is realized by initiating J2Bugzilla BugzillaConnector object, which is later used to perform the search operation for bug reports. Optionally, while building the session, users may be logged into their accounts (according to provided authorization data.
2. The application performs a search operation with BugzillaConnector object previously initiated. To specify search criteria (e.g. project, states, bug priorities) the BugSearch and the SearchQuery objects from J2Bugzilla are used. These objects use Bugzilla webservice Bug::Search, which returns the bugs objects

represented by an array of hashes (information on bugs and their ids). From the result of the search operation the application builds a list of bugs ids.

3. For each id from the list created in step 2, the application performs getBugHistory operation which launches webservice Bug::History. Webservice returns a hash with bug history. Any changes in the state and the date of implementation of the changes are saved to a temporary file of the application. After receiving history for all bug reports the PHG graph is generated from the temporary file. For the presentation of the graph JUNG library [5] has been integrated with BugAnalysis. It allows us to build a graph with customized vertices and edges and visualize it on the screen.

The generated graph is presented in a separate window which is divided into three main areas. The main frame displays PHG graph. Two additional frames are situated on the top of the window. The first one presents statistics for a selected vertex (state) of the graph, the other one is used for operations on the graph such as filtration and generation of appropriate statistics. The application can generate two output files. The first file contains statistics for states of the graph, like the number of state settings (counted all state settings, including loops), the number of unique state settings (related to bug handling process for unique problems), time statistics of bug presence in the state (minimal, maximal, average, median, the first and third quartiles). The second generated file contains path statistics, such as the number of bugs that followed the path, list of states in the path, time statistics of traversing the path (similar to those related to states). It is also possible to search for long lived bugs and identify states introducing delays in problem handling.

The generated PHG graphs can be configured to cover all problems, problems of specified priority, states and transitions exceeding a specified threshold of correlated problems (reduced graphs). Moreover, it is possible to edit graph view. The implemented tool is easy to use, but it allows performing in-depth analysis of bug handling process that will be presented in the following section of the paper.

4 Experiments

Using BugAnalysis tool we have analyzed PHG graphs for various open source and commercial projects. In particular, we were interested in identifying problem handling paths, their structure, numbers of processed problems and time features. Time features related to time spent in subsequent states or paths (e.g. minimal, maximal, average, median, 1st and 3rd quartiles of their values). For an illustration we give some results for Tomcat 6 project. Figure 2 presents the generated complete PHG graph. The names of states are supplemented with the numbers of problems entered in these states. Graph edges are labeled with the numbers of problems traversing them in the handling process. It is also possible to generate other graphs limited to specified problems (e.g. of specified severity). We can also use other state and edge labeling, e.g. expressed in percentiles of all problems.

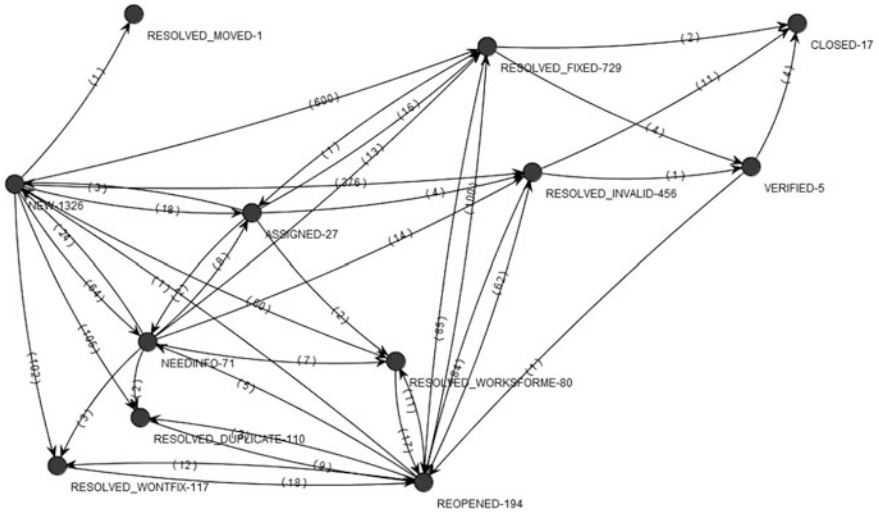


Fig. 2 PHG graph for Tomcat 6 project

An important issue is to generate reduced graph comprising only states and edges with specified throughput. Taking into account states and edges covering 90 % problems the graph of Fig. 2 reduces to 4 states with the following transition list:

NEW(1326)→{(600)RESOLVED_FIXED|(376)RESOLVED_INVALID};
RESOLVED_FIXED(729)→(65)REOPEND; RESOLVED_INVALID(458)→(81) REOPEND
REOPEND(194)→{(100)RESOLVED_FIXED|(82)RESOLVED_INVALID}

Numbers in state suffix brackets denote the number of problems related to the state, numbers in prefixes denote the number of problems traversing specified transitions with →. Multistate branches are specified in {} brackets and separated with |.

Some selected time characteristics are presented in Table 1. They include average, median, 1st and 3rd quartile calculated over problems which entered and exited the considered state. In addition, columns N_U and N specify the numbers of unique and all problems visiting the state. N may slightly exceed N_U because some problems appear more than once in the state due to loops (appearing scarcely). The average problem processing times can exceed significantly the median values,

Table 1 Sample of statistics for Tomcat 6

State	N_u	N	AV	Q1	M	Q3
NEW	1298	1326	41:14:15	0:5:22	5:23:28	51:15:28
REOPENED	159	194	93:12:9	0:2:13	1:13:35	28:20:42
NEEDINFO	65	71	86:6:47	0:1:40	2:5:11	27:11:37

due to problems with big delays. More advanced analysis involves tracing problem handling paths. The most frequently traversed paths (dominating) for Tomcat 6 project are given below with specified number of processed problems (in ‖ brackets) and processing times (Av, M, Q1, Q3—average, median, 1st and 3rd quartile values specified in days:hours:minutes, respectively):

|533| NEW,RESOLVED_FIXED (Av = 104:12:59; Q1 = 5:8:16; M = 26:13:49; Q3 = 84:22:36)
|300| NEW,RESOLVED_INVALID (Av = 20:3:48; Q1 = 0:0:12; M = 0:4:35; Q3 = 1:22:35)
|96| NEW,RESOLVED_DUPLICATE (Av = 98:17:9; Q1 = 0:0:22, M = 1:1:33; Q3 = 27:4:7)
|83| NEW,RESOLVED_WONTFIX (Av = 264:16:40; Q1 = 0:2:7; M = 19:14:57; Q3 = 211:6:3)
|46| NEW,RESOLVED_WORKSFORME (Av = 73:14:41; Q1 = 0:11:37; M = 5:11:54; Q3 = 37:20:47)

In the considered project we have identified 81 distinct paths. They involve from 2 (as the above presented) to 10 states (e.g. NEW, NEEDINFO, RESOLVED_WORKSFORME, REOPENED, RESOLVED_INVALID, REOPENED, NEEDINFO, NEW, NEEDINFO, RESOLVED_WORKSFORME). It is worth noting that longer paths are less populated. We have identified 27, 16 and 5 longer (more than 5 states) paths with the same final states as in the first 3 paths shown above. They covered 108, 300 and 96 problems, respectively. It is also interesting to track problems which did not achieve the terminal state (being in processing), in the considered project we have identified 52 such problems (on 19.11.2015), as compared with all problems (1327) this is a small percentile. However, some of these problems appeared 7 years before (seem to be outdated and could be terminated in some way). Resolved problems achieving states with postfixes: _FIXED (corrected code), _DUPLICATE (the same problem has been processed) and _INVALID (not justified), _WONTFIX (rejected) and _WORKSFORMES (cannot be replayed most probably due to system configuration inconsistencies) constituted about 46.8, 9, 24.8, 6.9 and 3.8 %, respectively. Unfortunately, most resolved problems haven't been verified and closed (poor management process and low responsibility of people involved in the project).

In a similar way we analyzed many open source projects and found that real terminal states (e.g. CLOSED) appeared rarely as opposed to commercial projects were project managers take care on the product quality. Moreover, we can compare different projects (e.g. time spent in different states, or paths) and find the reasons of detected differences, to improve the handling processes.

In the presented statistics we do not differentiate problem severity (priority). In the considered open source projects dominated level 2 of severity. In all Tomcat versions problems with other priorities contributed in total a few up to 7.5 %. In the case of Log4 J project problems with priority 2 and 3 contributed 47.6 and 46.0 %, respectively. This results from lacking well defined priority policy in the projects.

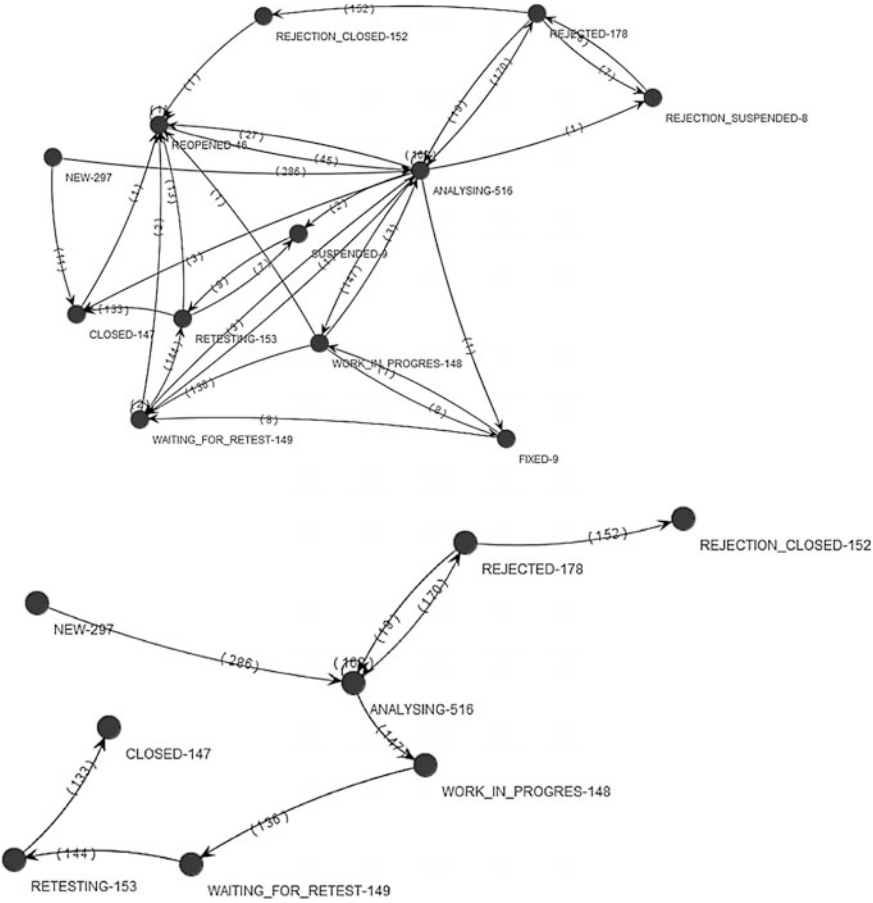


Fig. 3 PHG graphs (complete and reduced to 90 % coverage) for commercial project A

In the case of commercial projects better management and responsibility of project actors are typical. For an illustration in Fig. 3. we give PHG graph for one project A (it comprises 12 states, 47 different paths, path lengths ranged from 4 to 12 states).

The reduced graph corresponding to 90 % of problems comprises 8 states and 8 transitions. As opposed to open source projects practically all problems achieved states CLOSED or REJECTION_CLOSED (problems rejected due to wrong interpretation of specification by the actors). Most problems traversed the following paths:

|64|NEW,ANALYSING,REJECTED,REJECTION_CLOSED (Av = 0:18:16, Q1 = 0:0:16, M = 0:2:25, Q3 = 0:18:22)
|56|NEW,ANALYSING,WORK_IN_PROGRES,WAITING_FOR_RETEST, RETESTING,CLOSED (Av = 2:22:48, Q1 = 0:2:30, M = 1:1:14, Q3 = 2:22:55)

|46|NEW,ANALYSING,ANALYSING,REJECTED,REJECTION_CLOSED (Av = 1:11:49, Q1 = 0:9:48, M = 0:22:20, Q3 = 2:0:6)
|27|NEW,ANALYSING,ANALYSING,WORK_IN_PROGRES, WAITING_FOR_RETEST,RETESTING,CLOSED (Av = 17:15:22, Q1 = 0:5:47, M = 2:9:39, Q3 = 5:14:14)
|10|NEW,ANALYSING,REOPENED,ANALYSING,WORK_IN_PROGRES, WAITING_FOR_RETEST,RETESTING,CLOSED (Av = 3:11:6, Q1 = 1:2:12, M = 2:20:14, Q3 = 3:4:28)

Most problems have been resolved by changes of business and environment (Apache, network connections) configurations. Unfortunately, this was not clearly specified in reports. Code modifications (FIXED) were scarce. Many problems (over 50 %) have been rejected (REJECTION_CLOSED). Relatively frequent loops on state ANALYSING (over 160 iterations) resulted from assignment to non-competent person, and the need of finding other person to resolve it. As opposed to open source projects we observed more diversified priority distribution: P1—54.5 %, P2—35.7 %, P3—7.7 %, P4—2.0 % (P1 the highest priority).

As far as it concerns timing properties in the considered project we have much lower values than in the open source one (compare time parameters related to paths). Moreover, for illustration we give also some selected statistics for states (time parameter specification <Av, Q1, M, Q3> as in Table 1):

NEW: <0:3:48,0:17:0;0:1:30,0:3:0>; ANALYSING: <0:6:9,0:0:12,0:1:22; 0:22:23:> REOPEND: <0:15:45,0:0:5,0:2:32;0:14:17:>; WAITING_FOR_ RETEST: <0:15:13,0:1:22;0:6:41,1:2:59>:

Some long lived bugs have been resolved in about 1 year (in the case of open source projects some bugs exceeded 5 years). It is worth noting that higher priority problems are handled faster in states NEW and ANALYSING. The average values for NEW are 0:0:13, 0:1:42, 0:3:48 for P1, P2 and P3 severity levels, respectively. For the ANALYSING state they range from 7 to 10 h. Similar time ranges have been obtained for another project B from the same company. In the case of project C from another company we had 28 states and related averaged times were typically a fraction of day up to a few days. However, for some state transitions (involving code corrections, improvements) the average time was in the range 10–50 days (maximal values for some problems exceeded 100 days). For other projects in this company a few problems needed more than 1 year for resolution (bugs in purchased components).

5 Conclusion

The performed study confirmed that bug repositories provide a lot of useful information which can characterize development and maintenance processes. In practice, tracing and exploring the collected data is a cumbersome process. This process is significantly simplified by the introduced analysis tool. It is worth

mentioning that analysis time overhead is linear with the number of bug reports. An important issue is the capability of presenting complete problem handling graphs (PHGs) as well as their reduced and aspect-oriented forms. Combining these graphs with derived statistical data is helpful in managing and improving software life cycle processes. In particular, we can identify critical problems (long lived) or states and find their causes.

Further research is targeted at extending the scope of analysis to study fairness of assigning bugs to developers, correlation of handling times with software complexity, accuracy of bug localization, etc. (compare [6]). However, such analysis needs more detailed repositories.

We are grateful to P. Janczarek for his help in collecting data for the tests.

References

1. Cinque, M. et al.: On the impact of debugging on software reliability growth analysis: a case study. In: Proceedings of ICCSA Conference, Part V, LNCS, vol. 8583, pp. 461–475. Springer (2014)
2. Janczarek, P., Sosnowski, J.: Monitoring software development and usage. Przegląd Elektrotechniczny. Sigma NOT R. **90**(2), 117–120 (2014)
3. Janczarek, P., Sosnowski, J.: Investigating software testing and maintenance reports: case study. Inf. Softw. Technol. **58**, 272–288 (2015)
4. Janczarek, P., Sosnowski, J.: Managing complex software projects. Inf. Syst. Manag. (WULS Press, Warsaw) **4**(3), 171–182 (2015)
5. JUNG (the Java Universal Net-work/Graph Framework)—http://jung.sourceforge.net/
6. Madeyski, L., Jureczko, M.: Which process metrics can significantly improve defect prediction models? An empirical study. Softw. Qual. J. **23**, 393–422 (2015)
7. Messquida, A.-L., Mas, A.: A project management improving program according to ISO/IEC 29110 and PMBOK. J. Softw. Evol. Process 846–854 (2014)
8. Natarajan, R.: Top 10 Open Source Bug Tracking Systems (2010). http://www.thegeekstuff.com/2010/08/bug-tracking-system/
9. Ogasawara, H., Kusanagi, T., Aizawa, M.: Proposal and practice of software process improvement history since 2000. J. Softw. Evol. Process 521–529 (2014)
10. Petersen, K.: A palette of lean indicators to detect waste in software maintenance: a case study. In: Wohlin, C. (ed.) XP 2012, LNBIP vol. 111, pp. 108–122. Springer, Berlin (2012)
11. Saha, R.P., Kurshid, S., Perry, D.E.: Understanding the triaging and fixing processes of long lived bugs. Inf. Softw. Technol. **65**, 114–128 (2015)
12. Santana, F., et al.: XFlow: an extensible tool for empirical analysis of software systems evolution. In: Proceedings of 8th Experimental Software Engineering Latin American Workshop ESELAW, pp. 57–66 (2011)
13. Sommerville, I.: Software Engineering, 9th edn. Pearson, Boston (2011)
14. Sosnowski, J., Sabak, J.: Software reliability analysis in designing database oriented applications. In: Proceedings of the 27th Euromicro Conference, pp. 166–173, IEEE Comp. Society (2001)
15. Zhang, W., Wang, S., Wang, Q.: KSAP: an approach to bug report assignment using KNN search and heterogeneous proximity. Inf. Softw. Technol. **70**, 68–84 (2016)
16. Zhou, J., Zhang, H., Lo, D.: Where should the bugs be fixed? More accurate information retrieval-based bug localization based on bug reports. In: Proceedings of IEEE ICSEE Conference, pp. 14–24 (2012)

17. https://www.bugzilla.org/features/
18. https://en.wikipedia.org/wiki/Comparison_of_issue-tracking_systems
19. http://code.google.com/p/j2bugzilla/stronaprojektuJ2Bugzilla

Modification of Neural Network Tsang-Wang in Algorithm for CAD of Complex Systems with Higher Degree of Dependability

Mieczyslaw Drabowski

Abstract The paper includes a proposal of a new algorithm for Computer Aided Design (CAD) of complex system with higher degree of dependability. Optimal scheduling of tasks and optimal resources partition are fundamental problems in this algorithm. Presented the CAD algorithm, based on neural networks, may have a practical application in developing tools for rapid prototyping of such systems.

Keywords Complex system · Scheduling · Partition · Allocation · Dependable · Optimization · Neural algorithm · Tsang-Wang networks · CAD tools

1 Introduction

The aim of computer aided design of complex systems (i.e. which contain multiprocessors, additional resources and multitasking) is to find an optimum solution consistent with the requirements and constraints enforced by the given specification of the system. The following criteria of optimality are considered: costs of system implementation, its operating speed and power consumption.

The identification and partitioning of resources between various implementation techniques is the basic matter of automatic design. Such partitioning is significant, because every complex system must be realized as result of hardware implementation for its certain tasks. Additionally scheduling problems are one of the most significant issues occurring in design of operating procedures responsible for controlling the allocations of tasks and resources in complex systems.

In the new design methods—in the par-synthesis—which were presented in [1], the software and hardware components are developed together and parallel—otherwise, than so far (for example [2])—and then coherent connected together, which of the final solution decreased the costs and increased the speed. The resources

M. Drabowski (✉)
Faculty of Electrical and Computer Engineering,
Cracow University of Technology, Kraków, Poland
e-mail: drabowski@pk.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_11

distribution is to specify, what hardware and software are in system and to allocate theirs to specific tasks, before designing execution details.

Another important issue that occurs in designing complex systems is assuring their fault-free operation. Such designing concentrates on developing dependable and fault-tolerant architectures and constructing dedicated operating procedures for them [3]. In this system an appropriate strategy of self-testing during regular exploitation must be provided. The general model and new concept of parallel to tasks scheduling and resources partition for complex systems with higher degree of dependability was presented in [4]. We proposed the following schematic diagram of a coherent process of fault tolerant systems synthesis—Fig. 1.

The suggested parallel analysis consists of the following steps:

1. specification of requirements for the system,
2. specification of tasks,
3. assuming the initial values of resource set,
4. defining testing tasks and the structure of system, testing strategy selection,
5. scheduling of tasks,
6. evaluating the operating speed and system cost, etc., multi-criteria optimization,
7. the evaluation should be followed by a modification of the resource set, a new system partitioning into hardware and software parts and an update of test tasks and test structure (step 4).

Modeling fault tolerant systems consists of resource identification and tasks scheduling problems that are both hard NP-complete [5]. Algorithms for solving such problems are usually based on heuristic approaches.

The objective of the paper [6] was to presented of hybrid approach to the problem of fault tolerant systems design, i.e. a parallel solution to tasks scheduling and resource assignment problems. We suggested in this paper meta-heuristic and

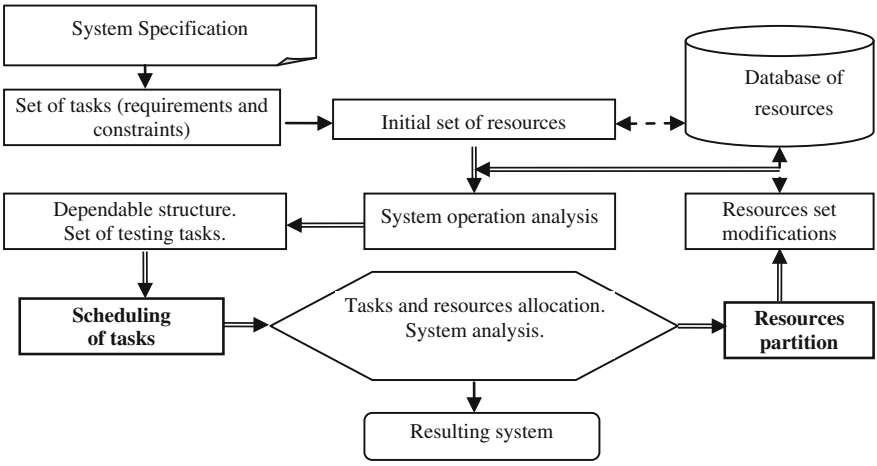


Fig. 1 The process parallel design (the par-synthesis) of dependable complex system

hybrid algorithm: evolutionary with simulated annealing, in which there are Boltzmann tournaments.

We present, in this paper, other the meta-heuristic algorithm, based for the adaptation a neural network (the approach similar to solution proposed by Tsang and Wang [7] for constraints satisfactions problem) what has been applied for resolving problems of resource allocation and task scheduling and also for a coherent solution to these problems.

2 The Neural Method for Coherent Synthesis of Computer System

2.1 Neural Network Model

As already mentioned, the starting point for defining the neural network model for coherent and parallel solving the problems of task scheduling and resource allocation are the assumptions for the constraint satisfaction problem [8]. CSP is the optimization problem which contains a certain set of variables, sets of their possible values and constraints forced on the values of these variables. On the basis of this problem assumption a network model of the following features is suggested:

- A neural network consists of components; each of them corresponds to another variable.
- Each component contains such number of neurons which equals the number of possible values of each variable.
- Assigning a specified value to a variable is the process of switching on a relevant neuron (neurons) and switching off the remaining ones in the component corresponding to this variable.
- Switching on a neuron means assigning the value "***1***" to its output.
- Switching off a neuron means assigning the "***0***" to its output.
- Constraints to the network are introduced by adding a negative weight connection between neurons ('−***1***'), symbolizing the variable values that cannot occur simultaneously.
- In the network there are additional neurons "the ones" that are switched on.

Each neuron has its own table of connections and each connection contains its weight and the indicator for the connected neuron. A characteristic feature of the network is the diversity of connections between neurons, but these never applied to all neurons. It is a consequence of the fact that connections between neurons exist only when some constraints are imposed. The constraints existing in the discussed network model may be of the following types:

(1) Resource.
(2) Time.
(3) Schedule.

The method of constraints implementation shall be discussed upon examples.

Example 1:

Such net (Fig. 2) blocks solution, in which $Z_1 = 1$ as well as $Z_2 = 2$ or $Z_3 = 3$ as well as $Z_4 = 2$.

Example 2:

Let us have two operations with unit execution times. The operation Z_1 arrives at the system in time t = 1 and it is to be executed before the expiry of time t = 4. The operation Z_2 arrives in time t = 1 and may be executed after the completion of operation Z_1. A fragment of the net for his case including all the connections is shown by Fig. 3.

Neuron "one" ('1')—a special neuron switched on permanently—is responsible for time constraints. Introducing connections between such neuron and the relevant network neurons excludes a possibility of switching them on when searching for the solution. Task Z_1 cannot be scheduled in moment ***0*** and moment ***4***, which corresponds to the assumption that this task arrives at the system at moment *1* and must be performed before moment ***4***. Analogical process applies to operation Z_2. The

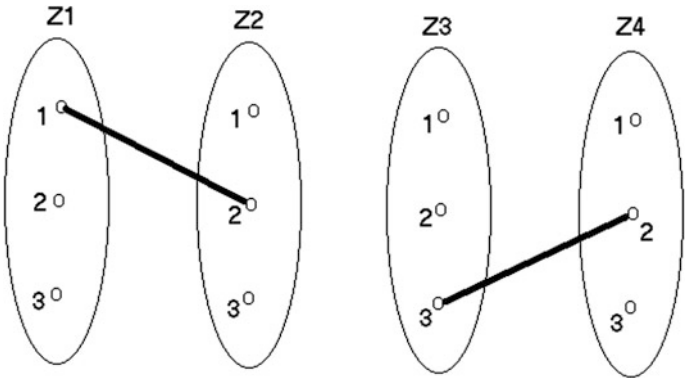


Fig. 2 The example 1 of constraints

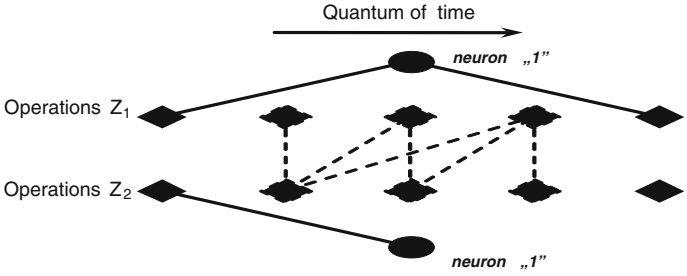


Fig. 3 The example 2 of constraints

sequence constraints are executed by the connections between the network neurons. The figure shows (with dotted line) all the connections making the performance of task Z_2 before task $\boldsymbol{Z_1}$ impossible.

2.2 The Idea of the Algorithm

After entering the input data (the system specification), the algorithm constructs a neural network, the structure of which and the number of neurons composing it, depend upon the size and complexity of the instance of problem. We will name the part of the net allocated to this task—an area. Constraints are introduced to the network by the execution of connections, occurring only between the neurons corresponding to the values of variables which cannot occur simultaneously.

The operation of the algorithm is the process of switching on appropriate neurons in each domain of network in order to satisfy the constraints imposed by the input data.

The algorithm course is as follows:

1. Allocating random values to consecutive variables.
2. Network relaxation:
   - Calculating the weighted sum of all neurons inputs.
   - Switching on the neuron with the highest input value.
   - Return to relaxation or—if there are no changes—exit from relaxation.
3. If there are connections (constraints) between the neurons that are switched on, each weight between two switched on neurons is decreased by 1 and there is a return to relaxation.

The algorithm starts from allocating weight '$\boldsymbol{-1}$' to all connections and then the start solution is generated. It is created by giving random values to the subsequent variables. This process takes place in a certain way: for each task i.e. in each area of the net such number of neurons is switched on as it is necessary for a certain task to be completed. The remaining, in the part which is responsible for its performance, neurons are being switched off. In the obtained result there are many contradictions, specified by switching on the neurons where the connections exist.

Therefore, the next step of the algorithm is the relaxation process, the objective of which is to "satisfy" the maximum numbers of limitations (backtracking) [9]. The objective is to obtain the result where the number of situations, where two switched on neurons of negative weight connection between them is the lowest. While switching on neurons with the biggest value at the start, in each area three instances may happen:

- If there is one neuron of the biggest value in the area, it is switched ***on***; the remaining ones are switched ***off***.

- If there are more neurons, among which there is a previously switched one, there is no change and it remains switched ***on***.
- If there are more neurons, but there is no-one previously switched **on**, one of them is switched on randomly, the remaining ones are switched ***off***.

A relaxation process finishes when the subsequent step does not bring any change and if all the requirements are met—the neurons between which a connection exist are not switched on—the right solution is found. If it is not still the case, it means that the algorithm found the local minimum and then the weight of each connection between two switched on neurons is decreased by "***1***" while its absolute value is being increased. It causes an increase in 'interaction force' of this constraint which decreases the chance of switching ***on*** the same neurons in a relaxation process where we return in order to find the right solution.

After a certain number of iterations the network should consider all the constraints—providing that there is the right solution, it should be found. Another factor is worth pointing out: in a relaxation process such an instance may occur where changes always happen. Then, this process might never be completed. Then a problem is solved in such a way that relaxation is interrupted after a certain number of calls.

Search for a solution by algorithm consists of two stages. At the first one, which is described by the above presented algorithm, some activities are performed which lead to finding the right solution for the given specification. After finding such a solution, in consequence of purpose function optimization there is a change of values for a certain criterion—in this case, decrease—then, the subsequent search for the right solution occur. In this case the search aims at a solution which possesses bigger constraints as the criteria value is sharper. Two criteria are taken into consideration for which a solution is being searched. It may be a cost function—where at the given time criterion, we search for the cheapest solution, or time function—where at the given cost criterion, we search for the quickest solution. Thus, the run of the algorithm is to seek a solution for smaller and smaller value of a selected criterion. However, if the algorithm cannot find the right solution for the recently modified criteria value of the algorithm, it returns to the previous criteria value for which it has found the right solution and modifies it by a smaller value.

For instance, if an algorithm has found the right solution for cost criterion which is e.g. 10, and it cannot find it for cost criteria which are 9, it tries to find a solution for cost 9.5 etc. In this way the program never finishes work, but all the time it tries to find a better solution in sense of a certain criterion.

In case of time criterion minimization, optimization goes at two planes. At the first one, subsequent neurons of the right side in task part of the network are connected to the neurons "***one***", in this way fewer and fewer quanta is available for the algorithm of task scheduling which causes moving a critical line to the left and at the same time its diminishing. However, at the second, an individual quantum of time is being diminished; at each step an individual neuron will mean a smaller and smaller time passage.

3 The Algorithm Description

On the basis of a general model we shall present now a neural network for solving submitted problems. In order to achieve this, we divide the time axis into parts, creating time quanta which are single neurons.

The task part:
Each area corresponds to one task (Fig. 4). For further area, the best possible setting for the task is selected. Which setting 'wins' at the given stage and in the given area—this shall be determined by the sum of neuron outputs in the setting, i.e. the one that introduces the smaller number of contradictions. Moreover, it is checked if among the found set of the best solutions there is no previous one, then it is left.

A neuron at the [***I***, ***k***] position corresponds to the presence of '***i***' task on the processor at the '***k***' moment. Between these neurons there are suitable inhibitory connections (−***1.0***). If, for example, task 1 must be performed before task 2, for all the neuron pairs.

[***1***, ***k***], [***2***, ***m***] there are inhibitory connections (denoting contradictions), if ***k*** > = ***m*** and if task 8 occurs in the system at moment 2, "***one***" neuron is permanently connected to neurons [***8***, ***0***] ***and*** [***1***, ***8***] (neuron which has 1.0 at the start which is permanently contradictory) and guarantees that in the final solution there is no quantum at moment 0 or 1.

We also take critical lines into account, which stand for time constraints that cannot be exceeded by any ***allocated*** tasks—connecting '***one***' will apply to the neurons of the right side of the network outside the critical line.

The resource part:
Before selecting the quanta positions in the areas, algorithm has to calculate inputs for all the neurons. The neurons of the resource part are also connected to these inputs, as the number and the remaining places in resources have an impact on the setting which is going to "win" at a certain stage of computation. Thus, before an algorithm sets an exact task, it calculates the value of neuron inputs in resource part. The [***r***, ***I***, ***k***] neuron is switched on if at '***k***' moment the resource '***r***' is overloaded (too many tasks are using t), or it is not overloaded, but setting the task of part '***i***' at the moment defined by '***k***' would result in overloading.

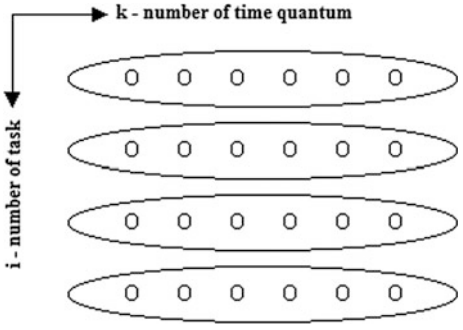


Fig. 4 The task part in the algorithm

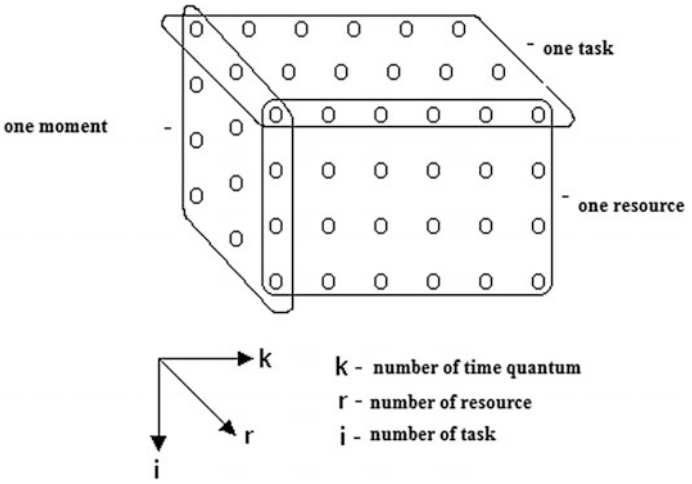


Fig. 5 The resource part in the algorithm

Neurons in resource part (Fig. 5) respond by their possible connection, resource overloaded, if part of the task were set and at moment '***k***'; therefore, neurons of resource part are connected to task inputs.

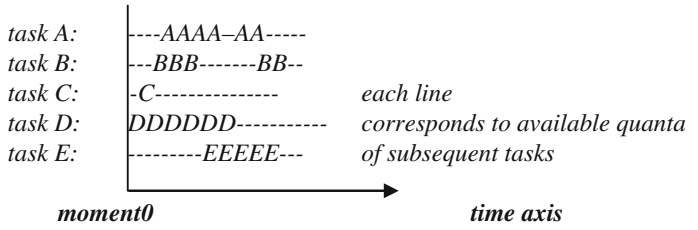
When in the resource part the neuron "'***r***' ***resource overload***' is switched ***on***, as task '***i***' is set at moment '***k***'", its signal (***1.0***) is transferred by weight (−***1.0***) to the neuron existing in the task part, which causes the negative input impulse (−1.0 * 1.0) at the input which results in a contradiction.

In other words—it "disturbs" function '***compute_in***' to set the task and at moment "***k***". Thus, in the network there are subsequent illegal situations implemented (constraints).

Each neuron [***r***, ***I***, ***k***] of the resource part is connected with neuron [***I***, ***k***] from the task part, so a possibility of task existence at a given moment with concurrent resource overloading is 'inhibited'.

Example of Algorithm

Let us assume that there are five tasks *A*, *B*, *C*, *D*, *E*. Task part works as follows (a letter means a neuron switched on, sign '-' means a switched off neuron):



These tasks should be allocated to a certain number of processors, so that one only task would be performed on one processor at an exact moment:

1. Algorithm allocates (at moment 0) fragment *DDDDDD*, adds a new processor (the first) and allocates on it:

   DDDDDD----------

2. Allocation—*C*: for this moment (1) there is no place on the first processor, so algorithm adds the next processor and allocates an operation:

   DDDDDD----------
   -C---------------

3. Allocation *BBB*: there is place on the second processor:

   DDDDDD----------
   -C-BBB----------

4. Allocation *AAAA:* there is no place at quantum 4—algorithm adds the third processor and allocates:

   DDDDDD----------
   -C-BBB----------
   ----AAAA---------

5. Allocation *EEEEE: there is place on the first processor*:

   DDDDDD---EEEEE---
   -C-BBB----------
   ----AAAA---------

6. Allocation *AA* there is place on the second processor

   DDDDDD---EEEEE---
   -C-BBB----AA------
   ----AAAA---------

7. Allocation *BB*: there is place on the second processor:

   DDDDDD---EEEEE---
   -C-BBB----AA-BB—
   ----AAAA---------

The result of the tasks on the processors is as follows:

P1:DDDDDD---EEEEE---
P2:-C-BBB----AA-BB--
P3: ----AAAA---------

4 Computational Experiments

The algorithm was tested by taking into account the parameters of different instances of problems, among others:

- The number and type of processors available in the database of resources.
- The number of tasks.
- The preemptability of tasks.
- Order and resources constraints.

4.1 The Influence of Task Preemptability on Computation

All tests were carried out for independent tasks, without any additional resources, the number of processors in the pool is 10, and the maximum number of processors in the system is 5, including the cost of memory. Constraints: maximum cost is 50 and maximum time 50. The parameters of algorithm: maximum time of calculation step is 1000; the number of time units is 100 (Table 1).

Experiment results show that the ***preemptability*** of tasks does not influence the neural network size, however, results are better, especially when it comes to cost factor. In the specification of input tasks there were no tasks, whose execution time would be lengthier than others, thus differences are small. The cost is slightly lower than in the case of divisible tasks, because—as a rule—they are allocated on a smaller number of processors with comparable C_{max}.

4.2 The Influence of Additional Resources on Computation

All tests were carried out for independent and indivisible tasks, the number of processors in the pool is 10, and the maximum number of processors in the system is 5, including the cost of memory. Constraints: maximum cost is 50 and maximum time 50. All resources have the number of units equal to 3.

Results shown in table Table 2 prove that the existence of additional resources significantly influences network size. Increasing the number of additional resources impacts scheduling time and cost, however with high number of tasks the quality of

Table 1 The influence of task's *preemptability* on computation

Number of tasks	Preemptability	C_{max}	Cost	Number of neurons	Number of connections
10	No	0.705	17.80	6,255	56,121
10	Yes	0.704	16.45	6,255	56,121
20	No	1.229	22.35	12,285	212,721
20	Yes	1.219	21.75	12,285	212,721
30	No	1.900	24.50	18,345	470,321
30	Yes	1.843	18.25	18,345	470,321
40	No	2.464	18.85	24,405	828,921
40	Yes	2.384	22	24,405	828,921
50	No	3.005	25.75	30,465	1,288,521
50	Yes	3.065	19.45	30,465	1,288,521

Table 2 The influence of additional resources on computation

Number of tasks	Number of additional resources	C_{max}	Cost	Number of neurons	Number of connections
10	1	1.519	8.4	7,336	62,484
10	2	1.509	13.1	,8447	69,756
10	5	2.338	7.25	11,780	89,754
20	1	2.839	17.7	14,406	225,447
20	2	3.145	12.6	16,527	240,900
20	5	4.096	8.65	22,890	281,805
30	1	4.434	14.1	21,476	490,319
30	2	4.677	18.5	24,607	509,408
30	5	6.569	12.4	34,000	571,220
50	1	8.058	13.8	35,616	1,322,154
50	2	9.786	21.1	40,767	1,354,878
50	5	13.03	13.6	56,220	1,457,595

results, especially when it comes to $\boldsymbol{C_{max}}$, worsens (Gantt graph shows more and more "gaps"—stoppages). A positive result is the fact that the bigger the number of additional resources, the better the result for both criteria jointly.

4.3 The Synthesis of Systems with High Degree Dependability—The Comparing Results Obtained by Algorithms: Neural and Evolutionary

In the presented comparative tests obtained results from the calculations of both algorithms: hybrid, an evolutionary with simulated annealing, and neural, which was discussed in this paper. The results are shown in the charts below, Fig. 6.

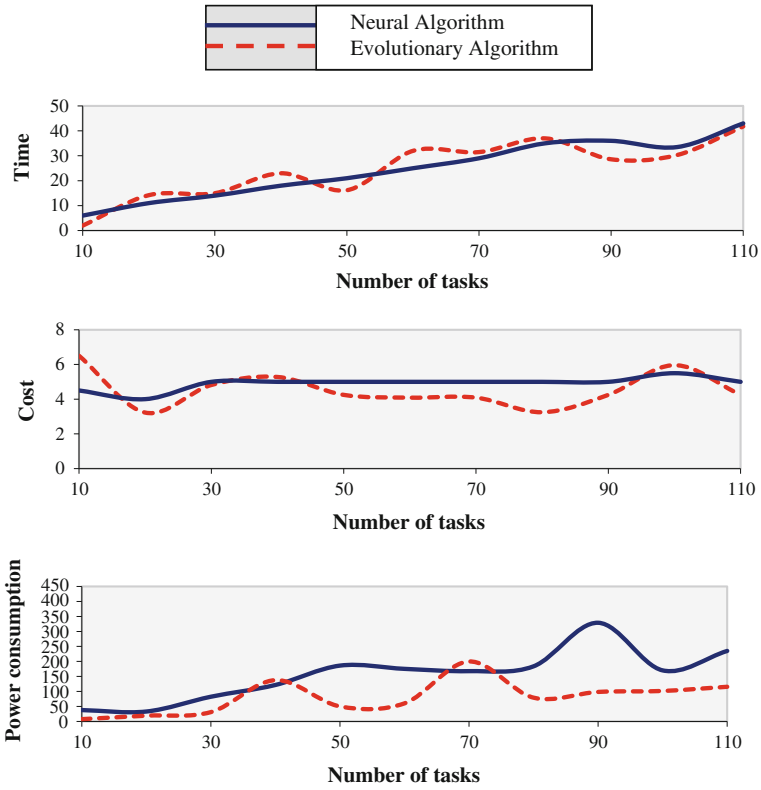


Fig. 6 Comparing the computational results: Hybrid, Evolutionary Algorithm [6] and Neural Algorithm for optimization criteria: speed, cost, power consumption of complex system

The neural algorithm generates a more stable results, especially for the criterion of minimizing power consumption, an evolutionary algorithm gives better results for the criterion of minimizing cost and power consumption and neural algorithm for the criterion of maximizing speed, but not for all input data.

5 Conclusions

This paper is about the problems of parallel design of complex systems with high degree of dependability. Such a design is carried out on a high level of abstraction in which system specification consists of a set of tasks, which should be implemented by a series of resources and these are listed in the database (container, pool, or a catalogues) and are available (exist or can be created). Resources possess certain constraints and characteristics, including speed, power, cost and dependability parameters. Thus such a design concerns complex systems of the following

type—resources, tasks and optimization criterions and the problems of resource partitioning (selection) as well as scheduling (sequencing) of tasks performed on these resources are determined on this level [10]. Optimization of afore-mentioned design actions occurs on the same high level.

In the paper one proposed the so called artificial intelligence methods for CAD. Obviously these methods were chosen out of many and proposed adaptations of these methods and can be different.

Among presented results of computational experiments the new solutions were obtained with the neural algorithm with Tsang, Wang networks.

The issues for other methods are now studied.

References

1. Drabowski, M.: Par-synthesis of multiprocessors parallel systems. Int. J. Comput. Sci. Netw. Secur. **8**(10), 90–96 (2008)
2. Dick, R.P., Jha, N.K.: COWLS: hardware-software co-synthesis of distributed wireless low-power client-server systems. IEEE Trans. Comput. Aided Des. Integr. Circ. Syst. **23**(1), 2–16 (2004)
3. Yhang, Z., Dick, R. Chakrabarty, A.: Energy-aware deterministic fault tolerance in distributed real-time embedded systems. In: 41st Proceedings of Design Automation Conference, pp. 550–555, Anaheim, California (2004)
4. Drabowski, M., Wantuch, E.: Deterministic schedule of task in multiprocessor computer systems with higher degree of dependability. In: Zamojski, W., Mazurkiewicz, J., Sugier, J., Walkowiak, T., Kacprzyk, J. (eds.) Advances in Intelligent Systems and Computing, Volume 286, Proceedings of the Ninth International Conference on Dependability and Complex Systems DepCos-RELCOMEX, pp. 165–175, Springer (2014)
5. Garey, M., Johnson, D.: Computers and intractability: a guide to the theory of NP-completeness. Freeman, San Francisco (1979)
6. Drabowski, M.: Boltzmann tournaments in evolutionary algorithm for CAD of complex systems with higher degree of dependability. In: Zamojski, W., Mazurkiewicz, J., Sugier, J., Walkowiak, T., Kacprzyk, J. (eds.) Advances in Intelligent Systems and Computing, Volume 365, of the Tenth International Conference on Dependability and Complex Systems DepCos-RELCOMEX, pp. 141–152, Springer (2015)
7. Wang, C.J., Tsang, E.P.K.: Solving constraint satisfaction problems using neural-networks. In: IEEE Second International Conference on Artificial Neural Networks (1991)
8. Dechter, R., Pearl, J.: Network-based heuristic for constraint satisfaction problems. Artif. Intell. **34**, 1–38 (1988)
9. Tsang, E.: Foundations of Constraint Satisfaction. Academic Press, Essex (1993)
10. Błażewicz, J., Drabowski, M., Węglarz, J.: Scheduling multiprocessor tasks to minimize schedule length. IEEE Trans. Comput. **C-35**(5), 389–393 (1986)

Simulation and Experimental Analysis of Quality Control of Vehicle Brake Systems Using Flat Plate Tester

A.I. Fedotov and M. Młyńczak

Abstract Paper describes simulation analysis of errors that arise in the control process of brake systems for vehicles using flat testers. Technical state of brake systems and their performance in operation are important aspects of safety. Method of numerical simulation of dynamic systems is described concerning vehicle braking capability. It is proposed dynamic model of the tire on the flat plate brake tester. Simulation model takes into account all factors influencing measurement error resulting from testing method. Methodology of brake testing on flat testers and measurement system is analyzed. There are discussed measurement errors observed on flat testers related to the dynamic properties of the tire, load on the wheel, mass of tester plate as well as speed and braking time and force applied by a driver. Calculations are illustrated with graphs. Square correlation coefficient of assumed functions is not less than 0.95.

Keywords Flat braking testers · Brake testing · Measurement error

1 Introduction

Vehicle brake testing is an obligatory periodic maintenance carried out in workshops equipped in special devices measuring brake parameters [2, 5]. Brake tests use two ideas based on rolling drums or flat plates suspended elastically [5–7]. Rolling drums are more popular and have more accurate measurements, though flat testers are more universal as can be used to test shock absorbers and suspension [6, 7]. A vehicle moving on plates engages brakes and causes forces and moments proportional to

A.I. Fedotov (✉)
Irkutsk State Technical University, Irkutsk, Russian Federation
e-mail: fai@istu.edu

M. Młyńczak
Faculty of Mechanical Engineering, Wroclaw University of Technology, Wroclaw, Poland
e-mail: marek.mlynczak@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_12

brake efficiency. Flat plate brake testers have usually two plates for wheel of the same axis with sensors measuring longitudinal and vertical forces. Computer application processes data and give an assessment about vehicle brake conditions. Problem with measurement accuracy appears due to wheels positioning on the plates, what is analyzed in the first part of the paper and on dynamic parameters of the system wheel—tester plate. Results depend on many dynamic parameters of braking process. Paper presents simulation approach to analysis of influence selected parameters on brake test results. The problem is concern because of its importance to the road safety. Brake testing is one of the elements of the conditional preventive maintenance. Safety depends in that case on: brake performance measured as ratio of total braking forces over vehicle weight, uniformity of braking forces of left and right wheel and dependence of braking force from the pressing force on the pedal. All that forces should bring a vehicle to stop at predefined distance and provide straightforward path of the move while braking. Brake systems are repairable objects, though not many accidents are caused by their failure [9]. Brake system was a cause of 18 out of over 40,000 road accidents what is 10^{-4}. High influence of that system on safety and strong requirements tested periodically makes those systems as highly reliable.

2 Basic Parameters of Flat Brake Testers

Flat brake tester consists of two plates to run onto it by two wheel of the same axis of a vehicle [2, 5–7]. Figure 1 shows a diagram of flat plate tester with the left—2 and right—3 plates with wheels of the car. The wheel may be located to the left or to the right side of the plate axis of symmetry. The best location is exactly on that axis but it is difficult to perform in real conditions, therefore it is observed distance

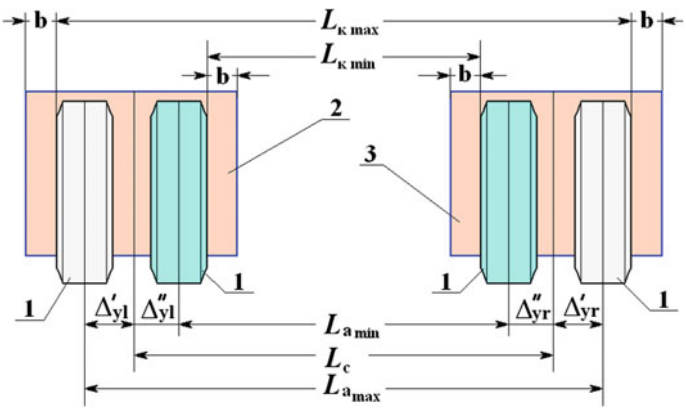


Fig. 1 Scheme for the geometric dimensions flat tester: *1* car wheel, *2 left* plate of the tester, *3 right* plate of the tester

between the plate axis of symmetry and current position of tire axis of symmetry, described as: Δ_{yl} and Δ_{yr} for left and right wheel respectively.

In order to determine the width of the pads and the distance between them it is necessary to know the following geometric dimensions:

1. $L_{\kappa\ max}$—the maximum distance between the outer sides of the car wheels having broad wheel track,
2. $L_{\kappa\ min}$—the minimum distance between the inner sides of the car wheels having narrow wheel track,
3. b—side margin to guarantee position of wheels on the surface of tester plate,
4. L_c—distance between the axes of symmetry of tester plates.

3 Computer Simulation of Flat Brake Tester Dynamics During Measurement Process

To analyze the process of interaction of the vehicle wheel with braking flat tester was drawn a scheme shown in Fig. 2. The model allows for analytical study of dynamic processes from the moment of run onto tester at point A until it stops. Vehicle of the weight M moves under the force of inertia F_{jx}^{M}. In the point of tire contact with the supporting surface reaction R_x arises. The scheme allows us to

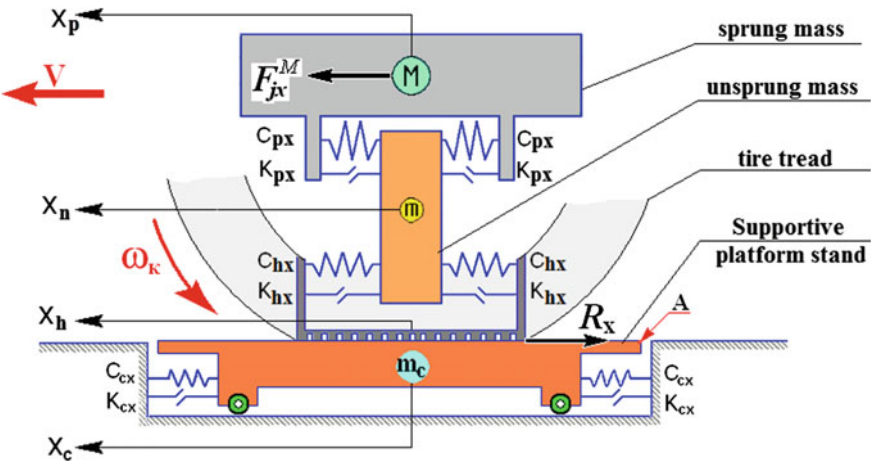


Fig. 2 Scheme of the interaction of a vehicle wheel braking pads with tester where: M part of the sprung mass at the wheel of the car, m unsprung weight, m_{sh} the mass of the tire tread contact patch; m_c mass of the plate tester; K_{px}, K_{hx}, K_{cx} damping coefficients, respectively, suspension, tires and tester [N s/m]; C_{px}, C_{hx}, C_{cx} respectively suspension stiffness, tires and tester [N/m], X_p, X_n, X_h, X_c coordinates of longitudinal displacement, respectively of the sprung mass (M), unsprung mass (m), the mass of the tire tread at the contact patch (m_h) and the weight of the tester plate (m_c)

analyze the influence of the longitudinal oscillations of the masses of the car and flat tester by the uncertainty of measurement of brake forces.

To calculate the vehicle deceleration we write the differential equation of its dynamic equilibrium in the general form (1):

$$\frac{d^2X}{dt^2} = \frac{\sum_{i=1}^{n} R_x}{(M + m \cdot n)} \tag{1}$$

where: n—number of wheels.

Dynamic equilibrium equation of the braking wheels can be written as (2):

$$\begin{cases} M\frac{d^2X_p}{dt^2} = C_{px}(X_{p2} - X_p) - C_{px}(X_p - X_{p1}) + K_{px}(\dot{X}_p - \dot{X}_{p2}) + K_{px}(\dot{X}_{p1} - \dot{X}_p); \\ m\frac{d^2X_n}{dt^2} = C_{hx}(X_{n2} - X_n) - C_{hx}(X_n - X_{n1}) + K_{hx}(\dot{X}_h - \dot{X}_{h2}) + K_{hx}(\dot{X}_{n1} - \dot{X}_n) \\ \qquad - \left[C_{px}(X_{p2} - X_p) - C_{px}(X_p - X_{p1}) + K_{px}(\dot{X}_p - \dot{X}_{p2}) + K_{px}(\dot{X}_{p1} - \dot{X}_p)\right]; \\ m_h\frac{d^2X_h}{dt^2} = C_{hx}(X_n - X_{n2}) - C_{hx}(X_{n1} - X_n) + K_{hx}(\dot{X}_{n2} - \dot{X}_n) + K_{nx}(\dot{X}_n - \dot{X}_{n1}) - R_x; \\ m_c\frac{d^2X_c}{dt^2} = C_{cx}(X_{c2} - X_c) - C_{cx}(X_c - X_{c1}) + K_{cx}(\dot{X}_c - \dot{X}_{c2}) + K_{cx}(\dot{X}_{c1} - \dot{X}_c) - R_x. \end{cases} \tag{2}$$

To relate the resulting equations system (3) with tangential reaction R_x we use a dynamic model of a vehicle wheel braking process [4] based on the characteristics of the stationary characteristics of elastic tires and the normalized slip function [8]:

$$f(s) = \sin\{A \cdot arctg(B \cdot s)\} \tag{3}$$

Dynamic equilibrium equation of the braking wheels can be written as (4):

$$\frac{d\omega_k}{dt} = \frac{R_x \cdot r_{ko} - M_t - M_f}{J_k} \tag{4}$$

where: M_t—current value of the braking torque supplied to the wheel [N m], M_f—moment of resistance to rolling wheel [N m], r_{ko}—the radius of the rolling wheels in the braking mode [m], J_k—wheel moment of inertia [kg m^2].

The current values of brake torque M_t applied to the wheel will be given in the form of a linear relationship (5):

$$M_{ti} = M_{t_{i-1}} + \frac{dM_t}{dt} \cdot \Delta t \tag{5}$$

where: $\frac{dM_t}{dt}$ is the rate of increase of the braking torque [N m/s].

The braking force (tangential reaction R_x) fulfils the formula (6) [4, 8]:

$$R_x = R_z \cdot \varphi_{max} \sin\{A \cdot arctg(B \cdot s)\}[\mathrm{N}] \tag{6}$$

where: φ_{max}—the maximum value of the friction coefficient, A, B—the stationary slip coefficients, S—tire slippage relative to the tester area, R_z—normal reaction at the contact tire patch [N].

It is obvious, that normal reaction R_z during vehicle deceleration changes. To determine reaction R_z the scheme shown in Fig. 3 is proposed. On that basis mathematical description of the process of mass vibrations of the car and tester is made.

To move the masses M, m and m_c along the OZ axis, a system of dynamic equilibrium equations, which is solved with respect to the higher derivatives, is proposed (7):

$$\begin{cases} M\frac{d^2Z_1}{dt^2} = C_n(Z_2 - Z_1) + K_n(\dot{Z}_1 - \dot{Z}_2) - M \cdot g \\ m\frac{d^2Z_2}{dt^2} = C_n(Z_1 - Z_2) - C_h(Z_2 - Z_3) - K_n(\dot{Z}_2 - \dot{Z}_1) + K_h(\dot{Z}_3 - \dot{Z}_2) - m \cdot g \\ m_c\frac{d^2Z_3}{dt^2} = C_h(Z_2 - Z_3) - K_h(\dot{Z}_3 - \dot{Z}_2) - C_c \cdot Z_3 + K_c \cdot \dot{Z}_3 - m \cdot g \end{cases} \tag{7}$$

where: K_n, K_h, K_c—suspension damping coefficients, respectively, suspension, tire and tester [Ns/m], C_n, C_h, C_c—stiffness of, respectively, suspension, tire and tester [N/m], Z_1, Z_2, Z_3—coordinates of the vertical movement, respectively, of the sprung mass (M), the unsprung mass (m) of the tester plate and the mass (m_c) [m].

Static deflection of the elastic elements of the suspension, tires and stiff plate were calculated relatively to the steady state of car mass according to Hooke's law using the following formulas (8):

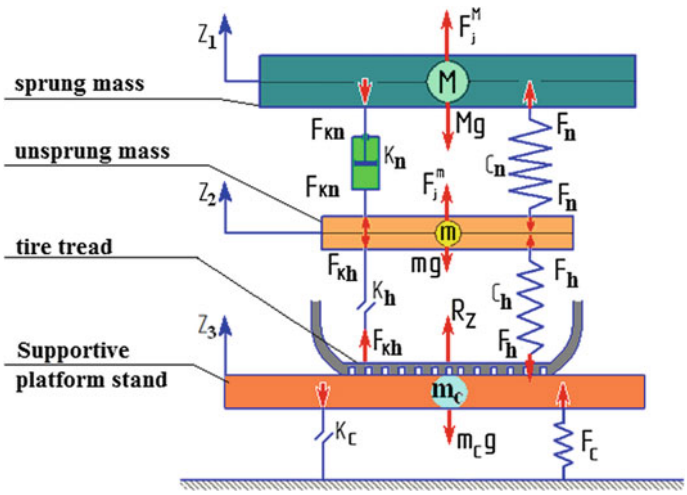


Fig. 3 Scheme to determine the normal reaction of wheels R_z during braking on flat tester

$$\Delta l_n = \frac{M \cdot g}{C_n}; \Delta l_k = \frac{(M+m) \cdot g}{C_h}; \Delta l_c = \frac{(M+m+m_c) \cdot g}{C_c} \tag{8}$$

Normal reaction R_z of supporting plate is determined by the formula (9):

$$R_z = K_c \cdot \dot{Z}_3 - C_c \cdot Z_3 \tag{9}$$

Numerical study was based on solutions of Eqs. (1)–(9) is performed together with the equations of the braking process of a car wheel [4]. As a result, mathematical model of the braking wheel process beginning from run onto tester is obtained and suitable graphs of two trials are shown in Fig. 4.

The graphs show that the longitudinal oscillations of the sprung and unsprung mass of the car, as well as areas of the tester having largest amplitude. They make significant changes in the dynamics of a vehicle wheel braking. This process can be divided into two stages: 1st phase is when the wheel runs onto plate of a tester and 2nd phase while blocking the wheel.

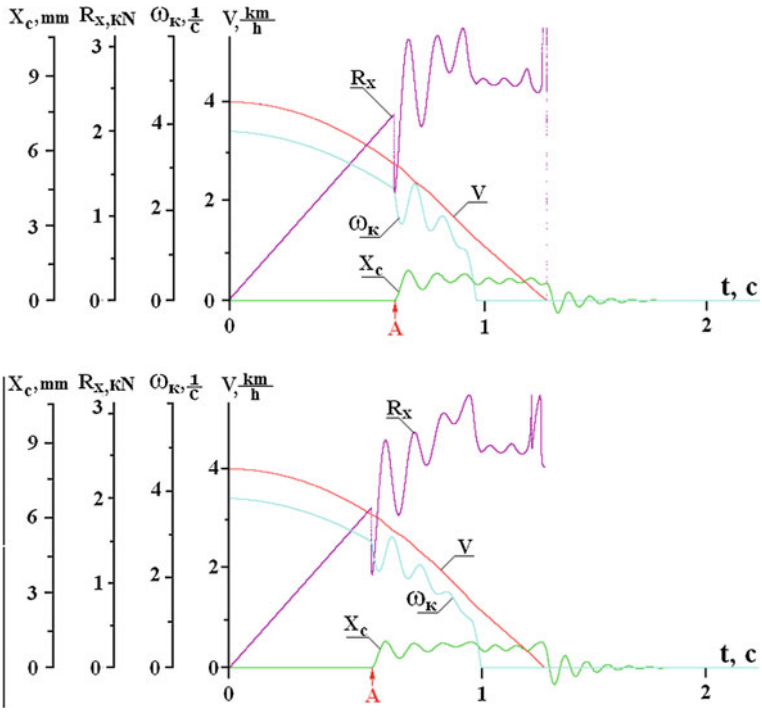


Fig. 4 Graphics car wheel braking process with collisions on a plate tester at point A

3.1 1st Phase—Wheel Runs onto Plate of a Tester

At the moment of running a wheel onto plate of a tester (Fig. 4, point A), under the influence of the braking force, the longitudinal displacement of the plate occurs and subsequent damped oscillations excite longitudinal vibrations of the tire, as well as fluctuations in the sprung and unsprung mass of the vehicle. The graph clearly shows the braking force of the amplitude of these oscillations. Calculated error at this stage according to formula (10) is 51 ÷ 57 %.

$$\delta = \frac{\Delta F_t}{F_{tmax}} \cdot 100\,\% \tag{10}$$

where: F_{tmax}—the maximum braking force,
ΔF_t—oscillation amplitude of the braking force.

3.2 2nd Phase—Blocking the Wheel

At the time of wheel blocking ($\omega_k = 0$), the wheel loses contact between tyre and a plate of the tester and reduced frictional properties of the tire are observed. This leads to a new cycle of oscillation, which is accompanied by periodic joining and detaching of both surfaces. Moreover, when decelerating the vehicle, this process is accompanied by the resonance, which causes an increase of oscillation amplitude of the braking force. Margin of error when locking a wheel is 24 ÷ 29 %.

4 Experimental Study of Fluctuations of Braking Properties on Flat Tester

4.1 Analysis of Initial Vehicle Speed Influence on Measurement Error

Experimental investigation of the influence of parameters of the car and tester by the uncertainty of measurement of brake forces was performed in the diagnostic laboratory of the Transport Department of Irkutsk State Technical University.

Firstly, initial values of vehicle speed V were varied in the range of 2–20 m/s (Fig. 5).

It is seen that with increasing initial speed of braking vehicle V, braking force measurement error is significantly reduced by the logarithmic dependence. This is due to the increase of the inertial moment of braking wheel at higher speed.

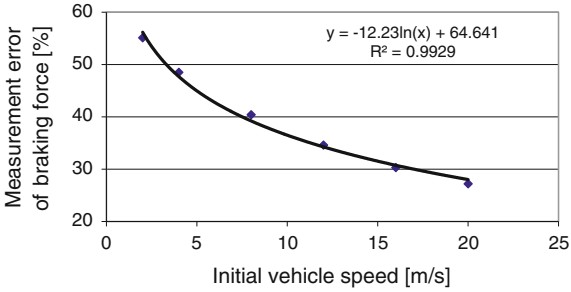


Fig. 5 Measurement error of the braking force in function of car speed

Therefore, to reduce the measurement error of the braking force initial deceleration rate should be increased. But in case of braking flat tester it is very difficult to stop the car just on the tester with limited length (in inertial testers). On power flat testers high speed is technically difficult to implement.

4.2 Analysis of Breaking Time Influence on Measurement Error

It is known that the brake pedal pressing takes some time before the wheel comes over to the tester plate. Therefore, the next phase of the study was the analysis of measurement error depending on the magnitude of the braking force interval t_n from the beginning of braking before the wheel is touching the plate (Fig. 6).

Test was conducted for the initial speed of $V = 4$ m/s and the rate of braking torque increase is 1497 N m/s.

The results indicate that with increasing time t_n braking force measurement error first increases, then reaches an extremum, and then decreases. This dependence is expressed by a quadratic form of a parabola.

The initial growth of the error is due to the fact that for a small braking time the braking force is not large and its reaction impact on the area of the tester is also not

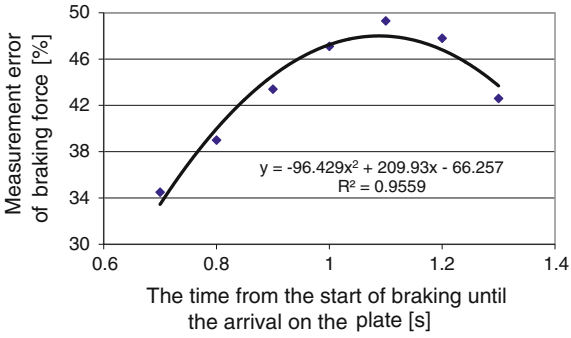


Fig. 6 Graph of the measurement error of the braking force at time t_n varying from the start of braking until hitting the tester plate

very large. Consequently, the oscillation amplitude of the braking force is relatively small. With increasing time t_n braking force measurement error increases as the braking force increases. Extremum occurs at a time when run onto the plate coincides with the wheel lock. At this point, there are the greatest oscillations of braking force amplitudes.

4.3 Analysis of Breaking Force Influence on Measurement Error

To analyse an impact of load applied to the wheel it is working-out an application simulated brake force in function of mass weight. Figure 7 shows the results of the analysis of impact of that load on the wheel R_z on braking force measurement error.

The graph in Fig. 7 discovers that with increasing load on the braking wheel braking force measurement error is significantly reduced by the law of a quadratic function. This is explained by the fact that with increasing load reduced is amplitude oscillation in the elements of the car and tester. The reason for this is that, firstly, load on tester is increasing and secondly, contact of the tire with the surface of the tester becomes stable.

4.4 Analysis of Plate Mass Influence on Measurement Error

Another element influencing brake force and measurement error is a mass of the plate. Weight of tester plate is made of steel and has a significant impact on the accuracy of the braking force measurement. As the results of the study (Fig. 8) with increase in mass of the plate from 5 to 40 kg the measurement error of the braking force decreases more than twice.

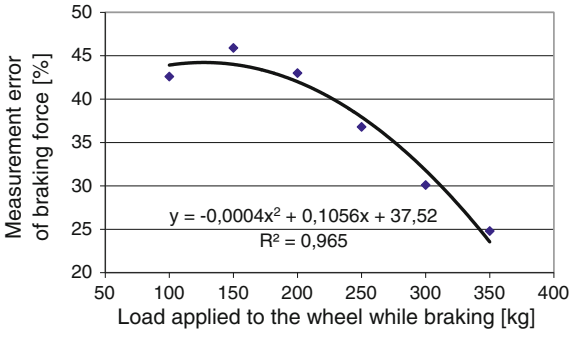


Fig. 7 Graph of the measurement error of the in function of load applied to the wheel while braking

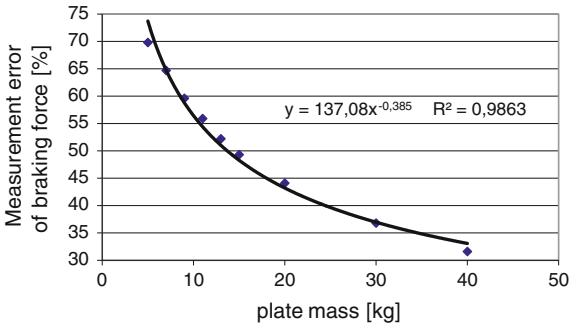


Fig. 8 Graph of the measurement error of the braking force of the weight of the tester site

4.5 *Analysis of the Error of Braking Force Lateral Displacement Influence on Measurement Error*

Figure 9 shows studies on dependence of measurement error of the braking force of the wheel due to relative lateral displacement of the longitudinal axis of symmetry of the tester. Increasing lateral displacement of the wheel relatively to the axis of symmetry of the tester increases measurement error of the braking force. As noted earlier (1–3), lateral displacement affects the accuracy of measurements of the braking force due to friction forces caused by moment twisting a plate. In lateral displacement, effect of disturbing vibrations only increases the error, but not because of growth of the amplitude fluctuations but by reducing (by the amount of friction force) the maximum value of the braking force.

4.6 *Analysis of Influence of Tire Dumping Properties on Measurement Error*

It is logical that in the conditions of mass oscillations of the car tires and pads on the margin of error of the braking force measurement affect the damping properties of the tire. This clearly demonstrates the resulting graph in Fig. 10.

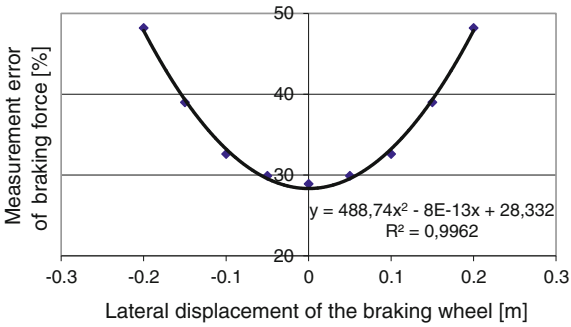


Fig. 9 Measurement error of the braking force of the lateral displacement of the wheel relative to the longitudinal axis of symmetry of the tester plate

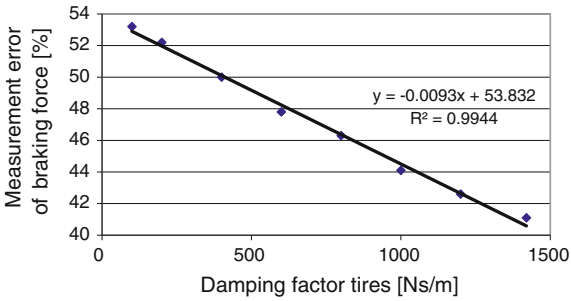


Fig. 10 Graph of the measurement error of the braking force of the damping properties of tires

5 Conclusions

Numerical simulation presented in the paper shows great advantage of that tool in dynamic process analysis. Proposed dynamic model of the system: wheel-flat plat tester is validated. Simulation results reasonably confirm doubts concerning measurements of braking force using simple testers not equipped with advanced compensating sensors.

Obtained simulation results strongly suggest that besides some benefits of the flat brake testers, they have number of drawbacks. The most significant of which are:

- longitudinal vibrations of tester plate cause disruptions of the contact between the wheel and tester plate at the moment of running onto tester plate as well as at the time of blocking the wheel,
- high complexity of braking wheels positioning into the centre of the tester plate and in consequence arising moments rotating tester plate,
- instability of test performance (variability of force and speed of pressing the brake pedal).

Collectively, all above reasons cause methodological mistake of the brake forces measurement on flat testers resulting the measurement error exceeding 50 % or more [1, 4].

This is not an exhaustive list of problems to be overcome in the operation of flat (plate) brake testers. Therefore, in some countries they are banned as a means of control of the brake systems.

The comparison of the metrological properties of flat brake testers with similar properties of brake roller dynamometers [2, 6, 7] shows a clear advantage of the latter.

Presented simulation analysis and experimental research confirm usability of the obtained results in safety analysis of dynamic systems.

References

1. Bojko, A.V., Fedotov, A.I., Potapov, A.S.: About repeatability car braking process parameters on a roller dynamometer. Herald ISTU. Release №1. ISTU, Irkutsk (2008)
2. Bojko, A., Fedotov, A.I., Khalezov, W.P., Mlynczak, M.: Analysis of brake testing methods in vehicle safety. In: Nowakowski et al. (eds.) Safety and Reliability: Methodology and Applications. Taylor & Francis Group, London (2015)
3. Fedotov, A.I., Boyko, A.V.: The Causes of the Errors in the Diagnosis of the Braking System of Cars on a Roller Dynamometer. Collected Scientific Works of IV All-Russian NPK Polytransport System (transport systems of Siberia). KSTU, Krasnoyarsk (2006)
4. Fedotov, A., Dick, A.B.: Rolling Retarding Wheels, Loaded with Variable Normal Load. The Collection of Scientific Papers "Active and passive safety and reliability of the car". MAMI, Moscow (1984)
5. Hebda, M., Niziński, S., Pelc, H.: Podstawy diagnostyki pojazdów mechanicznych. WKiŁ, Warszawa (1980)
6. Heka. Internet: www.heka-online.de
7. MAHA Maschinenbau Haldenwang. Internet: www.maha.de
8. Pacejka, H.B.: Some recent investigations into dynamics and frictional behaviour of pneumatic tires. In: Physical Tire Traction. Theory and Experiment, New-York, London (1974)
9. Wypadki drogowe w Polsce w 2011 roku (Road Accidents in Poland in 2011). Komenda Główna Policji, Biuro Ruchu Drogowego, Zespół Profilaktyki i Analiz, Warszawa (2012)

Analytical Identification of Parameters Influencing Measurement Quality Using Flat Brake Tester

A.I. Fedotov and M. Młyńczak

Abstract Vehicle braking capability may be tested in two options: on-road or in workshop conditions. On-road testing is the most credible but difficult to repeat the same conditions and hard to measure and collect braking data. On the other hand testers testing in workshop conditions provide comparable conditions but are less similar to real braking process on the road. Anyway periodical testing of braking systems is performed in workshops equipped with special testers based on flat plates or rolling drums. Both methods involve errors and it is difficult to decide about superiority of one of it. The basic problem is observed in contact area between at tire and surface of the tester. Drum testers are easier to operate and more popular as require less space but the contact area is not so adequate comparing to flat testers. Paper describes analytical approach to measurement errors identification. Methodology of brake testing on flat testers and measurement system is shown. There are discussed measurement errors on flat testers related to the positioning of the vehicle during braking as well as the type and parameters of the test modes. Another paper discusses simulation of the brake process on flat plate tester.

Keywords Brake flat testers · Brake roller tester · Brake system · Technical inspection · Error · Vehicle

1 Introduction

Checking of vehicle brakes in operation is carried out both on-road, in real conditions and in workshops, using artificial conditions [1, 2, 5]. The most common method of monitoring the technical condition of brake systems is testing on special

A.I. Fedotov (✉)
Irkutsk State Technical University, Irkutsk, Russian Federation
e-mail: fai@istu.edu

M. Młyńczak
Faculty of Mechanical Engineering, Wroclaw University of Technology, Wrocław, Poland
e-mail: marek.mlynczak@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_13

electro-mechanical devices and the most widely the drum brake tester (DBS) is used (Fig. 1). Testing a vehicle brake system in that tester consists in measuring reactions of rotating rollers when break is applied to the wheel (Fig. 1a) [6–8]. History shows, that periodically appear attempts to use for checking of vehicle technical state of brake systems the flat plate testers (tester), as inertia and power [10, 11] (Fig. 2). Arguments of using it as a rule are as follows:

- supporting surface (plate) corresponds to real flat road surface (favourably with roller) (Fig. 2),
- braking on inertia flat testers is characterized by redistribution of the load between the front and rear axles as it is in real driving conditions,
- flat testers are compact, less material-consuming than a roller dynamometer, structurally simpler [9].

Vehicle brake systems are vital from safety point of view and it is required that its functional performance is as high as possible. Maintenance of vehicle brake system is based on preventive maintenance and occasional repairs, wherein latter

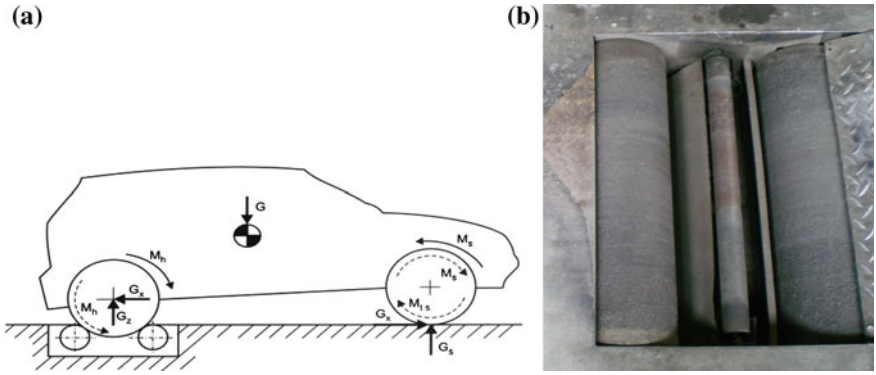


Fig. 1 Roller brake testing tester (scheme, rollers)

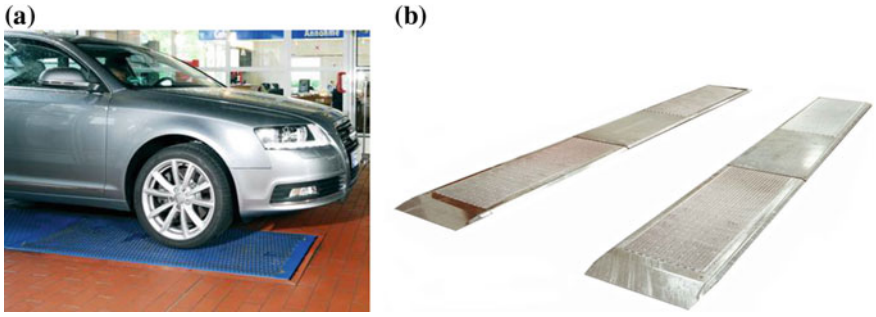


Fig. 2 Flat, plate brake tester; **a** with car wheel on it, **b** solo, two truck tester [8]

are mostly not acceptable. Preventive maintenance is performed as condition—based maintenance what requires making a test of selected parameters. The most important, functional parameter in technical objects is its main function what in case of brakes means abilities to effective stop a car when required. Looking for reliability measures for brake system one may consider availability, as for system which is repaired objects. Unfortunately, brakes as safety system should perform its function on each demand and availability covering up and down time is not suitable. The most applicable seems to be hazard rate function $\lambda(t)$ as conditional probability of failure at time t assuming that it has not failed until time t. It describes a certainty of effective braking between maintenances keeping a system as good as new. The above statements motivate to find out the differences and advantages/disadvantages of testing methods used in daily practice. Before concluding on advantage of drum brake tester over flat testers, problems that are associated with the characteristics of their design and operation should be analysed.

2 Analysis and Design of Metrological Parameters of Flat Testers

One of the most critical issues, while designing flat testers, is a set of their geometrical parameters. Obviously, construction of universal flat tester for all variety of variety of vehicles is not possible. So let us find out what geometrical dimensions should be described flat tester, for example for general use personal cars. Figure 3 shows a diagram of flat tester with the left (2) and right (3) plates with shape of car wheels.

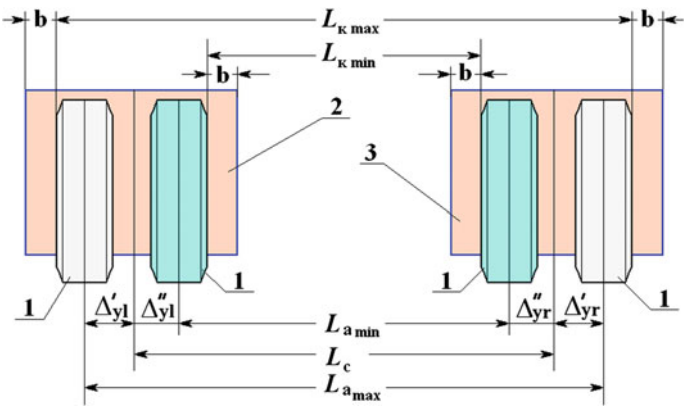


Fig. 3 Scheme for the geometric dimensions flat tester: *1* car wheel, *2 left* plate of the tester, *3 right* plate of the tester

In order to determine the width of the pads and the distance between them it is necessary to know the following geometric dimensions:

1. $L_{\kappa max}$—the maximum distance between the outer sides of the car wheels having broad wheel track,
2. $L_{\kappa min}$—the minimum distance between the inner sides of the car wheels having narrow wheel track,
3. b—side margin to guarantee position of wheels on the surface of tester plate,
4. L_c—distance between the axes of symmetry of tester plates.

And if the definition of the distance b is no problematic (it is usually $b = 0.1 – 0.15$ m), to determine the remaining geometric parameters like $L_{\kappa min}$, $L_{\kappa max}$, L_c it is necessary to know the dimensions of wheel truck $L_{\kappa min}$ and $L_{\kappa max}$ of the front and rear wheels. In Figs. 4 and 5 the size difference between the inner and outer surfaces of the front and rear axle wheels of passenger cars made in Germany, Italy, France, Russia, the USA, Sweden and Japan is shown.

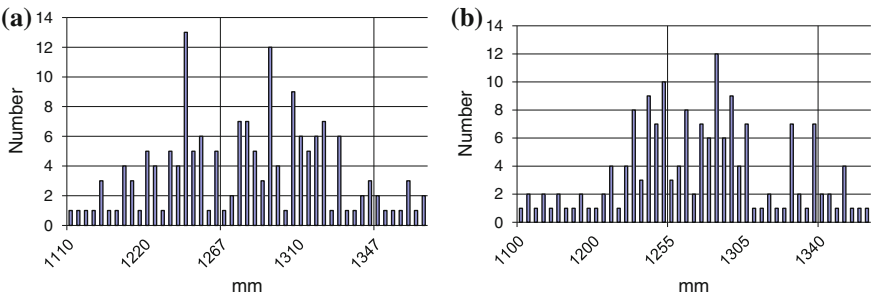


Fig. 4 Statistics of the distance between the inner surfaces of: the front axle wheels (**a**) and rear axle wheel (**b**)

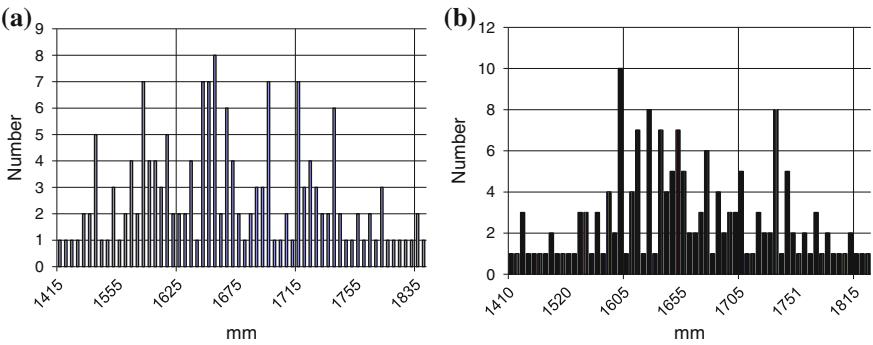


Fig. 5 Statistics of the distances between the outer surfaces of the front axle wheels (**a**) and rear axle wheels (**b**)

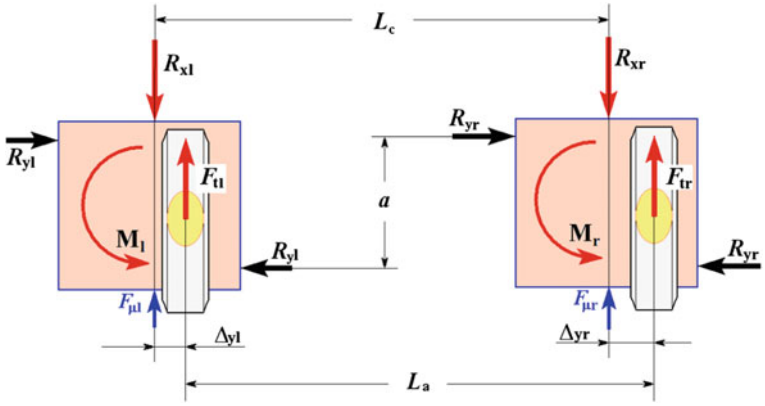


Fig. 6 Diagram of the forces and moments acting on the plate of brake tester

Analysis of data shown in Figs. 4 and 5 gives the following results. The smallest sizes between the inner surfaces of the wheels ($L_{\kappa min}$) have: VAZ 1101, Fiat Cinquecento 1100, Volkswagen Polo 1.05 and Renault Clio 1.4. Based on that analysis, the distance $L_{\kappa min}$ can be assumed as $L_{\kappa min}$ =1100 mm. The largest dimensions between the outer surfaces of the wheels ($L_{\kappa max}$) have: Mercedes-Benz S-600, Mercedes-Benz S-420, as well as such cars like Oldsmobile Aurora 4.0i and Ferrari F 50. Based on that analysis, the distance $L_{\kappa max}$ can be assumed as $L_{\kappa max}$ = 1900 mm.

Thus, the difference between the size of $L_{\kappa min}$ and $L_{\kappa max}$ is 800 mm. Having known side distance b (Fig. 3), the width of each tester plate for general use cars is b = 600 ÷ 700 mm. Taking that parameter of the tester we may consider the scheme of forces and moments that arise during operation flat tester (Fig. 6).

3 Moments and Forces Acting on the Plate of Brake Tester

Construction of the flat tester allows that car wheels are positioned not symmetrically about its longitudinal axis. It results for a large wheel track that axis of tester symmetry will be anywhere between the inner sides of the tire: to the outer part of the plate, at the right wheel and to the inner part of the plate, at the left wheel (Fig. 6).

Typically, sensors for measurement of braking force are set on the axis of symmetry of the plate. Therefore, an arm appears between the brake force F_{tr} and reaction R_{xr} that causes moment M_r rotating a plate. Assuming mass of the car m =

2000 kg, the value of $\Delta_{yr} \approx 0.2$ m and friction coefficient $\mu = 0.7$, moment can be determined as (1):

$$M_r = F_{tr} \cdot \Delta_{yr} = R_z \cdot \varphi \cdot \Delta_{yr} = 6000\,\text{N} \cdot 0.7 \cdot 0.2\,\text{m} = 840\,\text{N m} \tag{1}$$

where: $R_z = 6000$ N is a value of the normal reaction on the front wheel

To prevent rotation of the plate there are attached holding outriggers. Under the action of the moment M_r reactions arise as a pair of forces R_{yr}. Knowing the arm $a \approx 0.6$ m, reactions are (2):

$$R_{yr} = M_r/s = 840\,\text{N}/0.6\,\text{m} = 1400\,\text{N} \tag{2}$$

Forces R_{yl} and R_{yr} cause friction force $F_{\mu l}$ and $F_{\mu r}$ acting in the longitudinal direction and preventing longitudinal displacement of plates (3):

$$F_{\mu r} = 2R_{yr} \cdot \mu \quad F_{\mu l} = 2 \cdot R_{yl} \cdot \mu \tag{3}$$

where: μ—the coefficient of friction in the bearings of the tester plate.

In operation, supports are contaminated with grease what leads to monotonic increase of friction coefficient from 0.08 to 0.23 and more. As a result, the frictional forces increase from 220 to 640 N and this gives an error in determining the braking force, the value of which exceeds over 15 %.

Friction forces F_l and F_r occur only when moments M_l and M_r act and did not appear in the process of calibration of measurement system. Moreover, the magnitudes of the distances Δ_y between directions of braking forces application are not constant. They depend not only on the size of a wheel track, but also on the displacement of the longitudinal axis of symmetry of the vehicle with respect to the axis of symmetry of the tester. The larger the shift in setting the car on the tester happens, the greater difference between the values (Δ_{yl}, Δ_{yr}) and greater the difference in measurement error of braking forces F_{xl} and F_{xp}. Hence the incorrect determination of the relative difference between the braking forces, what is one of the braking stability indicator. In fairness, it should be noted that at roller dynamometer that problem doesn't exist.

As shown earlier by Sergeev [10, 11] measurement error of brake forces using flat testers reaches 50 %. Author explains that fact, that the margin of measurement error of braking forces on flat testers is affected by instability of testing procedure (change in force and speed on the brake pedal), as well as plate swinging caused by wheels). Impact of these factors on the dynamic measurement error of brake forces on flat testers is given by the formula (4):

$$\delta = \frac{\Delta F_t}{F_{tmax}} \cdot 100\,\% \tag{4}$$

where: F_{tmax}—the maximum braking force,

ΔF_t—oscillation amplitude of the braking force.

4 Analysis of Testing Process of Brake Systems on Flat Testers

For flat testers there are observed specific measurement errors not met in testing on the roller testers. These errors are structurally inherent of flat testers and cannot be compensated by tester modification. Their presence will increase the difference between flat testers tests with the results of road tests [1, 4]. Let us consider another source of systemic error on the example of scheme shown in Fig. 7a and graph in Fig. 7b. The process of braking the wheels of one axle on the tester is discussed. Based on the principle of reversibility of motion [3], the braking system requires that "road" under both wheels of the car is moving synchronous. Otherwise, the results of measurements of brake forces are not adequate. On the roller testers, this requirement is provided. For flat testers equal displacement of plates is required, not equal resistance forces, as we see in the existing and proposed structures with pneumatic and hydraulic displacement balancing.

During brake testing, at the time t_o the vehicle wheels rotate with the angular velocity of ω_k and then applied is braking torque M_t (not shown in Fig. 7b). In the contact point between a wheel and supporting plate appear brake forces F_{tl} and F_{tr}, for left and right wheel respectively. Upon reaching the limit of friction grip $R_z\,\varphi$, at time t_2 the right wheel locks up and the braking force will be equal to the force of adhesion (Fig. 7b) (5):

$$F_{tr} = R_z \cdot \varphi \tag{5}$$

As it is known, a quantitative measure of braking stability of the car under test-bench is a relative difference in braking forces K_n measured individually for each axle (6):

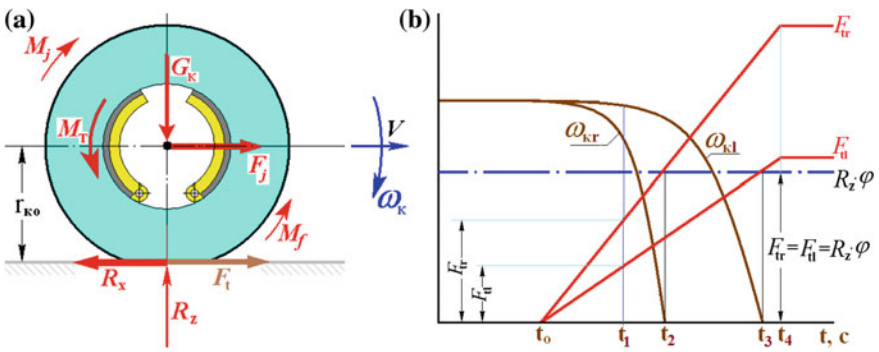


Fig. 7 Influence of non-uniformity of left and right wheels rotation in test: **a** drawing scheme, **b** graph of the braking process of the wheel on the testers

$$K_n = \left| \frac{F_{tl} - F_{tr}}{F_{tmax}} \right| \tag{6}$$

where: F_{tmax}—the larger of two measured brake force F_{tl} and F_{tr}.

To determine the braking force reading on the roller testers it is necessary to observe slippage S of the wheels that relatively synchronously rotate on the supporting rollers [4, 6]:

$$S = 1 - \frac{\omega_k \cdot r_{ko}}{\omega_p \cdot r_p} \tag{7}$$

where: ω_k—angular wheel rotation speed; r_{ko}—the radii of the wheels; r_p—the radius of tester supporting roller; ω_p—the angular velocity of the supporting rollers.

To correct determination of the relative difference in braking forces K_n it is required to measure brake forces F_{tl} and F_{tr} included in formula (6). It should be done at a time t_1 when one of the wheels reaches assumed level of slippage, usually $S \approx 0.2$.

Determination of the relative difference between the braking forces K_n will be correct only if the measurement is performed in operation of the braking system (in Fig. 7b—from time t_1 to time t_2), provided synchronous motion of the tester plates. This condition is strictly performed on roller testers, and very difficult to achieve on flat testers.

After time t_2 force increase on the brake pedal is not accompanied by an increase of the braking force F_t as the limit of the adhesion force (5) is achieved. Figure 7b shows the graphs of braking forces F_{tl} and F_{tr}, due to the potential efficiency of brake mechanisms. Therefore, measures of the relative difference between braking forces K_n after the time t_2 (Fig. 7b) will not be correct, because as the period from the time point t_2 gets closer to the point t_3 the difference between brake forces F_{tl} and F_{tr} will be reduced. Starting from time t_3 Eq. (8) is valid:

$$F_{tl} = F_{tr} = R_z \cdot \varphi \tag{8}$$

It means that after time t_3 there will be carried out measurement not braking forces but cohesive forces of tires with tester surfaces and a relative difference in braking forces will not be measured.

Equation (8) is as more accurate as the load is evenly distributed between the wheels of the controlled axle and assured is equality of friction coefficients on the sides. Relating to those conclusions, it is pointless to perform braking test on flat testers with locked wheels.

5 Conclusions

Popular in some countries flat plate brake testers are very sensitive on position of wheels on the plates. Run onto plates should be very precise to provide conformity of vehicle and tester axes. Otherwise tangential forces appear introducing measurement error. Similar error is caused by non-uniformity of angular wheel speed. Above methodological mistake in measurement methodology of brake forces on flat testers makes an error exceeding 50 % or more [10, 11]. Important safety aspect in that methodology consists in high impreciseness of braking force measurement what may lead to catastrophic consequences in the road. Wrong decision of diagnostician would not be considered then as human error in putting a vehicle into service.

The comparison of the metrological properties of flat brake testers with similar properties of brake roller dynamometer [1, 4] shows a clear advantage of the latter.

References

1. Bojko, A.V., Fedotov, A.I., Potapov, A.S.: About repeatability car braking process parameters on a roller dynamometer. Herald ISTU. Release №1. ISTU, Irkutsk (2008)
2. Bojko, A., Fedotov, A.I., Khalezov, W.P., Mlynczak, M.: Analysis of brake testing methods in vehicle safety. In: Nowakowski et al. (eds.) Safety and Reliability: Methodology and Applications. Taylor & Francis Group, London (2015)
3. Fedotov, A.I.: Vehicle Diagnosis: A Textbook for High Schools. ISTU Publisher, Irkutsk (2012) (468 p. Fig. 273)
4. Fedotov, A.I., Boyko, A.V.: The Causes of the Errors in the Diagnosis of the Braking System of Cars on a Roller Dynamometer. Collected Scientific Works of IV All-Russian NPK Polytransport System (transport systems of Siberia). KSTU, Krasnoyarsk (2006)
5. Fedotov, A., Dick, A.B.: Rolling Retarding Wheels, Loaded with Variable Normal Load. The Collection of Scientific Papers "Active and passive safety and reliability of the car". MAMI, Moscow (1984)
6. Hebda, M., Niziński, S., Pelc, H.: Podstawy diagnostyki pojazdów mechanicznych. WKiŁ, Warszawa (1980)
7. Heisler, H.: Advanced Vehicle Technology. Butterworth Heinemann, Oxford (2002)
8. Heka. Internet: www.heka-online.de
9. MAHA Maschinenbau Haldenwang. Internet: www.maha.de
10. Sergeev, A.G.: The accuracy and reliability of the vehicle diagnostics. Transport (1980)
11. Sergeev, A.G.: Metrological maintenance of motor vehicles. M.: Transport (1988)

Arithmetic in Finite Fields Supporting Type-2 or Type-3 Optimal Normal Bases

Sergey Gashkov, Alexander Frolov and Igor Sergeev

Abstract In this paper, we generalize an approach of switching between different bases of a finite field to efficiently implement distinct stages of algebraic algorithms. We consider seven bases of finite fields supporting optimal normal bases of types 2 and 3: polynomial, optimal normal, permuted, redundant, reduced, doubled polynomial, and doubled reduced bases. With respect to fields of characteristic $q = 7$ we provide complexity estimates for conversion between the bases, multiplication, and exponentiation to a power q^k, q-th root extraction. These operations are basic for inversion and exponentiation in $GF(7^n)$. One needs a fast arithmetic in $GF(7^n)$ for efficient computations in field extensions (7^{2n}), $GF(7^{3n}), GF\left(7^{6n}\right)GF(7^{14n})$, $GF(7^{3\times 14n})$ which are the core of the Tate pairing on a supersingular hyperelliptic curve of genus three. The latter serves for an efficient implementation of cryptographic protocols.

Keywords Finite fields · Polynomial basis · Optimal normal basis · Bases conversion · Multiplication · Exponentiation · Inversion · Root extraction · Supersingular elliptic curve of genus three · Tate pairing

S. Gashkov (✉)
MSU, Faculty of Mechanics and Mathematics,
Lomonosov Moscow State University, GSP-1,
1 Leninskiye Gory, Main Building, 119991 Moscow, Russia
e-mail: sbgashkov@gmail.com

A. Frolov
Moscow Power Engineering Institute, National Research University,
Krasnokazarmennaya, 14, 111250 Moscow, Russian Federation
e-mail: abfrolov@gmail.com

I. Sergeev
Research Development Institute "Kvant",
4th Likhachevsky, 15, 125438 Moscow, Russia
e-mail: isserg@gmail.com

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_14

1 Introduction

The theoretic basis for algebraic transforms in modern telecommunication systems is the theory of finite fields and rings over finite fields [1]. For instance, widely used BCH-codes, Reed-Solomon error-correcting codes are defined through algebraic operations in finite fields of polynomial rings over finite fields. Arithmetic in the mentioned algebraic structures involves a way of representation of a structure and a number of algorithms for algebraic operations and transforms. Recall the Berlecamp-Messey algorithm used for decoding of mentioned codes to illustrate. In its turn, arithmetic algorithms for finite fields [2] is the algorithmic basis for encoding-decoding in telecom systems and of hardware and software its information security subsystems intensively exploiting cryptographic protocols. So, the main purpose of development of algorithms in finite fields and their extension rings is an improvement of dependability and information rate of communication with account of the time required for implementation of coding-decoding systems and cryptoprotocols. While constructing a cryptosystem one faces a problem of the choice of suitable field representation and constantly needs to modify arithmetic algorithms to involve new knowledge of finite field's properties and applications, and new tendencies of cryptoprotocols evolution. The latter appears in discovering of protocols basing on the pairing of divisors of super-singular elliptic or hyper-elliptic curves [3–5]. An important feature of the underlying pairing algorithm is requirement of fast exponentiation as well as fast multiplication in a finite field. There are two strategies to improve an efficiency of such protocols—increasing of a field characteristic and exploiting of two different field representations: standard and normal. Indeed, at the present time one can register a migration from fields of characteristic 2 or 3 [6] to fields of characteristic 7 or above [7, 8]. The present paper deals with the aspects of implementation of arithmetic in finite fields of characteristic 7 and their extension rings to satisfy the requirements of the further application to the Tate pairing over a super-singular hyper-elliptic curve. We constantly kept in mind an application to the Tate pairing on supersingular hyperelliptic curves of genus three made on the arithmetic in extensions $GF(7^{3n})$, $GF(7^{2n})$, $GF(7^{6n})$, $GF(7^{14n})$, and $GF(7^{3\times 14n})$ of basic field $GF(7^n)$ [7, 8]. Recall that the Tate pairing is a widely applicable tool in elliptic curve cryptography, see e.g. [3, 4]. As far as the pairing algorithm involves stages with domination of different field-arithmetic operations one can gain a benefit from exploiting different representations (that is, different bases) of the finite field in the question [1, 2]. Here, we do not consider cryptographic properties of the discussed algebraic transforms.

Present paper consists of 5 sections. In Sect. 2 we discuss the idea of combining distinct bases for acceleration of computations with operations of exponentiation and multiplication. In Sect. 3 some bases of finite field and algorithms of transformations between them are considered and estimated. In Sect. 4 there is founded algorithm of multiplication in the ring GF(7)[X] with record running time in comparison with known algorithms. In Sect. 5 some algorithms of finite field's

arithmetic based on results of previous sections are described and estimated. In conclusion expected benefits of paper results application are shown.

2 Using of Distinct Bases in Finite Field Algebraic Computations

It is common knowledge that polynomial bases (p.b.) allow fast multiplication, due to the intensively developed theory of polynomial multiplication. On the other hand, normal bases (n.b.) provide low-cost exponentiation to the power of the field characteristic since it corresponds to a cyclic shift. Three types of optimal normal bases (o.n.b.) discovered in [9] allow straightforward multiplication algorithm of quadratic complexity, though it's not efficient enough. The true advantage of o.n.b. is in the existence of $O(n \ln n)$ complexity algorithms for conversion to and from p.b. [10] (in the case of the type-1 o.n.b. conversion algorithms have $O(n)$ complexity). Thus, one can switch to p.b. for multiplication and switch to n.b. for exponentiation to power q^j where q is the field characteristic.

Recall that the $GF(q^n)$ multiplication in p.b. is an ordinary multiplication of $(n-1)$-degree polynomials over $GF(q)$ to be followed by the reduction modulo an n-degree polynomial (generating polynomial of the field $GF(q^n)$). It was offered in [11, 12] to implement this reduction via conversion to the (double) n.b. and consequent reduction to an ordinary n.b. A slightly optimized algorithm across the same lines was given in [13].

Here, we consider fast arithmetic in finite fields of characteristic 7 with potential application to the Tate pairing on supersingular hyperelliptic curves of genus three [7, 8, 14]. The Tate pairing requires arithmetic in fields $GF(7^{2n})$ and $GF(7^{14n})$. Since any of these fields can be regarded as extension of the field $GF(7^n)$ one has to efficiently compute arithmetic operations in the latter field. For this purpose, we extend some previous results concerning fields of characteristic 2 and 3 [15]. To illustrate our techniques we choose the field $GF(7^{29})$ appropriate for application to the Tate pairing, see e.g. [4] and supporting type-3 o.n.b.

3 The Bases of Finite Fields with o.n.b.-2 or o.n.b.-3

Let $p = 2n+1$ be a prime number, p divides $q^{2n}-1$. Let q be a primitive root modulo p of 1, i.e. $\langle q \rangle = Z_p^*$ or $\langle q \rangle$ is the set of all quadratic residues modulo p. In the first case q is a quadratic non residue modulo p, in the second case $q^n \bmod p = 1$ and $q^k \bmod p \neq 1$ for $k < n$, or, in other words, q is a quadratic residue modulo p, whereas -1 is a quadratic non residue modulo p. In the first case, there exists a type-2 o.n.b. In the second case, there exists a type-3 o.n.b. Henceforth, o.n.b. denotes o.n.b. of any of these two types. To describe these bases, take

$\alpha \in GF(q^{2n}), \alpha \neq 1, \alpha^p = 1$. Denote $\beta = \alpha + \alpha^{-1}$. O.n.b. is the set $\{\xi_1, \ldots, \xi_i, \ldots, \xi_n\}, \xi_i = \beta^{q^{i-1}}$.

Consider an element $\beta_i = \alpha^i + \alpha^{-i} = \alpha^i + \frac{1}{\alpha^i} \in GF(q^{2n})$ for any integer i. Note that $\beta_i \in GF(q^n)$ when $q^n \equiv 1 (\mathrm{mod}\, p)$. Obviously, $\beta_i = \beta_{-i}, \beta_0 = 2 \in GF(q)$. The set $\{\beta_1, \ldots, \beta_i, \ldots, \beta_n\}$ is called a permuted basis, it can be represented as permutation of an o.n.b.: for $i = 1, \ldots, n$

$\beta_i = \xi_{\pi(i)}$ where π is defined as

$$\pi(i) = \begin{cases} q^i & \text{if } q^i \mod p \leq n, \\ p - q^i & \text{if } q^i \mod p > n. \end{cases}$$

The inverse conversion corresponds to the inverse permutation: $\xi_i = \beta_{\pi^{-1}(i)}$.

One can verify that $\sum_{i=1}^{n} \beta_i = \sum_{i=1}^{2n} \alpha^i = -1 = \frac{q-1}{2}\beta_0$.

We also consider reduced o.n.b. $\{\beta_0, \beta_1, \ldots, \beta_{n-1}\}$ and redundant o.n.b. $\{\beta_0, \beta_1, \ldots, \beta_{n-1}, \beta_n\}$. To convert a representation from the reduced o.n.b. to the permuted o.n.b. one uses the equality $\beta_0 = -2\sum_{i-1}^{n} \beta_i$: for $\mathbf{x} = x_0\beta_0 + \cdots + x_{n-1}\beta_{n-1}$ in the reduced o.n.b. one has

$$\mathbf{x} = (x_1 - 2x_0)\beta_1 + \cdots + (x_{n-1} - 2x_0)\beta_{n-1} - 2x_0\beta_n \tag{1}$$

in the permuted o.n.b. To implement the inverse conversion write $\beta_n = \frac{q-1}{2}\beta_0 - \sum_{i=1}^{n-1} \beta_i$: so, for $\mathbf{x} = y_1\beta_1 + \cdots + y_{n-1}\beta_n$ in the permuted o.n.b. we obtain

$$\mathbf{x} = \frac{q-1}{2} y_n\beta_0 + (y_1 - y_n)\beta_1 + \cdots + (y_{n-1} - y_n)\beta_{n-1}. \tag{2}$$

A conversion of the representation $\mathbf{x} = y_0\beta_0 + y_1\beta_1 + \cdots + y_{n-1}\beta_n$ from the redundant to the reduced o.n.b. exploits the same formula for β_n and differs from (2) in the first summand: $(y_0 + \frac{q-1}{2} y_n)\beta_0$.

A conversion from the redundant to the permuted o.n.b. may be performed as

$$\mathbf{x} = y_0\beta_0 + y_1\beta_1 + \cdots + y_n\beta_n = (y_1 - 2y_0)\beta_1 + (y_{n-1} - 2y_0)\beta_{n-1} - (y_n - 2y_0)\beta_n.$$

One can easily see that any of the above mentioned conversions imply at most one multiplication by a constant and at most n additions or subtractions in $GF(q)$.

Let us denote $F_t(\mathbf{x})$ a mapping $\mathbf{x} = (x_0, x_1, \ldots, x_{t-1}) \to \mathbf{y} = (y_0, y_1, \ldots, y_{t-1})$ satisfying $x_0\beta_0 + x_1\beta_1 + \cdots + x_{t-1}\beta_{t-1} = y_0\beta^0 + y_1\beta^1 + \cdots + y_{t-1}\beta^{t-1}$. Note, that in the case $t = n$ it defines a conversion from the reduced o.n.b. to the p.b. An efficient computation of the conversion $F_t(\mathbf{x})$ and the inverse conversion $F_t^{-1}(\mathbf{x})$ make use of

Lemma 1 *The following identities hold*

$$\beta_{q^k} = \beta^{q^k}; \beta_i\beta_j = \beta_{i+j} + \beta_{i-j}. \tag{3}$$

Proof

$$\beta_i\beta_j = (\alpha^i + \alpha^{-i})(\alpha^j + \alpha^{-j}) = \alpha^{i+j} + \alpha^{-(i+j)} + \alpha^{i-j} + \alpha^{-(i-j)} = \beta_{i+j} + \beta_{i-j};$$
$$\beta_{q^k} = \alpha^{q^k} + \alpha^{-q^k} = (\alpha + \alpha^{-1})^{q^k} = \beta_1^{q^k} = \beta^{q^k}. \qquad \square$$

Basis conversions in $GF(3^n)$ and $GF(2^n)$ were discussed in [15]. Here, we provide a detailed examination of the field $GF(7^n)$ supporting o.n.b. [16].

Corollary 1 *Let* $0 \le s \le 6$, $0 \le t \le 7^k - 1$, $k \le |\log_7 n|$. *Then the following representations define the same element of the field* $GF(7^n)$:

$$\begin{aligned}\beta^{j\times 7^k}\Bigg(&\sum_{i=0}^{7^k-1} x_{i+j\times 7^k}\beta_i + \sum_{i=0}^{7^k-1} x_{i+j\times 7^k}\beta_i + \sum_{i=0}^{7^k-1} x_{7^k+i+j\times 7^k}\beta_{7^k+i}\\ &+ \sum_{i=0}^{7^k-1} x_{2\times 7^k+i+j\times 7^k}\beta_{2\times 7^k+i} + \cdots + \sum_{i=0}^{t} x_{s\times 7^k+i+j\times 7^k}\beta_{s\times 7^k+i}\Bigg)\end{aligned} \tag{4}$$

and

$$\begin{aligned}\beta^{j\times 7^k}\Bigg(&\sum_{i=0}^{s} y_{i\times 7^k+j\times 7^k}\beta^{i\times 7^k} + \sum_{i=1}^{7^k-1} y_{i+j\times 7^k}\beta_i + \beta^{7^k}\sum_{i=1}^{7^k-1} y_{7^k+i+j\times 7^k}\beta_i\\ &+ \beta^{2\times 7^k}\sum_{i=1}^{7^k-1} y_{2\times 7^k+i+j\times 7^k}\beta_i + \cdots + \beta^{s\times 7^k}\sum_{i=1}^{t} y_{s\times 7^k+i+j\times 7^k}\beta_i\Bigg)\end{aligned} \tag{5}$$

where with the simplified notation $x_{j\times 7^k+i+j\times 7^k} = x_{j,i}, y_{j\times 7^k+i+j\times 7^k} = y_{j,i}$

$$\begin{aligned}&y_{0,0} = 2c(j)(x_{0,0} - x_{2,0} + x_{4,0} - x_{6,0}); y_{1,0} = x_{1,0} - 2x_{3,0} - v_3;\\ &y_{2,0} = x_{2,0} + x_{4,0} + 2v_4; y_{3,0} = v_3; y_{4,0} = x_{4,0} + x_{6,0};\\ &y_{5,0} = x_{5,0}; y_{6,0} = x_{6,0}\end{aligned} \tag{6}$$

where $v_3 = (x_{3,0} + 2x_{5,0})$, $v_4 = (x_{4,0} + x_{6,0})$, $c(j) = 1$ if $j = 0$, $c(j) = 4$ otherwise, *and for* $i = 1, \ldots, 7^k - 1$, $v_{4,i} = x_{4,i} - x_{5,7^k-i}, v_{5,i} = x_{5,i} - x_{6,7^k-i}$

$$\begin{aligned}&y_{0,i} = x_{0,i} - x_{1,7^k-i} - x_{2,i} + x_{3,7^k-i} + v_{4,i} - v_{6,i},\\ &y_{1,i} = x_{1,i} - x_{2,7^k-i} + 2(x_{4,7^k-i} - x_{3,i}) - 4v_{5,i},\\ &y_{2,i} = x_{2,i} - x_{3,7^k-i} + 4v_{4,i} - x_{6,i}, \quad y_{3,i} = x_{3,i} - x_{4,7^k-i} - 4v_{5,i},\\ &y_{4,i} = v_{4,i} + 2x_{6,i}, y_{5,i} = v_{5,i}, y_{6,i} = x_{6,i}.\end{aligned} \tag{7}$$

Now the required conversion $F_n(\mathbf{x})$ can be made up recursively. First, apply the mapping f_k provided by the Corollary 1 *to* $\mathbf{x}$ *with* $k = \lfloor \log_7 n \rfloor - 1$. *Then, split the resulting vector into blocks of length* 7^k *and apply the mapping* f_{k-1} *block-wise. Repeat the same way until* $k = 0$. *This recursion scheme can be viewed as a sequential program*

$$F_n(\mathbf{x}) = f_0(f_1(f_2(\ldots f_m(\mathbf{x})))), m = \lfloor \log_7 n \rfloor - 1$$

where transforms f_i are applied block-wise.

Corollary 2 *Representation* (5) *can be converted to* (4) *via substitution*

$$\begin{aligned} &x_{0,0} = 4(c(j)y_{0,0} - y_{6,0}) + y_{2,0} - v_2), \\ &x_{1,0} = y_{1,0} + y_{3,0} + 2v_{3,0}, x_{2,0} = y_{2,0} + v_{2,0} + y_{6,0}, \\ &x_{3,0} = v_3, x_{4,0} = y_{4,0} - y_{6,0}, x_{5,0} = y_{5,0}, x_{6,0} = y_{6,0}, \end{aligned} \tag{8}$$

where $v_2 = 4y_{4,0}$, $v_3 = y_{3,0} - 2y_{5,0}, c(j) = 1$ if $j = 0$, $c(j) = 2$ otherwise, *and for* $i = 1, \ldots, 7^k - 1$.

$$\begin{aligned} x_{0,i} &= y_{0,i} + y_{1,7^k-i} + 2y_{2,i} - 4(y_{3,7^k-i} + y_{5,7^k-i}) - y_{4,i} - y_{6,i}, \\ x_{1,i} &= y_{1,i} + y_{2,7^k-i} - 4(y_{3,i} - y_{4,7^k-i} + x_{5,i}) + y_{6,7^k-i}, \\ x_{2,i} &= y_{2,i} + y_{3,7^k-i} + 4y_{4,i} - 2y_{5,7^k-i} + y_{6,i}, \\ x_{3,i} &= y_{3,i} + y_{4,7^k-i} - 2y_{5,i} - y_{6,7^k-i}, \\ x_{4,i} &= y_{4,i} + y_{5,7^k-i} - y_{6,i}, x_{5,i} = y_{5,i}, x_{6,i} = y_{6,i}, \end{aligned} \tag{9}$$

The recursive description of the conversion $F_n^{-1}(\mathbf{y})$ is as follows. First, split vector $\mathbf{y}$ *into blocks of length at most* 7 *and apply the transform f_k^{-1} of the Corollary* 2 *block-wise to* $\mathbf{y}$ *with* $k = 0$. *Then, split the resulting vector into blocks of length 7^{k+1} and apply the transform f_{k+1}^{-1} block-wise. Repeat by analogy until* $k = [\log_7 n] - 1$. *A sequential program follows the decomposition*

$$F_n^{-1}(\mathbf{y}) = f_m^{-1}\left(f_{m-1}^{-1}\left(f_{m-2}^{-1}\left(\ldots f_0^{-1}(\mathbf{x})\right)\right)\right), m = [\log_7 n] - 1$$

where transforms f_i^{-1} are applied block-wise.

The complexity of $F_n(\mathbf{x})$ and $F_n^{-1}(\mathbf{y})$ is known to be $O(n \ln n)$ [6]. *We can use the above explicit formulae to derive an accurate estimate. The basic step of the recursion involves at most $n_0 = \lceil \frac{n}{7} \rceil$ conversions $f_0(\mathbf{x})$ (respectively, $f_0^{-1}(\mathbf{y})$) with total complexity $b_{rp}n_0$ (or $b_{pr}n_0$) where from formulae* (6, 8) *$b_{rp} = b_{pr} = 4m_c + 9a$, m_c is the complexity of multiplication by* 2 *or* 4, *and a is the complexity of an additive operation (subtraction or addition) in $GF(7)$.*

A j-th step of the recursion involves at most $n_j = \lceil \frac{n}{7^{j+1}} \rceil$ conversions (7) *of complexity $7^j a_{rp}$ where from formulae* (7) $a_{rp} = 5m_c + 17a$ *and the same number of conversions* (6) *of complexity b_{rp}, that is, totally $n_j \times (7^j a_{rp} + b_{rp})$ additive*

operations in GF(7). *Therefore, the complexity* L_{rp} *of* $F_n(\mathbf{x})$ *is at most* $\sum_{j=1}^{k-1} n_j \times \left(7^j a_{rp} + b_{rp}\right) + c_{rp} n_0$ *Analogously, one can check the complexity* L_{pr} of $F_n^{-1}(\mathbf{y})$ *is at most* $\sum_{j=1}^{k-1} n_j \times \left(7^j a_{rp} + b_{rp}\right) + c_{rp} n_0$, *where* where from formulae (9)$a_{pr} = 6m_c + 21a$. *To obtain an accurate estimate, take* $n = 7^k$. *Then* $n_0 = 7^{k-1}, n_j = 7^{k-j-1}$, *and for* $n = 7^k$

$$\begin{aligned} L_{rp}(n) &\le \sum_{j=1}^{k-1} 7^{k-j-1} \times (7^j a_{rp} + b_{rp}) + c_{rp} n_0 \\ &= \frac{a_{rp}}{7} n \log_7 n - \frac{a_{rp} - c_{rp}}{7} n + \frac{n-7}{42} b_{rp}, \\ L_{pr}(n) &\le \sum_{j=1}^{k-1} 7^{k-j-1} \times (7^j a_{pr} + b_{pr}) + c_{pr} n_0 \\ &= \frac{a_{pr}}{7} n \log_7 n - \frac{a_{pr} - c_{pr}}{7} n + \frac{n-7}{42} b_{pr} \end{aligned}$$

Hence, for arbitrary $n, n \le m = 7^{[\log_7 n]}$, $L_{rp}(n) \le L_{rp}(m)$.

In particular cases one can obtain more accurate estimates. For example, $L_{rp}(29) \le 96a + 22m_c$ *while* $L_{rp}(7^2) \le 191a + 67m_c$.

Furthermore, it makes sense to consider doubled polynomial basis (*d.p.b.*) $1, \beta, \beta^2, \ldots, \beta^{n-1}, \beta^n, \beta^{n+1}, \ldots, \beta^{2n-2}$ *and doubled reduced basis* (*d.r.b.*) $\beta_0, \beta, \beta_2, \ldots, \beta_{n-1}, \beta_n, \beta_{n+1}, \ldots, \beta_{2n-2}$. *Note, that the mapping* $F_{2n-1}^{-1} F_{2n-1}(\mathbf{y})$ *defines a conversion from d.p.b. to d.n.b.*

For example if $n = 29$, *the length of d.p.b. representation is* 57, *hence,*

$$L_{pr}(57) \le 237a + 78m_c \text{ while } L_{pr}\left(7^3\right) \le 2571a + 816m_c.$$

Lemma 2 *The redundant o.n.b. representation* $\mathbf{x} = x_0\beta_0 + x_1\beta_1 + \cdots + x_{n-1}\beta_{n-1} + x_{n-1}\beta_{n-1} + x_n\beta_n$ *can be obtained from the d.r.b. representation* $z_0\beta_0 + z_1\beta_1 + z_2\beta_2 + \cdots + z_{n-1}\beta_{n-1} + z_n\beta_n + z_{n+1}\beta_{n+1} + \cdots + z_{2n-2}\beta_{2n-2}$ *of the same element by the formulae*: $x_0 = z_0, x_1 = z_1$, $x_2 = z_2, x_i = z_i + z_{2n+1-i}, i == 3, \ldots, n-1$.

Proof For any $i = 3, \ldots, n-1$ the identity $\beta_i = \beta_{2n+1-i}$ holds. Indeed,

$$\beta_{2n+1-i} = \alpha^{2n+1-i} + \frac{1}{\alpha^{2n+1-i}} = \alpha^{2n+1-i} + \frac{1}{\alpha^{2n+1-i}} = \alpha^{-i} + \frac{1}{\alpha^{-i}} = \beta_{-i} = \beta_i.$$

The complexity of the stated conversion is $n - 2$.. □

4 Moderate Degree Multiplication in GF(7)[X]

The choice of a multiplication method depends somewhat on the encoding of the ground field $GF(7)$. We suppose the standard encoding via 3-bit unsigned integers. The encoding allows addition and subtraction of the field elements with the bit complexity 17 and virtually free multiplication by a constant. The latter requires just a cyclic shift of bits and/or inversion of the "sign" which can be performed via swapping some additions and subtractions in the surrounded program/circuit (see e.g. [8]). To implement the non-scalar multiplication we suggest the signed encoding (z, b_1, b_0) where $z = 1$ corresponds to the negative sign and $2b_1 + b_0$ expresses an absolute value. It can be easily checked that this encoding allows the non-scalar multiplication of bit complexity 12. Two bit operations suffice to switch between the encodings. Therefore, one needs at most 18 bit operations to perform the multiplication in the standard encoding.

Thus, we seek for a multiplication method that makes use of cheap scalar multiplications and requires a reasonably small total number of additive and multiplicative operations. For moderate values of n one can try a FFT-based method as described in [17]. For simplicity of presentation, put $n = 29$.

A product $p(x) = f(x)g(x)$ of degree-28 polynomials in $GF(7)[x]$ can be repaired from the partial products

$$p_l = fg \mod x^5, \quad p_m = fg \mod (x^{48} - 1), \quad p_h = \lfloor fg/x^{53} \rfloor$$

as

$$p = p_h(x^{48} - 1)x^5 + \left((p_m \bmod x^5) - p_l\right)x^{48} + \lfloor p_m/x^5 \rfloor x^5 + p_l. \tag{10}$$

To calculate p_m we use the FFT of order 48 over $GF(7^2)$. It is convenient to exploit an analogy between the field extension $GF(7^2) \cong GF(7)[i]/(i^2 + 1)$ and the field of complex numbers.

Denote $L_R(FFT_N)$ and $L_C(FFT_N)$ the complexity of FFT of order N with $GF(7)$-valued and $GF(7^2)$-valued arguments respectively.

Lemma 3 $L_R(FFT_{48}) \le 270a + 112m_c$.

Proof The Good-Thomas rule implies

$$L_R(FFT_{48}) \le 16L_R(FFT_3) + 3L_R(FFT_{16})$$

since the order-3 FFT doesn't extend the field of coefficients (roots of order 3 belong to $GF(7)$).

One can easily check that $L_R(FFT_3) \le 6a + 4m_c$. To estimate the complexity of order-16 FFT we adopt a standard recursion for real-argument complex FFTs:

$$L_R(FFT_{4N}) \le L_R(FFT_{2N}) + L_C(FFT_N) + (6N-2)a + (4N-4)m_c.$$

The required upper bound follows from the recurrent relations above and simple bounds $L_C(FFT_2) \le 4a, L_C(FFT_4) \le 16a, L_R(FFT_4) \le 6a$.

The complexity of the inverse transform is estimated exactly the same way.

Let us introduce an additional notation: m stands for the complexity of the non-scalar multiplication in the standard encoding of the ground field $GF(7)$, m' stands for the complexity of the multiplication in the alternative encoding, and t stands for the complexity of conversion between the two encodings. □

Corollary 3 *The complexity of multiplication of polynomials in* $GF(7)[x]/(x^{48} - 1)$ *is at most* $858a + 97m + 336m_c$ *or, alternatively,* $858a + 97m' + 195t + 336m_c$.

Proof To perform the multiplication we use two $GF(7)$-valued FFT's of order 48 and one inverse FFT, 24 multiplications in $GF(7^2)$ and one multiplication in $GF(7)$. Indeed, only 25 coefficients of $GF(7)$-input FFT-images are independent (zero coefficient is in $GF(7)$), others are conjugated with respect to factor-ring $GF(7^2)/GF(7)$. One left to notice that the complexity of $GF(7^2)$ multiplication is at most $2a + 4m$ or $2a + 4m' + 8t$. □

Lemma 4 *The complexity of multiplication of degree-28 polynomials in* $GF(7)[X]$ *is at most* $883a + 122m + 336m_c$ *or* $883a + 122m' + 238t + 336m_c$.

Proof The formula (10) implies 9 additive operations. The least significant coefficients of the product (that is, p_l) can be computed in a straight-forward way as $p_i = f_i g_0 + \cdots + f_0 g_i$ with complexity $10a + 15m$ or $10a + 15m' + 25t$. The most significant coefficients (that is, p_h) can be computed analogously with complexity $6a + 10m$ or $6a + 10m' + 18t$. Add the above estimates to the estimate of Corollary 6. □

The (suitably modified) multiplication method is rather efficient for values of n somewhere between 20 and 35. For instance, due to the record table [18] one can expect the Karatsuba-Toom optimal iteration method requires at least 1044 additive and multiplicative operations to multiply degree-28 polynomials.

5 Arithmetic in GF(7^n)

Multiplication in the reduced o.n.b. representation of the field $GF(7^n)$ can be fulfilled as follows.

1. Convert multiples to p.b.
2. Compute the product by algorithm in Sect. 3.

3. Convert the product from double o.n.b. to double r.n.b.
4. Convert the result to redundant o.n.b.
5. Convert the result to reduced o.n.b.

The complexity of this multiplication algorithm is

$$2L_{rp}(n-1)+M(n-1)+L_{pr}(2(n-1))+2(n-1)-2.$$

Example. For $n = 29$ the complexity of multiplication in $GF(7^n)$ is at most

$$\begin{aligned}&2(96a+22m_c)+883a+122m+336m_c+237a+78m_c+56a-2a\\&\quad=1366a+122m+458m_c\end{aligned}$$

It follows that a bitwise complexity of multiplication in $GF(7^n)$ is at most 26804.

Some modifications of the above scheme may be required:

(a) One of the multiples can be represented in p.b.
(b) On the step 5, the result of step 4 can be converted to permuted o.n.b.

Exponentiation to power q^j in $GF(q^n)$, when an input $\mathbf{x}$ is given in reduced o.n.b. can be fulfilled as follows.

1. Convert $\mathbf{x}$ to permuted o.n.b. by formula (1).
2. Raise to power q^j taking into account permutation π:

$$\mathbf{y}'=\mathbf{x}'^{7^j}=\left(x'_{\pi(\pi^{-1}(1)-j)},\ldots,x'_{\pi(\pi^{-1}(i)-j)},\ldots,x'_{\pi(\pi^{-1}(n)-j)}\right).$$

3. Convert the result to reduced o.n.b. by formula (2).

The first and the third steps have complexity $n - 1$, the second step requires no computations. Hence, the complexity of the above exponentiation algorithm is $2n - 2$.

Remark. If an input $\mathbf{x}$ is given in permuted o.n.b., the complexity of exponentiation to power q^j is $n - 1$.

One can also apply the exponentiation algorithm for extraction of the q-th root.

Example. Consider the computation of the 7-th root of an element $\mathbf{x} \in GF(7^{29})$ represented in reduced o.n.b. It is $\mathbf{x}^{7^{28}}$. The algebraic complexity is at most $54a$. The bitwise complexity is at most 918.

Inversion in reduced o.n.b. of $GF(7^n)$ can be implemented by the generalized in [19] Euclidean binary algorithm [20] with complexity $O(n^t, 2<t<3)$. Let $f(X)$ be irreducible polynomial, root of which generates o.n.b.

1. Convert element $\mathbf{x} \in GF(7^n)$ from reduced o.n.b. to p.b.;
2. $\mathbf{b} \leftarrow 1;\ c \leftarrow 0; \mathbf{u} \leftarrow \mathbf{x}; c = \mathbf{v} \leftarrow f(X);$
3. while deg **u**>0:
   while $u_0 = 0$:
   u← $\mathbf{u}/X$;
   if $b_0 = 0$: **b**← $\mathbf{b}/X$;
   else: **b**←(**b**−$f_0^{-1} b_0 f$(X))/X;
   if deg **u>0:**
   if deg **u**< deg **v**: **u** ↔**v**, **b**↔**c**;
   b← $(-v_0)$**b**+**c**u_0, **u**← $(-v_0)$**u**+**v**u_0.
4. Convert element $u_0^{-1}\mathbf{b}$ from p.b. to reduced o.n.b. and return.

For exponentiation of element $\mathbf{x} \in GF(7^n)$ to an arbitrary power d we use a standard algorithm exploiting the representation of d as a weighted sum $d = = \sum_{i=0}^{\log_7 d} d_i 7^i$. The exponentiation algorithm involves exponentiations to powers q^j (see above) and multiplications with one multiplier given in reduced o.n.b. and second multiplier $\mathbf{a}_1 = \mathbf{x},, \mathbf{a}_2 = \mathbf{x}^2, \ldots, \mathbf{a}_6 = \mathbf{x}^6$ represented in p.b. and the product to be represented in permuted o.n.b.:

b←**1**,j←1;
for i $\overline{0, \lfloor \log_7 d \rfloor}$:
if $d_i = 0$:
j←j+1
else:
$\mathbf{b} \leftarrow \mathbf{b}^{7^j}$, **b**← $\mathbf{b}\mathbf{a}_{d_i}, j = 1$
b← $\mathbf{b}^{7^{j-1}}$
return **b**

6 Conclusions

In this paper, we considered some aspects of fast computations in finite fields $GF(q^n)$ supporting optimal normal bases of 2-d and 3-d types. We put the focus on the basic arithmetic operations: multiplication, exponentiation to a power q^j, and q-th root extraction. A few ways to efficiently implement these operations via switching between several bases of the finite field have been proposed with the case $q = 7$ chosen for illustrations. Note, that fast algorithms for the above basic operations allow to efficiently compute inverses and powers. We expect that the use of a normal representation for exponentiation allows to improve algorithms based on the 7-th root extraction as it is already confirmed for the fields of characteristic 3. The normal representation is also helpful for algorithms based on the point representation of divisors of hyper-elliptic curves.

Acknowledgments This research was supported by the Russian Foundation for Basic Research, project 14-01-00671a.

References

1. Lidl, R., Niderreiter, H.: Finite Fields. Addison-Wesley Publishing Company, London (1983)
2. Jungnickel, D.: Finite Fields: Structure and Arithmetics. Wissenschaftsverlag, Mannheim (1993)
3. Koblitz, N.: Algebraic Aspects of Cryptography. Springer, Berlin (1998)
4. Koblitz, N., Menezes, A.: Pairing-based cryptography at high security levels. In: Proceedings of the Tenth IMA International Conference on Cryptography and Coding/LNCS 3796, pp. 3–36 (2005)
5. Joux, A.: One Round Protocol for Tripartite Diffie-Hellman. LNCS 1838, pp. 385–393 (2000)
6. Baretto, P.S.L.M., Kim, H.Y., Lynn, D., Scott, M.: Efficient Algorithms for Pairing-Based Cryptosystims. Crypto 2002, LNCS 2442, pp. 354–358 (2002)
7. Lee E., Lee H.-S., Lee Y. Fast computation of Tate pairing on general divisors for hyperelliptic curves of genus 3. — Cryptology ePrint Archive, Report 2006/125. — http://eprint.iacr.org/2006/125
8. Bolotov, A.A., Gashkov, S.B., Burtsev, A.A., Zhebet, S.Y., Frolov, A.B.: On hardware and software implementation of arithmetic in finite fields of characteristic 7 for calculation of pairings. J. Math. Sci. pp. 49–75 (2010) (Springer Science + Business Media, Inc., New York)
9. Mullin, R.C., Onyszchuk, I.M., Vanstone, S.A., Wilson, R.M.: Optimal Normal Bases in GF (p^n). Discrete Appl. Math. **22**, 149–161 (1988/89)
10. Bolotov, A.A., Gashkov, S.B.: On quick multiplication in normal bases of finite fields. Discrete Math. Appl. **11**(4), 327–356 (2001)
11. Jamshid Shokrollahi. Efficient implementation of elliptic curve cryptography on FPGA. PhD thesis, universitet Bonn, 2007
12. von zur Gathen, J., Shokrollahi, A., Shokrollahi, J.: Efficient multiplication using type 2 optimal normal bases. In: WAIFI'07, LNCS, pp. 55–68 (2007)
13. Bernstein, D.J., Lange, T.: Type-II optimal polynomial bases. Arith. Finite Fields Proc. Lect. Notes Comput. Sci. **6087**, 41–61 (2010)
14. Duursma, I., Lee, H.-S.: "Tate pairing implementation for hyperelliptic curves" $y^2 = x^p - x + d$. Asiacrypt-2003, LNCS 2894, pp. 111–123 (2003)
15. Gashkov, S., Frolov, A., Lukin, S., Sukhanova, O.: Arithmetic in the finite fields using optimal normal and polynomial bases in combination. In: Advances in Intelligent Systems and Computing. Volume 365. Theory and Engineering of Complex Systems and Dependability. Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCos-RELCOMEX, pp. 153–162, June 29–July 3 2015
16. Gashkov, S.B., Frolov, A.B., Lukin, S.A.: Optimal normal bases of 2-d and 3-d types in finite fields of characteristic 7. MPEI Bull. **1**, 45–49 (2016). (In Russian)
17. Gashkov, S.B.: On fast multiplication of polynomials, the Fourier and Hartley transforms. Discrete Math. Appl. **10**(5), 499–528 (2000)
18. Bernstein, D.J.: Minimum number of bit operations for multiplication. http://binary.cr.yp.to/m.html. Accessed 2009
19. Gashkov, S.B., Frolov, A.B., Shilkin, S.O.: On some algorithms of inversion and division in finite rings and fields. MPEI Bull. N6, 20–31 (2006) (in Russian)
20. Hankerson, D., Hernandez, J.L., Menezes, A.: Software implementation of elliptic curve cryptography over binary fields. Cryptographic Hardware and Embedded Systems, CHES 2000, LNCS 1965, pp. 1–24, Springer (2000)

Stochastic Runge–Kutta Software Package for Stochastic Differential Equations

M.N. Gevorkyan, T.R. Velieva, A.V. Korolkova, D.S. Kulyabov and L.A. Sevastyanov

Abstract As a result of the application of a technique of multistep processes stochastic models construction the range of models, implemented as a self-consistent differential equations, was obtained. These are partial differential equations (master equation, the Fokker–Planck equation) and stochastic differential equations (Langevin equation). However, analytical methods do not always allow to research these equations adequately. It is proposed to use the combined analytical and numerical approach studying these equations. For this purpose the numerical part is realized within the framework of symbolic computation. It is recommended to apply stochastic Runge–Kutta methods for numerical study of stochastic differential equations in the form of the Langevin. Under this approach, a program complex on the basis of analytical calculations metasystem Sage is developed. For model verification logarithmic walks and Black–Scholes two-dimensional model

M.N. Gevorkyan (✉) · T.R. Velieva · A.V. Korolkova · D.S. Kulyabov · L.A. Sevastyanov
Department of Applied Probability and Informatics,
Peoples' Friendship University of Russia, Miklukho-Maklaya Street 6,
Moscow 117198, Russia
e-mail: mngevorkyan@sci.pfu.edu.ru

T.R. Velieva
e-mail: trvelieva@gmail.com

A.V. Korolkova
e-mail: avkorolkova@gmail.com

D.S. Kulyabov
e-mail: yamadharma@gmail.com

L.A. Sevastyanov
e-mail: leonid.sevast@gmail.com

D.S. Kulyabov
Laboratory of Information Technologies, Joint Institute for Nuclear Research,
Joliot-Curie 6, Dubna, Moscow Region 141980, Russia

L.A. Sevastyanov
Bogoliubov Laboratory of Theoretical Physics, Joint Institute for Nuclear Research,
Joliot-Curie 6, Dubna, Moscow Region 141980, Russia

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_15

are used. To illustrate the stochastic “predator–prey” type model is used. The utility of the combined numerical-analytical approach is demonstrated.

Keywords Runge–Kutta methods · Stochastic differential equations · RED queueing discipline · Active queue management · Computer algebra software · Sage CAS

1 Introduction

The mathematical models adequacy may be largely increased by taking into account stochastic properties of dynamic systems. Stochastic models are widely used in chemical kinetics, hydrodynamics, population dynamics, epidemiology, filtering of signals, economics and financial mathematics as well as different fields of physics [1]. Stochastic differential equations (SDE) are the main mathematical apparatus of such models.

Compared with numerical methods for ODEs, numerical methods for SDEs are much less developed. There are two main reasons for this: a comparative novelty of this field of applied mathematics and much more complicated mathematical apparatus. The development of new numerical methods for stochastic case in many ways is similar to deterministic methods development [2–7]. The scheme, that has been proposed by Butcher [8], gives visual representation of three main classes of numerical schemes.

The accuracy of numerical scheme may be improved in the following way: by adding additional steps (multi-step), stages (multi-stage), and the derivatives of drift vector and diffusion matrix (multi-derivative) to the scheme.

Multi-step (Runge–Kutta like) numerical methods are more suitable for the program implementation, because they can be expressed as a sequence of explicit formulas. Thus, it is natural to spread Runge–Kutta methods in the case of stochastic differential equations.

The main goal of this paper is to give a review of stochastic Runge–Kutta methods implementation made by authors for Sage computer algebra system [9]. The Python programming language and NumPy and SciPy modules were used for this purpose because Python programming language is an open and powerful framework for scientific calculations.

The authors have faced the necessity of stochastic numerical methods implementation in the course of work on the method of stochastization of one-step processes [10, 11]. As The RED [10, 12] model was taken as an example of the methodology application in order to verify the obtained SDE models by numerical experiments.

1.1 The Choice of Programming Language for Implementing Stochastic Numerical Methods

The following requirements to computer algebra system's programming language were taken into account:

- advanced tools for manipulations with multidimensional arrays (up to four axes) with a large number of element are needed;
- it is critical to be able to implement parallel execution of certain functions and sections of code due to the need of a large number of independent calculations according to Monte-Carlo method;
- functions to generate large arrays of random numbers are needed;

Computer algebra system Sage generally meets all these requirements. NumPy module is used for array manipulations and SciPy module—for n-point distribution creation.

2 Wiener Stochastic Process

In this section only the most important definitions and notations are introduced. For brief introduction to SDE see [13, 14], for more details see [1, 15, 16].

A standard scalar Wiener process $\mathrm{W(t)}, \mathrm{t} \geq 0$ is a stochastic process which depends continuously on $\mathrm{t} \in [\mathrm{t_0}], \mathrm{T}$ and satisfies the following three conditions:

- $\mathrm{P}\{\mathrm{W}(0) = 0\} = 1$, in other words $\mathrm{W}(0) = 0$ almost surely;
- $\mathrm{W(t)}$ has independent increments i.e. $\{\Delta \mathrm{W_i}\}_0^{\mathrm{N}-1}$—independently distributed random variables; $\Delta W_i = W(t_{i+1}) - W(t_i)$ and $0 \leq \mathrm{t_0} < \mathrm{t_1} < \mathrm{t_2} < \cdots < \mathrm{t_N} \leq \mathrm{T}$;
- $\Delta \mathrm{W_i} = \mathrm{W}(\mathrm{t}_{i+1}) - \mathrm{W}(\mathrm{t_i}) \sim \mathcal{N}(0, \mathrm{t}_{i+1} - \mathrm{t_i})$, where $0 \leq t_{i+1} < t_i < T, i = 0, 1, \ldots, N-1$.

Notation $\Delta \mathrm{W_i} \sim \mathcal{N}(0, \Delta t_i)$ denotes that ΔW_i is a normally distributed random variable with zero mean $\mathbb{E}[\Delta W_i] = \mu = 0$ and unit variance $\mathbb{D}[\Delta W_i] = \sigma^2 = \Delta t_i$.

Multidimensional Wiener process $W^{\alpha}(t) : \Omega \times [t_0, T] \to \mathbb{R}^m, \forall \alpha = 1, \ldots, m$, is consist of independent $\mathrm{W}^1(\mathrm{t}), \ldots, \mathrm{W}^{\mathrm{m}}(\mathrm{t})$ scalar Wiener processes. The increments $\Delta \mathrm{W}_{\mathrm{i}}^{\alpha}$ are mutually independent normally distributed random variables with zero mean and unit variance.

2.1 Itô SDE for Multidimensional Wiener Process

Let's consider a random process $\mathbf{x}(t) = \left(x^1(t), \ldots, x^d(t)\right)^T$, where $\mathbf{x}(t)$ belongs to the function space $L^2(\Omega)$ with norm $\|\cdot\|$. We assume that stochastic process $x(t)$ is a solution of Ito SDE [1, 16]:

$$x^{\alpha}(t) = f^{\alpha}(t, x^{\gamma}(t))dt + \sum_{\beta=1}^{m} G^{\alpha}_{\beta}(t, x^{\gamma}(t))dW^{\beta}, \tag{1}$$

where $\alpha, \gamma = 1, \ldots, d, \beta = 1, \ldots, m$, vector value function $f^{\alpha}(t, x^{\gamma}(t)) = f^{\alpha}\left(t, x^{1}(t), \ldots, x^{d}(t)\right)$ is a *drift vector*, and matrix value function $g^{\alpha}_{\beta}(t, x^{\gamma}(t))$ is a *diffusion matrix*, $W^{\alpha} = (W^{1}, \ldots, W^{m})^{T}$ is a multidimensional Wiener process, also known as a driver process of SDE.

Let's introduce the discretization of interval $[t_0, T]$ by sequence $t_0 < t_1 < \cdots < t_N = \mathrm{T}$ with step $h_n = t_{n+1} - t_n$, where $n = 0, \ldots, N-1$ and $h = max\{h_{n-1}\}_1^N$—minimal step. We also consider the step $h_n = h$ to be a constant; x_n—mesh function for random process $x(t)$ approximation, i.e. $\mathbf{x}_0 = \mathbf{x}(t_0)$, $\mathbf{x}_n \approx \mathbf{x}(t_n) \forall n = 1, \ldots, N$.

2.2 Strong and Weak Convergences of Approximation Function

It is necessary to define the criterion for measuring the accuracy of process $\mathbf{x}(t)$ approximation by sequences of functions $\{\mathbf{x}_n\}_1^N$. Usually *strong* and *weak* [2, 7, 16] criteria are defined.

The sequence of approximating functions $\{\mathbf{x}_n\}_1^N$ converges with order p to exact solution $\mathbf{x}(t)$ of SDE in moment T in *strong sense* if constant $C > 0$ exists and $\delta_0 > 0$, such as $\forall h \in (0, \delta_0]$ the condition (2) is fulfilled.

$$\mathbb{E}(\|\mathbf{x}(T) - \mathbf{x}_N\|) \leq Ch^p \tag{2}$$

The sequence of approximating functions $\{\mathbf{x}_n\}_1^N$ converges with order p to solution $\mathbf{x}(t)$ of SDE in moment T in *weak seance* if constant $C_F > 0$ exists and $\delta_0 > 0$, such as $\forall h \in (0, \delta_0]$ the condition (3)

$$|\mathbb{E}[F(\mathbf{x}(T))] - \mathbb{E}[F(\mathbf{x}_N)]| \leq C_F h^p \tag{3}$$

is fulfilled.

3 Stochastic Runge–Kutta Methods

3.1 Strong Stochastic Runge–Kutta Methods for SDEs with Scalar Wiener Process

In the case of scalar SDE and Wiener process the following scheme is valid:

$$X_0^i = x_n + \sum_{j=1}^{s} A_0^i f\left(t_n + c_0^j h_n, X_0^j\right) h_n + \sum_{j=1}^{s} B_{0j}^i g\left(t_n + c_1^j h_n, X_1^j\right) \frac{I^{10}(h_n)}{\sqrt{h_n}}, \quad (4)$$

$$X_1^i = x_n + \sum_{j=1}^{s} A_{1j}^i f\left(t_n + c_0^j h_n, X_0^j\right) h_n + \sum_{j=1}^{s} B_{1j}^i g\left(t_n + c_1^j h_n, X_1^j\right) \sqrt{h_n}, \quad (5)$$

$$\begin{aligned} x_{n+1} = x_n &+ \sum_{i=1}^{s} a_i f\left(t_n + c^i h_n, X_0^i\right) h_n \\ &+ \sum_{i=1}^{s} \left(b_i^1 I^1(h_n) + b_i^2 \frac{I^{11}(h_n)}{\sqrt{h_n}} + b_i^3 \frac{I^{10}(h_n)}{h_n} + b_i^4 \frac{I^{111}(h_n)}{h_n} \right) g\left(t_n + c_1^i, X_1^i\right) \end{aligned} \quad (6)$$

In Robler preprint [7] there are two realisations of this scheme for $s = 4$. The first scheme we will denote as SRK1W1, the second—as SRK2W2. The method SRK1W1 has strong order $(p_d, p_s) = (2.0, 1.5)$, the method SRK2W1 has strong order $(p_d, p_s) = (3.0, 1.5)$. Another scheme with strong order $p_s = 1.0$ can be found in [1].

3.2 Strong Stochastic Runge–Kutta Methods for SDE System with Multidimensional Wiener Process

For Itô SDE system with multidimensional Wiener process the stochastic Runge-Kutta scheme with strong order $p_s = 1.0$ can be obtained using single and double Itô integrals [7].

$$\begin{aligned} X^{0i\alpha} &= x_n^\alpha + \sum_{j=1}^{s} A_{0j}^i f^\alpha\left(t_n + c_0^j h_n, X^{oj\beta}\right) h_n + \sum_{l=1}^{m} \sum_{j=1}^{s} B_{0j}^i G_l^\alpha\left(t_n + c_1^j h_n, X^{lj\beta}\right) I^l(h_n) \\ X^{ki\alpha} &= x_n^\alpha + \sum_{j=1}^{s} A_{1j}^i f^\alpha\left(t_n + c_0^j h_n, X^{oj\beta}\right) h_n + \sum_{l=1}^{m} \sum_{j=1}^{s} B_{1j}^i G_l^\alpha\left(t_n + c_1^j h_n, X^{lj\beta}\right) \frac{I^l(h_n)}{\sqrt{h_n}}, \\ X^{0i\alpha} &= x_n^\alpha + \sum_{j=1}^{s} A_{0j}^i f^\alpha\left(t_n + c_0^j h_n, X^{oj\beta}\right) h_n + \sum_{l=1}^{m} \sum_{j=1}^{s} B_{0j}^i G_l^\alpha\left(t_n + c_1^j h_n, X^{lj\beta}\right) I^l(h_n) \\ x^{\overline{\alpha}_{n+1}} &= x_n^\alpha + \sum_{i=1}^{s} a_i f^\alpha\left(t_n + c_0^i h_n, X^{oj\beta}\right) h_n + \sum_{k=1}^{m} \sum_{i=1}^{s} \left(b_i^1 I^k(h_n) + b_i^2 \sqrt{h_n} \right) G_k^\alpha\left(t_n + c_1^j h_n, X^{ki\beta}\right) \end{aligned} \quad (7)$$

where $n = 0, 1, \ldots, N-1; i = 1, \ldots, s; \beta, k = 1, \ldots, m; \alpha = 1, \ldots, d$.

There are two realisations of this scheme for s = 3 in Robler preprint [7]. The SRK1Wm method has the strong order $(p_d, p_s) = (1.0, 1.0)$ of convergence and the method SRK2Wm has the strong order $(p_d, p_s) = (2.0, 1.0)$ of convergence.

It is also important to mention, that the deterministic part of methods SRKp1W1 and SRKp2W1 does not essentially influence the error value.

3.3 Stochastic Runge–Kutta Methods with Weak Convergence of P = 2.0 Order

Numerical methods with weak convergence are the best for approximation of characteristics of the distribution of the random process $x^{\alpha}(t)$. Weak numerical method does not require information about the exact trajectory of a Wiener process W_n^{α}, and the random variables for these methods can be generated on another probability space. So we can use distribution which is easily generated on the computer.

For Itô SDE system with multidimensional Wiener process the following stochastic Runge-Kutta scheme with weak order $p_s = 2.0$ is valid [7].

$$
\begin{aligned}
X^{0i\alpha} &= x_n^{\alpha} + \sum_{j=1}^{s} A_{0j}^{i} f^{\alpha}\left(t_n + c_0^j h_n, X^{oj\beta}\right) h_n + \sum_{j=1}^{s}\sum_{l=1}^{m} B_{0j}^{i} G_l^{\alpha}\left(t_n + c_1^j h_n, X^{lj\beta}\right)\hat{I}^{l},\\
X^{ki\alpha} &= x_n^{\alpha} + \sum_{j=1}^{s} A_{1j}^{i} f^{\alpha}\left(t_n + c_0^j h_n, X^{oj\beta}\right) h_n + \sum_{j=1}^{s} B_{1j}^{i} G_l^{\alpha}\left(t_n + c_1^j h_n, X^{lj\beta}\right)\sqrt{h_n},\\
\widehat{X}^{ki\alpha} &= x_n^{\alpha} + \sum_{j=1}^{s} A_{2j}^{i} f^{\alpha}\left(t_n + c_0^j h_n, X^{oj\beta}\right) h_n + \sum_{j=1}^{s}\sum_{l=1,l\neq k}^{m} B_{2j}^{i} G_l^{\alpha}\left(t_n + c_1^j h_n, X^{lj\beta}\right)\frac{\hat{I}^{kl}}{\sqrt{h_n}},\\
\overline{x^{\alpha}_{n+1}} &= x_n^{\alpha} + \sum_{i=1}^{s} a_i f^{\alpha}\left(t_n + c_1^i, X^{ki\beta}\right) h_n + \sum_{i=1}^{s}\sum_{k=1}^{m}\left(b_i^1\hat{I}^{k} + b_i^2\frac{\hat{I}^{kk}}{\sqrt{h_n}}\right) G_k^{\alpha}\left(t_n + c_1^i h_n, X^{ki\beta}\right)\\
&\quad + \mathrm{I}\sum_{i=1}^{s}\sum_{k=1}^{m}\left(b_i^3\hat{I}^{k} + b_i^4\sqrt{h_n}\right) G_k^{\alpha}\left(t_n + c_2^i h_n, \widehat{X}^{ki\beta}\right)
\end{aligned}
\tag{8}
$$

In the weak stochastic Runge–Kutta scheme the following random variables are used:

$$
\hat{I}^{kl} = \begin{cases} \frac{1}{2}\left(\hat{I}^{k}\hat{I}^{l} - \sqrt{h_n}\tilde{I}^{k}\right), k<l,\\ \frac{1}{2}\left(\hat{I}^{k}\hat{I}^{l} - \sqrt{h_n}\tilde{I}^{l}\right), l<k,\\ \frac{1}{2}\left(\left(\hat{I}^{k}\right)^2 - h_n\right), k = l, \end{cases}
\tag{9}
$$

The random variable $\hat{I}^k$ has three-points distribution, it can take on three fixed values: $\left\{-\sqrt{3h_n}, 0, \sqrt{3h_n}\right\}$ with probabilities 1/6, 2/3 and 1/6 respectively. The variable $\tilde{I}^k$ has two-points distribution with two values $\left\{-\sqrt{h_n}, \sqrt{h_n}\right\}$ and probabilities 1/2 and 1/2.

4 Sage SDE Module Reference

Our library is a common python module. To connect it to Sage users simply perform a standard command |import sde|.

The library contains a number of functions for internal use. The names of these functions begin with the double bottom underscore, as required by the style rules for python code PEP8. with the double bottom underscore, as required by the style rules for python code PEP8.

- $(dt, t) = \_time(N, interval = (0.0, 1.0))$—function divides the time interval into Nsubintervals and returns numpy array t with step dt;
- $(dW, W) = \_scalar-$ wiener-process(N,dt,seed = None)—function generates a trajectory of a scalar Wiener process W from N subintervals with step dt;
- $(dW, W) = \_multidimensional\_wiener\_process(N, dim, dt, seed = None)$—similar function for generating multidimensional (dim dimensions) Wiener process;
- $(\mathrm{dW}, \mathrm{W}) =$ _cov_multidimensional_wiener _process$(\mathrm{N}, \mathrm{dim}, \mathrm{dt}, \mathrm{seed} = \mathrm{None})$—another function for generating multidimensional (dim dimensions) Wiener process, which uses numpy.multivariate_normal. $\mathrm{W}^1, \mathrm{W}^2, \ldots, \mathrm{W}^{\mathrm{m}}$ processes;
- $(\mathrm{dt}, \mathrm{t}, \mathrm{dW}, \mathrm{W}) =$ wiener_process $(\mathrm{N}, \mathrm{dim} = 1, \mathrm{interval} = (0.0, 1.0), \mathrm{seed} = \mathrm{None})$—the main function which should be used to generate Wiener process. Positional argument N denotes the number of points the interval (default [0,1]). Argument dim defines a dimension of Wiener process. Function returns sequence of four elements $\mathrm{dt}, \mathrm{t}, \mathrm{dW}, \mathrm{W}$, where dt is a step, $(\Delta \mathrm{t_i} = \mathrm{h} = \mathrm{const}), \mathrm{t}$ is one-dimensional numpy-array of time points $\mathrm{t}_1, \mathrm{t}_2, \ldots, \mathrm{t_N}; \mathrm{dW}, \mathrm{W}$ are also $\mathrm{N} \times \mathrm{m}$ dimensional numpy-arrays, consisting of increments $\Delta \mathrm{W}_{\mathrm{i}}^{\alpha}$ and trajectory points $\mathrm{W}_{\mathrm{i}}^{\alpha}$, where $\mathrm{i} = 1, \ldots, \mathrm{N}; \alpha = 1, \ldots, \mathrm{m}$.
- $\_strong\_method\_selector(name)\_weak\_method\_selector(name)$—set Butcher table for specific method.

The following set of functions are made for Itô integrals approximation in strong Runge-Kutta schemas. All of them get an array of increments `dW` as positional argument. Step size `h` is the second argument. All functions return a list of integral approximations for each point of driving process trajectory. For multiple integrals this list contains arrays.

Now we will describe core functions that implement numerical methods with strong convergence.

- EulerMaruyama(f, g, h, x_0, dW)EulerMaruyamaWm(f, g, h, x_0, dW)—Euler Maruyama method for scalar and multidimensional Wiener processes, where f—a drift vector, g—a diffusion matrix;
- strongSRKW1(f, g, h, x_0, dW, name = 'SRK2W1')—function for SDE integration with scalar Wiener process, where f(x) and g(x) are the same as in above functions; h—a step, x_0—an initial value, dW—an array of Wiener process increments N; argument name can take values 'SRK1W1' and 'SRK2W2';
- oldstrongSRKp1Wm(f, G, h, x_0, dW, name = 'SRK1Wm')—stochastic Runge–Kutta method with strong convergence order p = 1.0 for multidimensional Wiener process. This function is left only for performance testing for nested loops realisation of multidimensional arrays convolution;
- strongSRKp1Wm(f, G, h, x_0, dW, name = 'SRK1Wm')—stochastic Runge–Kutta method with strong convergence order p = 1.0 for multidimensional Wiener process. We use NumPy method numpy.einsum because it gives sufficient performance goal.

5 Stochastic RED Model

We use our library to calculate the numerical approximation for stochastic network queue management protocol RED (Random Early Detection) model [10, 11, 13, 17].

RED was first introduced in [12] and there was a lot of in the future original algorithm improvements, particularly ARED algorithms. Basic principles of operation of all modifications of the algorithm RED are very similar, so it makes sense to talk about the RED family-type algorithms. RED model is defined by the following SDE:

$$\begin{cases} dW(t) = \left(\frac{1}{T(t)} - \frac{W^2(t)P\left(\widehat{Q}\right)}{T(t)}\right)dt + \sqrt{\frac{1}{T(t)} + \frac{W^2(t)P\left(\widehat{Q}\right)}{T(t)}}dV^1, \\ dQ(t) = \left(\frac{W(t)}{T(t)} - C\right)dt + \sqrt{\frac{W(t)}{T(t)} - C}dV^2, \\ d\widehat{Q}(t) = w_q C\left(Q(t) - \widehat{Q}(t)\right). \end{cases} \tag{10}$$

There $W(t)$—TCP window size, $Q(t)$—instant queue length, $\widehat{Q}(t)$—an exponentially weighted moving average queue length, $T(t)$—time dual circulation (during the dual circulation come all the confirmation at the TCP sent box), $P(\widehat{Q})$—function calculating dropping probability package, C—the intensity of service, dV^1, dV^2—Wiener processes, the corresponding random processes $W(t)$ and $Q(t)$.

We use weak Runge–Kutha method to calculate exact realizations of SDE numerical solutions. Based on this data we find time evolution of three variables involved in RED SDE model. Thus from Fig. 1 we can see an evolution of RED

queue characteristics over time, also Fig. 2 illustrates the same behaviour for ARED protocol (modification of RED).

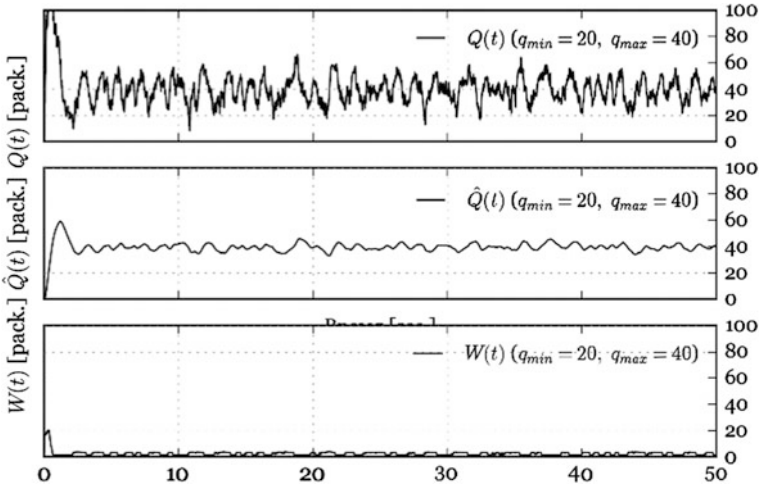


Fig. 1 The solution of SDE for RED protocol

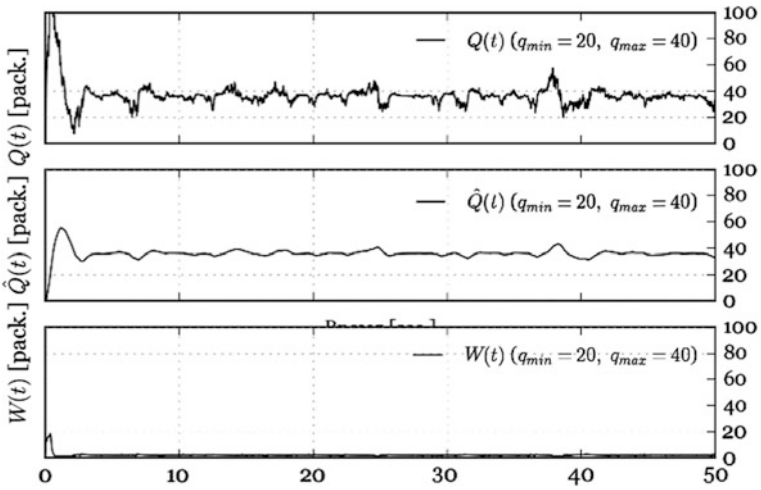


Fig. 2 Solution of SDE for ARED protocol

6 Conclusions

Some realizations of stochastic Runge-Kutta methods were considered in this article. The authors gradually developed and refined the library by adding a new functionalities and optimizing existing ones. To date, the library uses numerical methods by the Rossler's article [7], as the most effective of the currently known to the authors. However, the basic functions strongSRKW1 and weakSRKp2Wm are written in accordance with the general algorithm and can use any Butcher table with appropriate staging. This allows any user of the library to extend functionality by adding new methods.

Additional examples of library usage and of all files source codes are available at https://bitbucket.org/mngev/red-modeling-public. The authors supposed to maintain the library by adding new functions.

Acknowledgments The work is partially supported by RFBR grants No's 14-01-00628, and 15-07-08795, and 16-07-00556.

Calculations were carried out on computer cluster "Felix" (RUDN) and Heterogeneous computer cluster "HybriLIT" (Multifunctional center storage, processing and analysis of data JINR).

References

1. Kloeden, P.E., Platen, E.: Numerical Solution of Stochastic Differential Equations, 2nd edn. Springer, Berlin (1995)
2. Roßler, A.: Runge-Kutta Methods for the Numerical Solution of Stochastic Differential Equations, Ph.D. thesis, Technischen Universit at Darmstadt, Darmstadt (February 2003)
3. Burrage, K., Burrage, P.M.: High strong order explicit Runge-Kutta methods for stochastic ordinary differential equations. Appl. Numer. Math. **22**, 81–101 (1996)
4. Burrage, K., Burrage, P.M., Belward, J.A.: A bound on the maximum strong order of stochastic Runge-Kutta methods for stochastic ordinary differential equations. BIT **37**, 771–780 (1997)
5. Burrage, K., Burrage, P.M.: General order conditions for stochastic Runge-Kutta methods for both commuting and non-commuting stochastic ordinary differential equation systems. Appl. Numer. Math. **28**, 161–177 (1998)
6. Burrage, K., Burrage, P.M.: Order conditions of stochastic Runge-Kutta methods by B-series. SIAM J. Numer. Anal. **38**, 1626–1646 (2000)
7. Roßler, A.: Strong and Weak Approximation Methods for Stochastic Differential Equations—Some Recent Developments, http://www.math.uni-hamburg.de/research/papers/prst/prst2010-02.pdf (2010)
8. Butcher, J.C.: Numerical Methods for Ordinary Differential Equations, 2nd edn. Wiley, New Zealand (2003)
9. Stein, W., et al.: Sage Mathematics Software (Version 6.5), The Sage Development Team, http://www.sagemath.org (2015)
10. Eferina, E.G., Korolkova, A.V., Gevorkyan, M.N., Kulyabov, D.S., Sevastyanov, L.A.: One-step stochastic processes simulation software package, bulletin of peoples' friendship University of Russia. Math. Inf. Sci. Phys. (3), 46–59 (2014). arXiv:1503.07342

11. Velieva, T.R., Korolkova, A.V., Kulyabov, D.S.: Designing installations for verification of the model of active queue management discipline RED in the GNS3. In: 6th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), IEEE, 2014, pp. 570–577. arXiv:1504.02324, doi:10.1109/ICUMT.2014.7002164
12. Floyd, S., Jacobson, V.: Random early detection gateways for congestion avoidance. IEEE/ACM Trans. Netw. **1**, 397–413 (1993)
13. Hnatič, M., Eferina, E.G., Korolkova, A.V., Kulyabov, D.S., Sevastyanov, L.A.: Operator approach to the master equation for the one-step process. EPJ Web Conf **108**, 02027 (2016). doi:10.1051/epjconf/201610802027
14. Higham, D.J.: An algorithmic introduction to numerical simulation of stochastic differential equations. SIAM Rev. **43**(3), 525–546 (2001)
15. Malham, S.J.A., Wiese, A.: An introduction to SDE simulation (2010). arXiv:1004.0646
16. Platen, E., Bruti-Liberati, N.: Numerical Solution of Stochastic Differential Equations with Jumps in Finance. Springer, Heidelberg (2010)
17. Demidova, A.V., Korolkova, A.V., Kulyabov, D.S., Sevastyanov, L.A.: The method of constructing models of peer to peer protocols. In: 6th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT). IEEE, pp. 557–562 (2014). doi:10.1109/ICUMT.2014.7002162

The Assessment Method of the Organization of Municipal Waste Collection Zones

Robert Giel and Marcin Plewa

Abstract The problem of proper organization of waste management systems these days becomes more and more significant. Plans of waste management, assume lowering the number of waste indicated for treatment, encouraging the reuse of waste and creating preferences in order to raise the level of recovery processes, in particular, recycling. Realization of the aforementioned establishments in the following years will be significantly dependent from the proper organization of waste management systems. In the article, the authors presented a method of evaluation of the process of planning of waste collection zones, which uses the simulation model. Assuming that the dependability of the system is its capability to fulfill what is expected of the system, we might assume that, in relation to the subsystem of waste collection, the main measure of evaluation will be the reception of the waste in a timely manner and having required the values of indicators defining its efficiency. The presented method allows one to assign points (places of waste gathering) in already existing zones, in case of which, it should be considered to transfer them to other zones of waste collection. The improvement of zones for waste collection will raise the access of the means of transport and will improve the timeliness of the collection, significantly influencing the level of dependability of the collection subsystem.

Keywords Waste management · Waste collection · Collection zones · Zoning problem · Evaluation method

R. Giel (✉) · M. Plewa
Wroclaw University of Technology, Smoluchowskiego 48, Wroclaw, Poland
e-mail: robert.giel@pwr.edu.pl

M. Plewa
e-mail: marcin.plewa@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_16

1 Introduction

The problem of proper organization of the waste management system is becoming more and more significant these days. The plans of waste management assume lowering the number of waste indicated for treatment, encouraging the reuse of the waste and creating preferences in order to raise the level of recovery processes, in particular, recycling. Realization of aforementioned establishments in the following years will be significantly dependent from the proper organization of waste management systems.

The system of waste management is formed by the following elements:

- subsystem for gathering waste;
- subsystem for waste collection and transport;
- subsystem for waste processing;
- subsystem for waste treatment.

The analysis of the available literature allowed one to separate the models of waste management, according to the criterion of their use, to four main groups. These are:

- models of localization of choice of areas of waste collection/designing routes of waste gathering [1–3];
- models of planning routes of waste collection [4–6];
- models of the processes of stock management and recycling planning [1, 7, 8];
- models with service of planning of the systems of waste management, including the use of the system of decision support/expert systems, tools of evaluation in the research of the functioning of the system and case studies.

The model presented in this article concerns the subsystem of waste gathering. Modeling of the processes of waste gathering and waste transport is broadly analyzed in the literature. The most commonly described problem in the literature is the planning of the routes for the vehicles, realizing the collection and methods of designing service zones. Available publications are focused on specifying the order of service of the certain points of waste gathering and the optimization of use of the available resources (means of transport, human resources). In the article [9], there has been presented a method of planning routes in India. The method was concerned with the proper choice of vehicles for the realization of the collection process. Moreover, we might find similar problems in [10, 11]. The history of waste transportation might be found in [12].

The literature lacks the extensive methods and models for evaluating the systems of waste collection and transport. Available models assume the evaluation of the system on the basis of simple indicators like the quotient of the gathered waste mass and number of travelled kilometers, the volume of gathered waste in a unit of time and the time needed for service of the specific zone etc. These are, above all, controlling indicators allowing one to evaluate decisions made at a previous stage of the process. In available literature, there is a lack of the models of evaluation,

which could be implied on the stage of planning of service zones and would support the planner in making decisions about the allocation of available resources.

The aim of the article is to work out the method of evaluation, supporting the planning process for collecting municipal waste. The introduced method consists in the evaluation of the process of connecting the particular points of waste gathering to form zones serviced by single vehicles. This method is supposed to support the planner in the evaluation of existing zones, taking into account the level of concentration of the points located in the zone. Thanks to this method, the planner will be able to find points, which significantly influence the worsening of the indicators describing the process of collection realization. The problem of waste collection, considering a very high possible number of combinations, is a NP-Hard problem [13]; that is why, in the developed method, the authors use computer simulation, similar to [14–17].

2 Municipal Waste Collection and Transport System Subsystem Model

This chapter presents a municipal waste collection subsystem, on the basis of which, the method of evaluation is developed.

The subsystem of collection and transport of waste is specified by the set of elements E (the set of containers, set of vehicles realizing the collection and days of waste removal) having certain properties (attributes) A (type of container, method of access to the container, type of building development in the container zone, localization of the container, limitations and requirements relating to the time of waste removal, type of the vehicle, etc.) and relations between them—R. The job of the planner is to assign the particular containers to vehicles and also assigning the days for waste reception.

The schedule of waste removal for particular vehicles is a three-argument relation between sets E1, E2 and E3, where:

- E1 is a set of all containers,
- E2 is a set of all vehicles,
- E3 is a set of days for waste removal.

The service zone is defined as a first domain of 3-element relation describing the waste containers of a given fraction, removed by a given vehicle, in a given day, with the assumption, that each of the vehicles services maximum of one zone every day and collects only one code of waste at the same time.

The vehicle visits particular points of waste gathering (N^{WCP}), located in the zone, in a set order (i = 1, 2, 3, … N^{WCP}). The distance travelled by the vehicle is defined as directed acyclic graph, where vertexes are consecutively visited points. The distance has a starting point (i = 0) and finishing point (i = $N^{WCP} + 1$), in which there are no containers and which constitute the driving in and out of the

zone. The points for gathering waste (WCP) consist of containers characterized by the same value of the geographic location attribute. The point of gathering waste may be assigned with more than one address of the participating property, and every address may be assigned with more than one container. In each of the points for waste gathering, there is specified number of containers for a given waste code (lp_i^k) with volume ($c_{j,k}^{p,i}$) expressed in cubic meters. The total volume of waste gathered in *i*-th PGO is described by the following expression:

$$C_i = C_{i-1} + c_i[m^3], \text{ for i-th} = 1, 2, 3, \ldots, \tag{1}$$

where:

c_i—volume of waste collected from *i*-th WCP expressed by the following equation:

$$c_i = \sum_{j=1}^{lp_i^k} c_{j,k}^{p,i} \cdot d^{zg}[m^3], \tag{2}$$

where:

lp_i^k—number of containers in *i*-th WCP for the given fraction;

$c_{j,k}^{p,i}$—volume j of container in *i*-th WCP;

d^{zg}—rate of volume reduction rate for the given fraction;

The vehicle is coming back to the base if:

$$C_i \geq C^{dop} \text{ or } i = N^{WCP},$$

where:

C^{dop}—permissible load capacity of the vehicle expressed in m^3;

N^{WCP}—number of WCP intended to be serve;

Total collecting time to *i*-th WCP:

$$T_i = T_{i-1} + t_i^{WCP}, \text{ for } i = 1, 2, 3, \ldots, N^{WCP} + 1, \tag{3}$$

where:

T_0—collection zone entry time;

$T_{N^{WCP}+1}$—collection zone exit time;

t_i^{WCP}—service time for *i*-th WCP expressed by equation:

$$t_i^{WCP} = t_{i-1,i}^{p} + \sum_{j=1}^{lp_i^k} t_j^{ob,i}, \tag{4}$$

where:

$t_j^{ob,i}$—service time for *j*-th container in *i*-th WCP;

$t^{p}_{i-1,i}$—driving time from i–1 to *i*-th WCP expressed by equation:

$$t^{p}_{i-1,i} = s_{i-1,i} \cdot t^{km}_{i,T^a}, \tag{5}$$

where:

$s_{i-1,i}$—distance between i−1 and *i*-th WCP;

t^{km}_{i}—driving time per 1 km in the area of *i*-th WCP.

The efficiency of realization of the process for waste collection may be specified using the indicators describing the quotient of the volume of collected waste and travelled kilometers and the quotient of the volume of collected waste and the duration of collection.

$$W_i^{CS} = \frac{C_i}{\sum_{j=1}^{j=i} s_{j-1,j}}, \tag{6}$$

$$W_i^{CT} = \frac{C_i}{T_i}, \tag{7}$$

3 The Method of Evaluation of the Process of Collection Zone Planning

3.1 The Level of Concentration of the Waste Gathering Points

In this chapter, we present the algorithm (Fig. 2) that uses the method of center of gravity, to determine the level of concentration of the points of waste gathering in the collection zone. During determining the center of gravity, it was assumed that the quantity of waste in particular places of gathering is equal to the volume of the containers in these places. The presented method allows one to evaluate the possibility of improving the specific zone of gathering, through identification of the waste gathering points that have a significant influence on lowering the level of concentration of the zone and should be transferred to another zone (Fig. 1). The level of concentration may also be raised by an addition of points from neighboring collection zones. During the transfer of points, it is necessary to take into account, among others, the level of usage of the means of transport servicing analyzed zones and the requirements relating to the time of receipt. Developing an algorithm of transferring points between zones will be the subject of further work. At this stage, the decisions relating to points' transfers are left to the planner.

The first step of algorithm (Fig. 2) is to appoint a center of gravity considering the number of gathered waste for all of the points from which the receipt is realized. The next step is calculating the average value and standard deviation of the distance from the center of gravity for each of the points. On the basis of this calculation, we

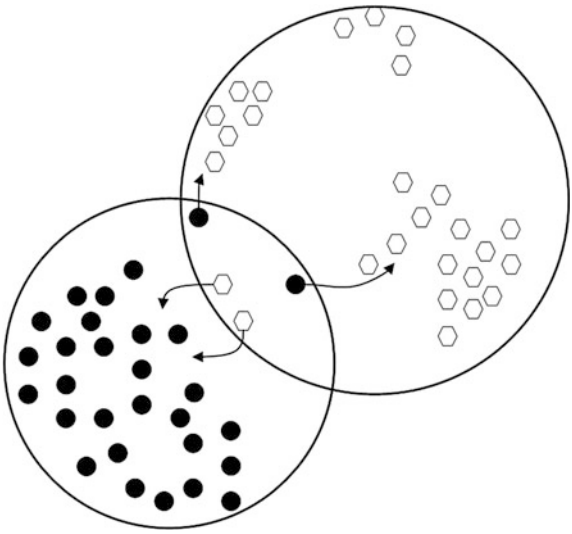


Fig. 1 Scheme of interaction between waste collection zones

determine the value of the coefficient of the variation. The variation coefficient describes the level of concentration—level of aggregation of waste gathering points inside the zone. Another step is to cut off the farthest points from the center of gravity, which significantly influence the lowering of the level of concentration. Iterative calculation of the variation coefficient allows one to determine the influence of particular points on the level of concentration/aggregation of the zone.

If cutting off consecutive points raises the level of concentration, it means that the cut off points were located outside of the concentration, and the receipt of waste from these points could negatively influence the value of the efficiency indicators for the process of waste collection. Cutting off the points should be stopped at the moment in which the variation coefficient starts rising or cutting off the following points does not significantly influence the level of concentration of the zone.

Application of the algorithm allows one to identify points, which significantly interfered with the structure of the collection zone. These points (if possible) may be transferred to a neighboring zone.

3.2 Simulation Model

In the previous chapter, the method of evaluation of the concentration level of collection zones was presented. The following stage was to develop an algorithm using computer simulation and developed a method for optimization of the zone for waste collection. The use of simulation allows one to determine the efficiency indicators of the process for the defined collection zone. The developed simulation model is based on following assumptions:

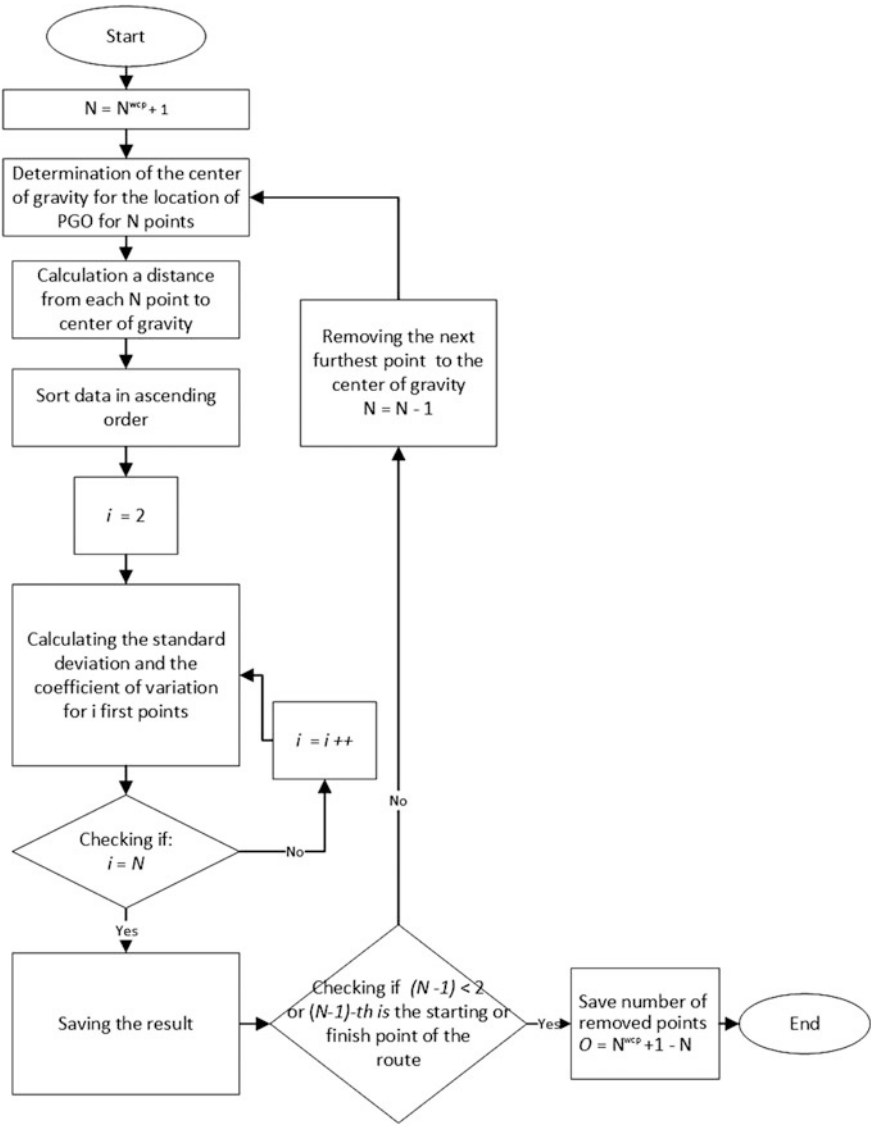


Fig. 2 Algorithm for calculating the coefficient of variation for further scenarios

- C^{dop}—permissible load capacity of the vehicle expressed in m^3 is unlimited.
- lp_i^k—number of containers is equal 0 for i = 0 and i = $N^{WCP} + 1$.
- $s_{i-1,i}$—distance between i−1 and *i*-th WCP is determined as a product of routes curvature coefficient r_{cc} and geographical distance between points.

- The order of visiting points is set and does not undergo any changes after removing the chosen gathering places. After removing a point from the route, the neighboring points are connected.

The algorithm (Fig. 3) presents the process of route optimization with the use of the developed method. The process starts with determining the variation coefficients for the evaluated zone, for the following cut off points located on the route. The algorithm uses the simulation model to analyze the changes of the collection zone structure resulting from the process of the method presented in (Fig. 2). Analysis is conducted on the changes of the value of W^{ct} indicator for the following simulations of waste collection process. The aim is to receive the zone with the highest possible level of concentration and high efficiency of collection process.

The algorithm in (Fig. 4) presents the process of simulation used to investigate the influence of the elimination of waste gathering points, significantly influencing the level of concentration of the zone, on the W^{ct} indicator value. FlexSim software was used to build the simulation model.

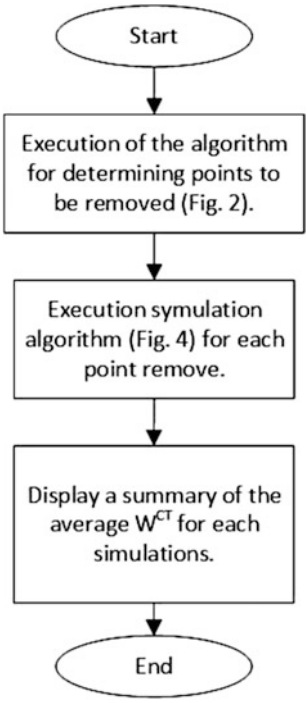


Fig. 3 Algorithm of optimization of the waste collection route

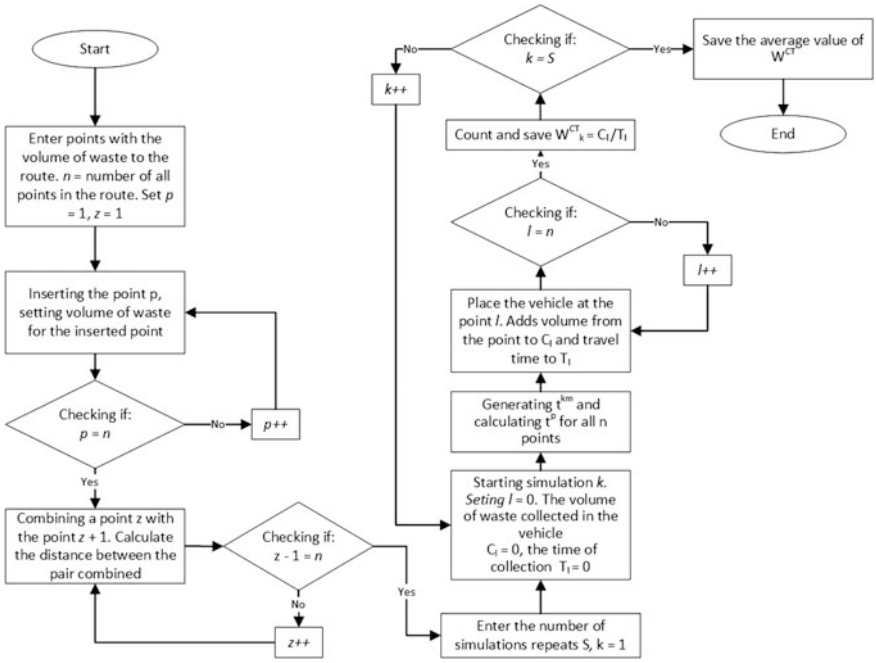


Fig. 4 Algorithm of the process of simulation to determine the efficiency in the following scenarios of the realization of the waste collection

3.3 Model Sensitivity Analysis

In order to determine the influence of the level of concentration on the indicator describing the efficiency of the collection process, an analysis of simulation model sensitivity has been conducted. In addition, the influence of the r_{cc} factor upon the received result has been researched. With regard to the analysis of sensitivity, we chose a zone consisting of 30 points. Arrangement of the points is presented on (Fig. 5). The entrance and exit of the zone are in the same place as the first and the last of waste gathering points. Those points were marked with arrows on the graph. The center of gravity is marked with the letter X.

- $N^{WCP} = 30$
- d^{zg}—rate of volume reduction rate for the given fraction is equal to 1.
- $t_j^{ob,i}$—service time for *j*-th container in *i*-th WCP is equal to 0.
- lp_i^k—number of containers in *i*-th WCP for the given fraction is equal 0 for $i = 0$ and $i = N^{WCP} + 1$
- t_i^{km}—is a random variable described by exponential distribution for average speed of 20 km/h.

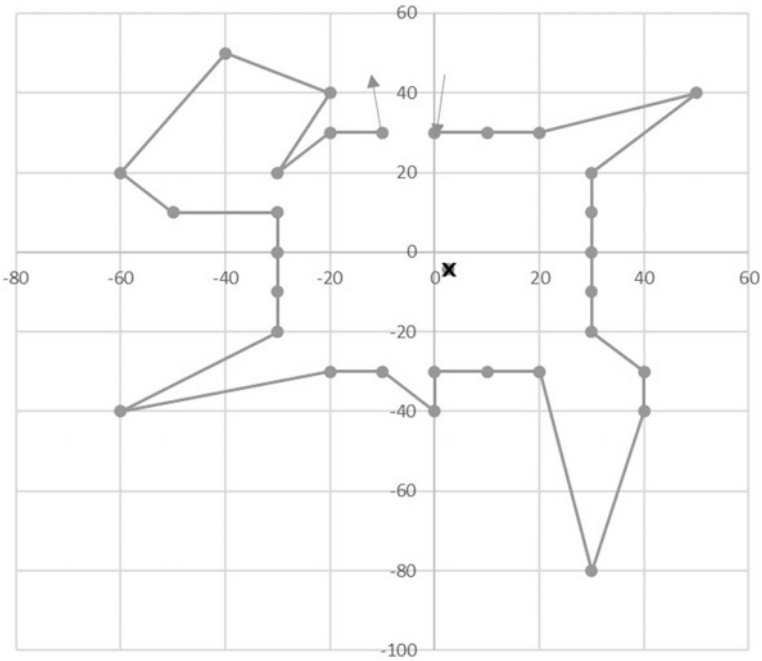


Fig. 5 Theoretical route of collection with 30 points (before realization of the method)

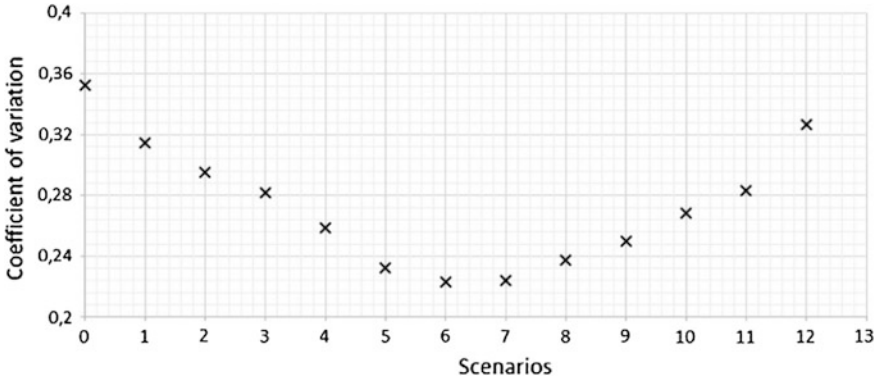


Fig. 6 Coefficients of variation for the routes after the removal of consecutive points

According to the algorithm presented in the figure (Fig. 3), the variation coefficients for the following scenarios were determined. The graph (Fig. 6) presents the results of the analysis of the influence of elimination of particular points on the value of the variation coefficient describing the level of zone concentration. On the cut off axes, there are presented (from the left) following scenarios of cutting off

points. The lowest value of the variation coefficient (the highest level of zone concentration) was observed in the seventh scenario. Cutting off the following points resulted in an increase in the variation coefficient.

In the (Fig. 7), we present the result of analysis of the influence on the level of zone concentration and the influence of the actual distance between points on the value of W^{ct} (Volume per hour) indicator, describing the efficiency of the waste collection process. The highest value of the indicator was obtained as a result of cutting off seven points (value on the graph represented by an arrow). The conducted analysis of sensitivity also showed a significant influence of the difference of actual and geographic distance between points, on the value of W^{ct} indicator. Through the analysis of change of the factor r_{cc}, the developed simulation model allows to take under consideration the type of building development in the moment of setting of the collection zones. It is important in the case of mixed building development. Points located in the zone of multi-family building development will be characterized by considerably bigger difference of the actual and geographic distance. It may have a significant influence on the decision about detaching from the zone, points that lower the level of geographic concentration, which in reality will not be located significantly further from the rest. Further work on the model

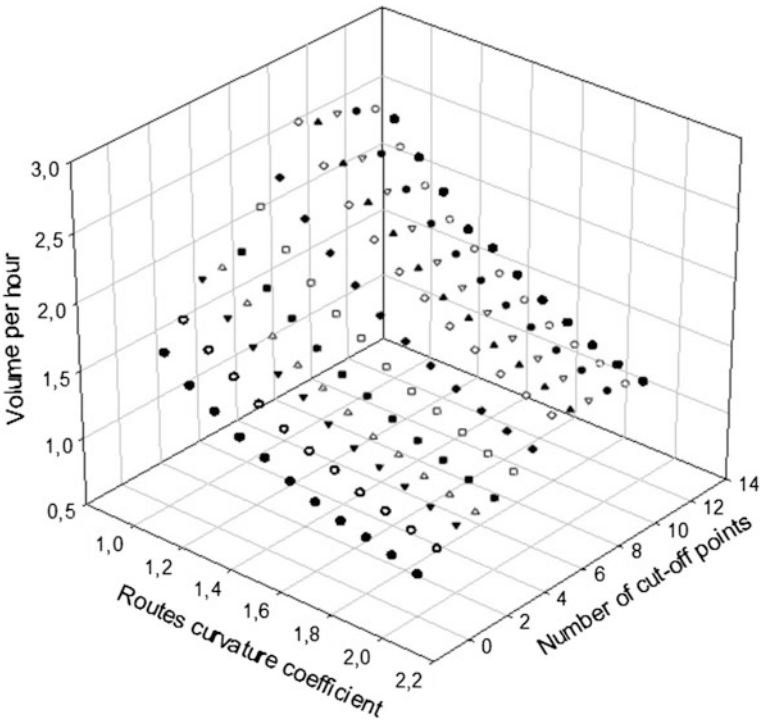


Fig. 7 The change of efficiency while cutting off following points/analysis of sensitivity

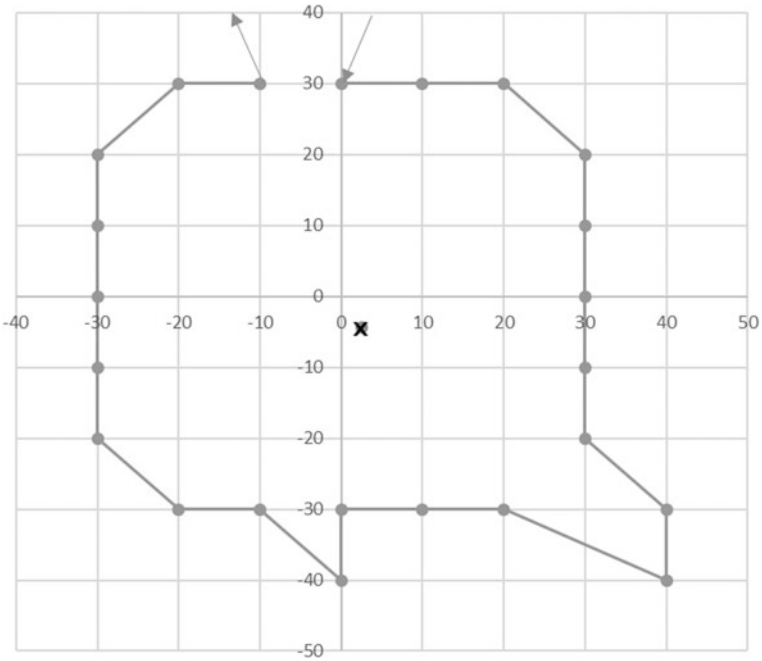


Fig. 8 Theoretical route after using the method of evaluation of three-dimensional route planning

take into consideration a number of zones, level of use of the means of transport servicing the analyzed zones, requirements relating to the time and date of receipt of the waste and economic factors relating to the waste collection. Analysis of sensitivity was finished after cutting off 13 points, it happened this way because the algorithm appointed the initial point for cutting off.

As a result of suggested method, it was possible to determine that the optimized collection zone should contain twenty-three of thirty presented points. The route obtained as a result of the method is presented in the figure below (Fig. 8).

4 Conclusions

The aim of the article was to present the method of the evaluation of process of municipal waste collection zone planning. The method may be used by a planner to evaluate the already existing collection zones and determine the points that are significantly interfering with the concentration of waste gathering points in the zone. The article defines the model of subsystem of collection and transport of waste and presents an algorithm of the method of evaluation. In the following section, the article presents a simulation model that is used to verify planned method. On the basis of the conducted analysis of sensitivity, conducted on the

model, it has been shown that there is a significant influence of a change of the level of concentration of serviced points, on the value of indicator describing the efficiency of collection process. Thereby, the efficiency of described method is confirmed.

The analysis of sensitivity also showed that there is an influence, of a change of actual distance between the waste gathering points, on the results.

Further work will be devoted to expand the method with transferring the points between particular collection zones. The fundamental flaw of the method at the current stage is leaving the decision about what to do with the points to the planner.

References

1. Aringhieri, R., Bruglieri, M., Malucelli, F., Nonato, M.: An asymmetric vehicle routing problem arising in the collection and disposal of special waste. Electron. Notes Discrete Math. 41–47 (2004)
2. Costi, P., Minciardi, R., Robba, M., Rovatti, M., Sacile, R.: An environmentally sustainable decision model for urban solid waste management. Waste Manag. **24**, 277–295 (2004)
3. El Saadany, A., Jaber, M.: The EOQ repair and waste disposal model with switching costs. Comput. Ind. Eng. **55**, 219–233 (2008)
4. Bautista, J., Pereira, J.: Modeling the problem of locating collection areas for urban waste management. An application to the metropolitan area of Barcelona. Omega. **34**, 617–629 (2006)
5. Dekker, R., Fleischmann, M., Inderfurth, K., Van Wassenhove, L.: Reverse Logistics, Quantitative Models for Closed-Loop Supply Chains. Springer, Belin Heidelberg (2004)
6. Inderfurth, K.: Simple optimal replenishment and disposal policies for a product recovery system with leadtimes. OR Spektrum **19**, 111–122 (1997)
7. Andrews, A., Gregoire, M., Rasmussen, H., Witowich, G.: Comparison of recycling outcomes in three types of recycling collection units. Waste Manag. **33**, 530–535 (2013)
8. Ayininuola, G., Muibi, M.: An engineering approach to solid waste collection system: Ibadan North as case study. Waste Manag. **28**, 1681–1687 (2008)
9. Ghose, M., Dikshit, A., Sharma S.K.: A GIS based transportation model for solid waste disposal—a case study on Asansol municipality. Waste Manag. (2006)
10. Apaydin, O., Gonullu M.T.: Emission control with route optimization in solid waste collection process: a case study. Sadhana **33**, 71–82 (2008)
11. Tung, D., Pinnoi, A.: Vehicle routing–scheduling for waste collection in Hanoi. Eur. J. Oper. Res. **125**, 449–468 (2000)
12. Lewandowski, K.: Kontenerowe systemy transportowe. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław (2015)
13. Zajac, M., Swieboda, J.: An unloading work model at an intermodal terminal. In: Theory and Engineering of Complex Systems and Dependability (2015)
14. Kierzkowski, A., Kisiel, T.: The simulation model of logistic support for functioning ground handling agent, taking into account the probabilistic time of aircrafts arrival. In: Carpathian Logistics Congress—Congress Proceedings, pp. 463–469 (2013)
15. Kierzkowski, A., Kisiel, T.: Functional readiness of the check-in desk system at an airport. In: Theory and Engineering of Complex Systems and Dependability: Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, Brunów (2015)

16. Kierzkowski, A., Kisiel, T.: Functional readiness of the security control system at an airport with single-report streams. In: Theory and Engineering of Complex Systems and Dependability: Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, Brunów (2015)
17. Tubis, A., Werbinska-Wojciechowska, S.: Inventory management of operational materials in road passenger transportation company—case study. Carpathian Logistics Congress—Proceedings, Cracow (2013)

NuSMV Model Verification of an Airport Traffic Control System with Deontic Rules

Paweł Głuchowski

Abstract The main cause of major accidents in airport traffic is an incursion of a vehicle into a runway of a landing or taking-off aircraft. This article presents a method to increase the runway safety, where it strongly depends on human made decisions, regulated by national and local laws of deontic nature, i.e. rules of obligation, permission etc. We propose to model and verify such a communication control system, and related behavioral deontic rules, as finite state automata in the Symbolic Model Verifier NuSMV, where the deonticity is built in the automata, and the verification thereof uses CTL temporal logic formulas. The method is simple in modeling the system and in specifying and verifying it. It can also easily find a possible path of states leading to a user-defined hazard.

Keywords Safety-related system · System model verification · Airport communication control · Deontic rules · CTL temporal logic · NuSMV

1 Introduction

Airport traffic control is safety-related. As mentioned in [1, 2], an often cause of a hazardous situation is imperfection of safety procedures, where the rules to obey (by a human or a machine) are unsound or do not meet a real situation in a way to prevent or repair from a hazard. Such rules have deontic nature, i.e. they specify obliged, permitted, or forbidden actions to be done in presence of given events. Had they been obeyed, they would prevent a hazardous consequence of the cause events.

It is important to prove, that such a deontic rule, once introduced into the control system: it will not contradict the already implemented rules of behavior of the system; it will not obstruct the system from a correct and safe behavior; and it will not allow the hazard to occur. More on this subject may be found in [1, 3].

P. Głuchowski (✉)
Department of Computer Engineering,
Wrocław University of Technology, Wrocław, Poland
e-mail: pawel.gluchowski@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_17

The main cause of major accidents in airport traffic is an incursion of a vehicle (aircraft, car, machine, etc.) into a runway of a landing or taking-off aircraft. Different preventing and alerting systems have been implemented, using i.a. stop light bars; electric, magnetic or sound sensors; etc. [4]. There is a need for preventive measures to increase safety, where it depends on human made decisions, regulated by deontic national and local laws. Thus, a method considering deontic rules of behavior in the airport traffic should be implemented. Such approaches are presented i.a. in [5] on deontic security policies for safety protocols; in [3] on an anticipatory prevention system ARIPS, using deontic instructions in presence of alerts; and in [6] on deontic regulations of China aviation, used to predict future safety or hazard states.

In this article we propose to model and verify such a communication control system, with related behavioral deontic rules, as finite state automata in the Symbolic Model Verifier NuSMV [7], by writing and executing a.smv script of a certain form. The deonticity will be built in the automata, instead of using any deontic logic, which NuSMV lacks. The verification will be done using CTL temporal logic formulas [8].

Organization of the article: Sect. 2 explains the NuSMV and organization of its typical script. Section 3 presents a case study of an airport runway control system. Section 4, the main part, presents how to model such a system with deontic rules of its behavior, and how to verify it in the NuSMV. Section 5 contains conclusions.

2 Symbolic Model Verifier NuSMV

NuSMV (New Symbolic Model Verifier) is a tool to model a system as finite state automata, and to automatically verify satisfaction of its properties in temporal logic.

A system model is a state transition system, where state variables are declared by their finite set of values; there are assignment and/or logic expressions, specifying: initial states, invariants and (deterministic or not) value changes of variables. A system may be modeled as a module (automaton), or every its part or actor—as an individual module. The code below presents an example, where comments begin with.

```
MODULE main
  VAR m : actor; --an instance 'm' of a module 'actor'
MODULE actor
  VAR role : {active,passive,undecided}; --a variable
  INIT role = active --initial value of 'role'
  TRANS next(role) in case --transition of the 'role'
    role = active : passive --the next 'role'
    --is 'passive', if it is 'active' now
    role = passive : {active,passive} --the next 'role'
    --is randomly chosen, if it is 'passive' now
  esac;
```

A property of a system can be expressed as a temporal logic formula (i.a. CTL), or an invariant expression, or an expression calculating the minimal or maximal path from one specified state to another. The property may be verified to be true or false. The code below presents examples, where: (1) a *role* of a module *m* may finally be *active*; (2) a *role* of *m* is invariantly *active* or *passive*; (3) the minimal path, from a state where a *role* of *m* is *active*, to a state where it is *passive*, is calculated.

```
1 CTLSPEC EF(m.role = active)
2 INVARSPEC (m.role in {active,passive})
3 COMPUTE MIN[m.role = active, m.role = passive]
```

For a restricted space, let us only explain the meaning of CTL formulas used in this article. A formula may be annotated with a pair of operators. The first declares a possibility (E) or a necessity (A) of the formula to be true; the second declares the formula to be true i.a.: in some state (F) or in all states (G) in the future. E.g.: AG (*f*) states that the formula *f* is true in all states on all paths (informally: *f* certainly is and will always be true); EF(*f*) − *f* is true in some state on some path (*f* possibly is or will finally be true). Notice, the informal understanding of CTL formulas is deontic.

It is possible to automatically find a path of states leading to a state, described by certain values of a module variables, by stating a meant-to-be false property, that such a state will certainly never be reached. If it can be reached in the model, a path to it will be presented. The code below presents an example, where verification of the formula results in a path leading to the state described by *m.role* = *undecided*, if it exists.

```
CTLSPEC AG!(m.role = undecided)
```

Hence, an airport communication system model analysis in the NuSMV consists of: modeling actors of the system with their behavior as modules interacting with each other; specifying the behavior formally in CTL; verifying the properties; and searching for any violation thereof. More on using NuSMV can be found in [7, 9, 10].

3 Case Study of an Airport Runway Control System

Let a case study be a rejected take-off of an aircraft to avoid a collision with a vehicle (named “intruder”) on the aircraft’s runway. Such a rejected take-off is performed once in 3000 flights, when there is a “grave hazard”, as said in [11].

The place. As presented in Fig. 1, a runway crosses a taxiway. The runway is 3 km long, and intersects the taxiway after 1.2 km on its 70 m section. The controlled section of the taxiway is 20 m long on both sides of the runway.

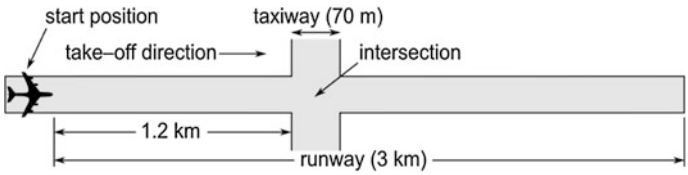


Fig. 1 The case study situation on the airport

The actors. There are 3 actors: an aircraft Boeing 747-8 on the runway, an intruder (a car-sized vehicle) on the taxiway, and a human controller in a control tower. The aircraft is on the start position of the runway. In a normal situation, while speeding up to take off, it moves to the right side of the runway (see Fig. 1), and some time after the velocity V1 (the highest allowed to safely reject a take-off) is reached, it takes off. If the take-off procedure is rejected, the aircraft speeds down to safely stop. It may be rejected when the pilot receives an appropriate order from the controller, or when he sees the intruder approaching the intersection. At any moment the intruder may drive into the controlled section of the taxiway, moving toward the intersection, leading to a hazard of collision. The situation is observed (by eyesight and by alarms from sensors) and controlled by the controller, who sends orders (i.a. to stop) to the aircraft and the intruder by voice (radio) and by setting light bars on the runway and taxiway.

Used symbols. The aircraft, the intruder and the controller are denoted by indexes a, i and c, respectively. A moment (a time instance), is denoted by t, e.g. t_0. Road passed until t is denoted by $s(t)$, e.g. $s_a(t_0)$. Velocity of the aircraft at t is denoted by $v_a(t)$, e.g. $v_a(t_0)$. Velocity of the intruder is denoted by v_i. Acceleration and deceleration of the aircraft are denoted by a_{a1} and a_{a2}, respectively. Deceleration of the intruder is denoted by a_i. Length of reaction time is denoted by τ (will be explained later).

The safe scenario. At t_0 the aircraft starts moving, constantly accelerating to V1 (v_a). It reaches V1 at t_1, and later takes off (detaches its front wheels) at t_2. Based on real cases we assume, that at $t_2 = 45$ s the aircraft takes off, heaving passed $s_a(t_2) = 2$ km of the runway. We also assume, that $v_a(t_1) = 270$ km/h.

The unsafe scenario. At t_0 the aircraft starts moving, constantly accelerating to V1 (v_a). Before it reaches V1, at t_5 the intruder starts approaching and is noticed by the controller and by the aircraft. Then the controller signals the aircraft and the intruder to stop after reaction time τ_{c1} and τ_{c2}, respectively. In reaction to noticing the intruder (after reaction time τ_{a1}) or in reaction to the signal received (after reaction time τ_{a2}) the aircraft starts to constantly decelerate at t_3, and stops at t_4. In reaction to the signal received (after reaction time τ_i) the intruder starts to constantly decelerate at t_8, and stops at t_9. Had the intruder not stopped, it would enter the intersection at t_6. Based on real cases we assume, that: if the aircraft starts to decelerate at $t_3 = t_1$ (the latest possible), it stops at $t_4 = t_1 + 20$ s. We assume: $v_a(t_1) = 270$ km/h; reaction times are: $\tau_{c1} \in [2\text{ s}, 6\text{ s}]$, $\tau_{c2} \in [1\text{ s}, 4\text{ s}]$, $\tau_{a1} \in [1\text{ s},$

4 s], τ_{a2} = 1 s, τ_i = 1 s; the intruder's velocity v_i = 18 km/h; its total deceleration road is 2.5 m. For the intruder we calculate t_6 = 40 s.

The dynamics. Heaving assumed the above, using physic laws for linear motion, let us give dynamics of the actors, where the road is given in m, and time in s. The accelerating aircraft: $v_a(t_1) = 75$, $a_{a1} \approx 1.975$, $v_a(t) \approx 1.975$t, $s_a(t) \approx 0.988t^2$. The decelerating aircraft: $a_{a2} \approx -3.750$, $v_a(t) = 5.725t_3 - 3.750t$, $s_a(t) \approx -2.863t_3^2 + 5.725tt_3 - 1.875t^2$. The constantly moving intruder: $v_i = 5$, $s_i(t) = 5t$. The decelerating intruder: $a_1 = -5$, $s_i(t) = s_i(t_8) - 2.5t^2 + 5t + 5tt_8 - 5t_8 - 2.5t_8^2$, $v_i(t) = -5t + 5t_8 + 5$.

4 Modeling and Verification with Deontic Rules in NuSMV

We propose to model the airport traffic control system as finite state automata in the NuSMV. In our case, there are actors: an aircraft, an intruder and a controller. Each of them have to be modeled as distinct, yet interconnected modules (automata). A module for the global clock must also be modeled, that would increment its value with every change of state of the whole model, being a time reference for other modules.

The controller module is characterized by its ability to receive and send information, i.e. to observe the intruder's presence, and to send to the aircraft and the intruder signals to stop them. The aircraft and intruder modules receive informations too, i.e. the signals from the controller to stop them. The aircraft receives also information about the intruder's presence. They are mobile and are characterized by their position (in relation to their start point and the intersection) and their motion kind (acceleration, deceleration, none, etc.). All the actors are characterized by time of their reaction to certain events, and time of their certain behavior, which will be explained later.

Because of limited space, only some extractions of the model's code will be presented, to show how to model time, communication, position and changes thereof.

How to model information communication. Receiving information about a current state of another module can be modeled, using the state's variable as a property. The main module's code extraction below is a declaration of an instance of the intruder module (*m_intruder*), where one of its properties is a state variable *stopIntruder* of the *controller* module, written in the dot-notification.

```
VAR intruder : m_intruder(clock, intruderIsPresent,
                          controller.stopIntruder);
```

Sending information about a current state of a module to another module can be therefore modeled by simply changing the variable's value be the sender module.

How to model position. It has to be modeled: how far an actor is in relation to its start position (the passed road), and if it is before, on or behind the intersection. Let us present it for the aircraft. Current position of the aircraft, in relation to the intersection, is used i.a. to determine if a collision with the intruder is possible, when the aircraft is not yet behind the intersection. It is modeled by a state variable *where* in the aircraft module, being an enumeration, as presented in the code extraction below. Its initial value *before* (being before the intersection) is defined by the INIT statement.

```
VAR where : {before,on,behind};   INIT where = before;
```

Current position of the aircraft, in relation to its start position, determines the variable *where*. It depends on the *clock* and dynamics of the aircraft. The position has to be given in m, where the start point is 0. It is modeled by a variable *passedRoad* in the aircraft module, where its maximal (2200 m) value is given, as presented below.

```
VAR passedRoad : 0..2200;          INIT road = 0;
```

How to model events. Events independent from actor modules should be modeled in the main module as boolean variables. In our case a variable *collision* may be set TRUE when the aircraft and the intruder are on the intersection (*where* = *on*; line 3).

```
1 TRANS next(collision) in case
2   --the collision will be TRUE or FALSE:
3   !collision & next(intruder.where)=on &
      next(aircraft.where)=on : {TRUE,FALSE};
4   --otherwise, the collision remains permanent:
5   TRUE : collision; esac;
```

How to model dynamics. Because of space limitation, only main aspects of the aircraft module will be presented. The code below defines aircraft module's variables.

```
VAR passedRoad : 0..2200;
  motion : {acceleration,deceleration,none,detached};
  where : {before,on,behind};
  t3 : -1..60; --time stamp of the start of deceleration
  halt : boolean; --aircraft's deontic decision to halt
  reactionToIntruder : boolean;
    --reaction to the intruder's appearance
  reactionToIntruderTime : 0..4; --real time of the above
  reactionToOrder : boolean;
    --reaction to the controller's order to stop
  reactionToOrderTime : 0..1; --real time of the above
INIT passedRoad = 0 & motion = acceleration &
  t3 = -1 &  -- -1 means, it is unused while acceleration
  halt = FALSE & reactionToIntruderTime = 0 &
  reactionToIntruder = FALSE & reactionToOrderTime = 0 &
  reactionToOrder = FALSE;
```

The following code presents how to determine value of the *passedRoad* variable, where *clock* gives the current moment, and *t3* is the start moment of deceleration.

```
INVAR motion = acceleration ->
  passedRoad = 988*clock*clock/1000;
INVAR motion = deceleration ->  passedRoad = 5725*clock*
  t3/1000 - 2863*t3*t3/1000 - 1875*clock*clock/1000;
TRANS (next(motion) != acceleration & next(motion) !=
  deceleration) -> next(passedRoad) = passedRoad;
```

The 1st INVAR statement is an invariant implication: for every state, if the aircraft is accelerating at t, the passed road is calculated as $s_a(t) = 988t^2/1000$. The 2nd INVAR statement is an invariant implication: for every state, if the aircraft is decelerating at t, the passed road is calculated as $s_a(t) = -2863t_3^2/1000 + 5725tt_3/1000 - 1875t^2/1000$. The TRANS statement declares no change of the *passed-Road* value if the aircraft has stopped or detached from the runway (final states).

If the aircraft is to halt (to break the take-off procedure to stop), the variable *halt* has to be set TRUE. Such a decision has deontic nature, and appropriate statements, setting the variable, will be presented later in a paragraph on modeling deontic rules.

The code below presents how to determine value of the *motion* variable, where: *clock* gives the current moment; t_3 is the start moment of deceleration; *reactionToOrder* and *reactionToIntruder* are set TRUE after a given time reaction

to the controller's order to stop, and after noticing the intruder by the aircraft, respectively; the moment of the take-off is t_2 = 45 s, and the moment of reaching V1 is t_1 = 38 s.

```
1 TRANS next(motion) in case
2   motion = acceleration & !(next(halt)) &
      next(clock) = 45 : detached;
3   motion = acceleration & clock < 38 & next(halt) &
      next(reactionToOrder) : deceleration;
4   motion = acceleration & clock < 38 & next(halt) &
      next(reactionToIntruder) : deceleration;
5   motion = deceleration & (5725*t3 - 3750*next(clock)
      <= 0) : none;
6   TRUE : motion; esac; --otherwise, no change
```

The take-off occurs in the 45th second of acceleration (t_2), provided there has not been taken the *halt* decision (line 2). The deceleration occurs if the *halt* decision has been taken, after a given time reaction (*reactionToOrder* or *reactionToIntruder*), provided the acceleration lasts less than 38 s ($t < t_1$; lines 3–4). The moment when the motion ceases is based on the equation for $v_a(t)$ for the decelerating aircraft (line 5).

The code below presents a TRANS statement, that copies the value of *clock* to t_3 at a moment of start of deceleration (line 2). t_3 remains unchanged thereafter (line 3).

```
1 TRANS next(t3) in case
2   motion = acceleration & next(motion) = deceleration :
      clock+1;
3   TRUE : t3; esac;
```

The code below presents a TRANS statement, that sets *reactionToIntruder* TRUE when *reactionToIntruderTime* allows, i.e. $\tau_{a1} \in [3, 5]$ s passes from the moment the intruder appears, to the moment the aircraft can react to it. *intruderIsPresent* is a module's parameter referencing a global variable, set TRUE when the intruder is present on the taxiway. Notice, that for $\tau_{a1} \in [1,4)$ s it is possible, yet not obligatory, to set the variable TRUE (line 2); then for τ_{a1} = 4 s it is obligatory now (line 3).

```
1 TRANS next(reactionToIntruder) in case
2   !reactionToIntruder & intruderIsPresent &
      next(reactionToIntruderTime) > 0 &
      next(reactionToIntruderTime) < 4 : {TRUE,FALSE};
3   !reactionToIntruder & intruderIsPresent &
      next(reactionToIntruderTime) = 4 : TRUE;
4   TRUE : reactionToIntruder; esac; --oth., no change
```

The code below shows a TRANS statement, incrementing *reactionToIntruderTime*.

```
TRANS next(reactionToIntruderTime) in case
  reactionToIntruderTime = 4 | !intruderIsPresent :
    reactionToIntruderTime; --no incrementation
  TRUE : reactionTimeToIntruder+1; esac; --incrementation
```

How to model time. Time in the system is measured in s by a global clock, modeled by a global integer variable in the main module, as presented below; where its maximal value has to be calculated in analysis of all possible scenarios, to be greater than the latest possible moment when the aircraft safely takes off or stops.

```
VAR clock : 0..60;
INIT clock = 0;
TRANS next(clock) in case --the clock in the next state:
  clock = 60 : clock; --not changed if the maximal value
  TRUE : clock+1; esac; --otherwise, incremented
```

The *clock* increments steadily by 1 s, which takes place with every change of state of the system model. Verification of the COMPUTE statements below proves it: the minimal and maximal paths from a state where $clock = i$, to a state where $clock = i + 1$ (for any i), contain exactly 1 state change. We assume the *clock* steadily increments, to be a reference frame for all other modules, to verify their time properties!

```
COMPUTE MIN[clock=0,clock=1]; --the minimal path = 1
COMPUTE MAX[clock=0,clock=1]; --the maximal path = 1
```

How to model deontic rules. The simplest way to model a system's deontic behavior is to add INVAR (invariant) statements of the kind *condition* → *conclusion*. The condition describes the situation in which a deontic rule determines to react to, and the conclusion describes the reaction. Thus, NuSMV creates a model, where these statements are invariantly satisfied, directly modeling obliged and forbidden behavior.

Let us include into the aircraft module, the following deontic rules of, supposedly, safe behavior of the aircraft, where *stopAircraft* and *intruderIsPresent* are references to other module's variables, meaning the controller orders the aircraft to stop, and the intruder is present, respectively. Other variables have been explained earlier.

```
INVAR stopAircraft & reactionToOrder -> halt
INVAR intruderIsPresent & reactionToIntruder ->
  halt | !halt
```

The 1st INVAR statement defines an obligation: If the controller gives the order to stop (*stopAircraft*), and the reaction to the order is possible because sufficient reaction time has passed (*reactionToOrder*), than the aircraft breaks to stop (*halt*). The 2nd INVAR statement defines a permission: If the intruder is present (*intruderIsPresent*), and if reaction to the intruder is possible because sufficient reaction time has passed (*reactionToIntruder*), than the aircraft breaks to stop (*halt*) or not (*!halt*). Notice, the implied *halt* reaction is possible here, yet the opposite reaction is equally possible. After adding a deontic rule to the model, the system's properties should be verified.

Verification of system properties. A property can be written as a specification statement, e.g. a CTL formula (CTLSPEC), or an invariant statement (INVARSPEC). If a property concerns the whole system, its statement must be placed in the main module; if it concerns a certain module only, its statement should be placed there. By running the system model script, a result will be returned: *true* if the property is satisfied, *false* otherwise, with a counterexample showing a shortest path of system's states leading to a state, that violates the property. Verification should prove the system to perform its functions properly. The code below are CTLSPEC statements, verifying the aircraft module (lines 1–3) and the whole system (line 4).

```
1 CTLSPEC EF(reactionToOrder & !halt) --should be false
2 CTLSPEC AG(reactionToOrder -> halt) --should be true
3 CTLSPEC AG(stopAircraft -> AF(halt)) --should be true
4 CTLSPEC AG(intruderIsPresent &
    aircraft.motion = acceleration & clock < 38 ->
    AF(aircraft.motion = deceleration)) --should be true
```

The 1st statement disproves, that: possibly there is or will be a moment, when the reaction to the controller's order to stop (*reactionOnOrder*) is allowed, and the aircraft does not have to stop (*!halt*) then. Thus, the 2nd statement proves, that: certainly whenever the reaction is allowed, the aircraft has to stop then. Moreover, the 3rd statement proves, that: certainly whenever there is the order (*stopAircraft*), the aircraft has to stop (*halt*) then or later. The last statement proves, that: certainly whenever the intruder is present (*intruderIsPresent*), and the aircraft accelerates,

and time of reaching V1 has not passed yet (*clock* < 38), the aircraft will certainly decelerate then or later.

Calculation of time. Time between given events can be measured by calculating the minimal or maximal path between two states representing the events. The proposed model of the clock makes the path's length minus 1 being the time that passes along the path. The code below are CALCULATE statements for the minimal and maximal paths from the state when the intruder appears (t_5 gets the *clock*'s value then), to the state when the aircraft starts decelerating (t_3 gets the *clock*'s value then).

```
COMPUTE MIN[t5=clock & clock<38,aircraft.t3=clock] --is 1
COMPUTE MAX[t5=clock & clock<38,aircraft.t3=clock] --is 7
```

Hazard analysis. It must be proved, the system behaves safely and no hazardous state can be reached. The code below is an INVARSPEC statement, checking if a hazard is never possible. The hazard is when the aircraft and the intruder are on the intersection. An INVARSPEC is a statement with a non-temporal formula being checked to be true in all states on all paths, being therefore an invariant property.

```
INVARSPEC !(aircraft.where = on & intruder.where = on)
```

If the formula is verified to be false, a counterexample is generated showing a possible shortest path of states leading to the hazard. Hence, it can be useful in a hazard cause–consequence analysis. For the presented case, the hazard was proved to be possible. Then, by manual changes of the model, it was proved, that if the longest controller's reaction time to the intruder's appearance is limited to 3 s, the hazard is impossible. Hence, the deontic INVAR statement below was added, stating that the controller is obliged to react to the intruder's appearance by setting *stopIntruder* TRUE, before 3 s of the *reactionTime* pass.

```
INVAR reactionTime >= 3 -> stopIntruder
```

Deonticity analysis. It is recommended to perform verification when the system model is built, and every time any deontic safety rule is added, to find a state violating the rule. Also deontic rules should be verified to hold, as presented by the code below.

```
INVARSPEC stopAircraft & reactionToOrder -> halt
INVARSPEC intruderIsPresent & reactionToIntruder ->
  halt | !halt --both should be true
```

The 1st statement proves, that during an appropriate time of reaction, the order to stop results in the decision to halt. The 2nd statement proves, that during an appropriate time of reaction, the intruder's appearance may (!) result in the decision to halt. If a rule is verified not to hold, an example violation path of states is returned.

5 Conclusions and Future Work

To our knowledge, the presented method is novel in using NuSMV in modeling and verification of a safety-critical system of moving objects, where quantitative time dependencies and deontic rules of behavior of the objects is considered. The script language is relatively simple to understand, temporal properties can automatically be verified, and a possible path of states leading to a user-defined hazard can be found.

I.a. the method allowed us, for the presented case study, to do a hazard analysis, and, in result thereof, the model has been changed to prevent the hazard.

An interesting problem of checking compliance have been stated in [1], where deontic rules of behavior in an airport and air traffic change, that can violate existing rules and lead to a hazard. Our future work will be on a method of checking compliance between new or changed rules and the system model, especially searching for allowed and forbidden changes.

References

1. Cholvy, L., Saurel. C.: Checking compliance of a system with regulations: towards a formalisation. In: Proceedings of ReMoD (2008)
2. Demasi, R. et al.: Synthesizing masking fault-tolerant systems from deontic specifications. In: Automated Technology for Verification and Analysis. Springer International Publishing (2013)
3. Shi, K.: Making existing reactive systems anticipatory: methodology and case studies. Doctoral Dissertation, Graduate School of Science and Engineering, Saitama Univ. (2013)
4. Schönefeld, J., Möller D.: Runway incursion prevention systems: a review of runway incursion avoidance and alerting system approaches. In: PiAS 51 (2012)
5. Gabillon, A., Laurent, G.: An availability model for avionic databuses. In: Proceedings of the Workshop on Issues in Security and Petri Nets. Eindhoven, Netherlands. vol. 23 (2003)
6. Han, CH. et al.: Spatio-temporal deontic relevant logic as the logical basis for air traffic control systems. In: Advances in Biomedical Engineering **8** (2012)
7. Cimatti, A. et al.: NuSMV: a new symbolic model checker. In: International Journal on Software Tools for Technology Transfer 2.4 (2000)
8. Emerson, E.A.: Temporal and modal logic. In: Handbook of Theoretical Computer Science, Volume B: Formal Models and Semantics (B) (1990)
9. Cavada, R. et al.: NuSMV 2.5 User Manual. FBK-irst, Trento (2011)
10. Cavada, R. et al.: NuSMV 2.5 Tutorial. FBK-irst, Trento (2011)
11. Aviation Investigation Report A13O0049. Transportation Safety Board of Canada (2014)

Semi-Markov Model of Damage Process

Franciszek Grabski

Abstract Semi-Markov model of an object damage process is discussed in the paper. Presented here models deal with unrepairable object. The multi-state reliability functions and corresponding expectations, second moments and standard deviations are evaluated for the presented cases of the object damage. A special case of the model is a multistate model with two kinds of failures. The presented model is illustrated by numerical example.

1 Introduction

We can find many papers which are devoted to the reliability of multistate monotone systems [1–9]. Some results of investigation of the multistate monotone system (MMS) with components modelled by the independent semi-Markov processes are presented in [4–6]. We consider an object with finite sets of the ordered reliability states $S = \{0, 1, \ldots, n\}$, where the state 0 is the worst while the state n is the best. We suppose that the probabilistic model of reliability evolution of the object damage is stochastic process $\{X(t) : t \geq 0\}$, taking values in the set of states $S = \{0, 1, \ldots, n\}$. We suppose that the object damage process evolution is described by the semi-Markov process which is defined by the general renewal kernel

$$Q(t) = \begin{bmatrix} Q_{00}(t) & 0 & 0 & \cdots & 0 \\ Q_{10}(t) & Q_{11}(t) & 0 & \cdots & 0 \\ Q_{20}(t) & Q_{21}(t) & Q_{22}(t) & \cdots & 0 \\ Q_{30}(t) & Q_{31}(t) & Q_{32}(t) & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & 0 \\ Q_{n0}(t) & Q_{n\,1}(t) & \cdots & Q_{n\,n-1}(t) & 0 \end{bmatrix}, \tag{1}$$

F. Grabski (✉)
Polish Naval University, Śmidowicza 69, 81-103 Gdynia, Poland
e-mail: F.Grabski@amw.gdynia.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_18

where some of functions $Q_{ij}(t)$ can be equal to 0. The trajectories of the process are the right continuous and piecewise constant functions. Recall that

$$Q_{ij}(t) = P(\tau_{n+1} - \tau_n \leq t, \quad X(\tau_{n+1}) = j | X(\tau_n) = i), \quad t \geq 0 \tag{2}$$

denotes one step transition probability from the state i to the state j during time not greater than t. A sequence $\{X(\tau_n) : \ n = 0, 1, \ldots\}$ is a homogenous Markov chain which is called an embedded Markov chain of SM process. One step transition probability of the embedded Markov chain is given by the rule

$$p_{ij} = P(X(\tau_{n+1}) = j \,|\, X(\tau_n) = i) = \lim_{t \to \infty} Q_{ij}(t), \ i, j \in S. \tag{3}$$

The function

$$G_i(t) = P(T_i \leq t) = P(\tau_{n+1} - \tau_n \leq t \,|\, X(\tau_n) = i) = \sum_{j \in S} Q_{ij}(t), \quad i, j \in S \tag{4}$$

is cumulative distribution function (CDF) of a random variable T_i denoting duration of the state i. The random variable T_i is called a *waiting time* in state *i,* when a successor state is unknown. Let

$$P_{iB}(t) = P(\forall u \in [0, t] \quad X(u) \in B | X(0) = i), \quad i \in B \subset S. \tag{5}$$

It denotes the probability that the process which starts from the state i during all the time from the interval $[0, t]$ occupies the states belonging to a subset B. Functions $P_{iB}(t)$, $i \in B \subset S$ satisfy the system of integral equations

$$P_{iB}(t) = 1 - G_i(t) + \sum_{j \in B} \int_0^t P_{jB}(t - x) \, dQ_{ij}(x), \quad i \in B. \tag{6}$$

We should mention that these equations are derived in Sect. 3.4 [4]. Using Laplace transformation we obtain system of linear equation

$$\widetilde{P}_{iB}(s) = \frac{1}{s} - \widetilde{G}_i(s) + \sum_{j \in B} \tilde{q}_{ij}(s) \widetilde{P}_{jB}(s), \quad i \in B. \tag{7}$$

where

$$\widetilde{P}_{iB}(s) = \int_0^\infty e^{-st} P_{iB}(t) dt. \tag{8}$$

If B is a subset of working states then the function

$$R_i(t) = P_{iB}(t), \quad i \in B \subset S \tag{9}$$

is the reliability function of a system with initial state $i \in B$ at $t = 0$

2 Time to Damage

Let

$$A_{[l]} = \{0, \ldots, l-1\} \quad \text{and} \quad A'_{[l]} = S - A_{[l]} = \{l, \ldots, n\}.$$

A random variable

$$T_{[l]} = \inf\{t : X(t) \in A_{[l]}\} \tag{10}$$

denotes the first passage time to the subset $A_{[l]}$ for the SM process $\{X(t) : t \geq 0\}$.

The function

$$\Phi_{i[l]}(t) = P(T_{[l]} \leq t | X(0) = i), \quad i \in A'_{[l]} \tag{11}$$

represents the cumulative distribution function (CDF) of the first passage time from the state $i \in A'_{[l]}$ to the subset $A_{[l]}$ for the $\{X(t) : t \geq 0\}$. If $X(0) = n$ then the random variable $T_{[l]}$ represents the l-level lifetime of the object. A corresponding reliability function is

$$R_{n\,[l]}(t) = P\big(T_{[l]} > t | X(0) = n\big) = 1 - \Phi_{n\,A_{[l]}}(t) \tag{12}$$

This function is called l-level reliability function. On the other hand

$$P(T_{[l]} > t | X(0) = n) = P(\forall u \in [0, t], \quad X(u) \in A'_{[l]} | X(0) = n) \tag{13}$$

Applying Eq. (7) we obtain system of linear equations for Laplace transforms of l-level reliability functions.

$$\widetilde{R}_{i\,[l]}(s) = \frac{1}{s} - \widetilde{G}_i(s) + \sum_{j \in A'_{[l]}} \tilde{q}_{ij}(s) \tilde{R}_{j\,[l]}(s), \quad i \in A'_{[l]}. \tag{14}$$

where

$$\widetilde{G}_i(s) = \int_0^\infty e^{-st} G_i(t)dt, \quad \widetilde{R}_{i\,[l]}(s) = \int_0^\infty e^{-st} R_{i\,[l]}(t)dt$$

are the Laplace transforms of the functions $G_i(t)$, $R_{i\,[l]}(t)$, $t \geq 0$. Passing to a matrix form we get

$$\left(I - q_{A'_{[l]}}(s)\right) R_{A'_{[l]}}(s) = G_{A'_{[l]}}(s), \tag{15}$$

where

$$I = \left[\delta_{ij} : \; i,j \in A'_{[l]}\right]$$

is the unit matrix and

$$q_{A'_{[l]}}(s) = \left[\tilde{q}_{ij}(s) : \quad i,j \in A'_{[l]}\right], \quad G_{A'_{[l]}}(s) = \frac{1}{s}\left[1 - \sum_{j \in S} \tilde{q}_{ij}(s) : \; i \in A'_{[l]}\right]^T,$$

$$R_{A'_{[l]}}(s) = \left[R_{i\,[l]} : \quad i \in A'_{[l]}\right]^T.$$

A vector function

$$R(s) = \left[\widetilde{R}_{n\,[0]}(s), \widetilde{R}_{n\,[1]}(s), \ldots, \widetilde{R}_{n\,[n]}(s)\right] \tag{16}$$

is the Laplace transform of the multi-states reliability function of the object.

3 Example 1

Let $S = \{0, 1, 2, 3\}$. Hence

$$\begin{array}{ll} A_{[1]} = \{0\}, & A'_{[1]} = \{1, 2, 3\}, \\ A_{[2]} = \{0, 1\}, & A'_{[2]} = \{2, 3\}, \\ A_{[3]} = \{0, 1, 2\}, & A'_{[3]} = \{3\}. \end{array}$$

Now the kernel (1) takes the form

$$Q(t) = \begin{bmatrix} Q_{00}(t) & 0 & 0 & 0 \\ Q_{10}(t) & Q_{11}(t) & 0 & 0 \\ Q_{20}(t) & Q_{21}(t) & Q_{22}(t) & 0 \\ Q_{30}(t) & Q_{31}(t) & Q_{32}(t) & 0 \end{bmatrix}. \tag{17}$$

For $l = 1$ the matrices from Eq. (15) have following form

$$I - q_{A'_{[l]}}(s) = \begin{bmatrix} 1 - \tilde{q}_{11}(s) & 0 & 0 \\ -\tilde{q}_{21}(s) & 1 - \tilde{q}_{22}(s) & 0 \\ -\tilde{q}_{31}(s) & -\tilde{q}_{32}(s) & 1 - \tilde{q}_{33}(s) \end{bmatrix},$$

$$G_{A'_{[l]}}(s) = \frac{1}{s} \begin{bmatrix} 1 - \tilde{q}_{10}(s) - \tilde{q}_{11}(s) \\ 1 - \tilde{q}_{20}(s) - \tilde{q}_{21}(s) - \tilde{q}_{22}(s) \\ 1 - \tilde{q}_{30} - \tilde{q}_{31}(s) - \tilde{q}_{32}(s) - \tilde{q}_{33}(s) \end{bmatrix}.$$

We are interested in the element $\widetilde{R}_{3[1]}(s)$ of the solution.

$$R_{A'[l]}(s) = \begin{bmatrix} \widetilde{R}_{1[1]}(s) \\ \widetilde{R}_{2[1]}(s) \\ \widetilde{R}_{3[1]}(s) \end{bmatrix}.$$

For $l = 2$ the matrices from Eq. (15) take the form

$$I - q_{A'_{[2]}}(s) = \begin{bmatrix} 1 - \tilde{q}_{22}(s) & 0 \\ -\tilde{q}_{32}(s) & 1 - \tilde{q}_{33}(s) \end{bmatrix},$$

$$G_{A'_{[l]}}(s) = \frac{1}{s} \begin{bmatrix} 1 - \tilde{q}_{20}(s) - \tilde{q}_{21}(s) - \tilde{q}_{22}(s) \\ 1 - \tilde{q}_{30} - \tilde{q}_{31}(s) - \tilde{q}_{32}(s) - \tilde{q}_{33}(s) \end{bmatrix}.$$

For $l = 3$ the matrices from Eq. (15) are

$$I - q_{A'_{[3]}}(s) = [1 - \tilde{q}_{33}(s)],$$

$$G_{A'_{[3]}}(s) = \frac{1}{s}[1 - \tilde{q}_{30} - \tilde{q}_{31}(s) - \tilde{q}_{32}(s) - \tilde{q}_{33}(s)]$$

Mostly, the elements $Q_{ii}(t), \quad i = 1, 2, \ldots, n$ are equal to 0. Let us suppose that

$$Q(t) = \begin{bmatrix} Q_{00}(t) & 0 & 0 & 0 \\ Q_{10}(t) & 0 & 0 & 0 \\ Q_{20}(t) & Q_{21}(t) & 0 & 0 \\ Q_{30}(t) & Q_{31}(t) & Q_{32}(t) & 0 \end{bmatrix}. \tag{18}$$

Hence we obtain

$$R_{3[1]}(\mathrm{s}) = \frac{1 - \tilde{q}_{30}(s) - \tilde{q}_{10}(s)\tilde{q}_{31}(s) - \tilde{q}_{20}(s)\tilde{q}_{32}(s) - \tilde{q}_{10}(s)\tilde{q}_{21}(s)\tilde{q}_{32}(s)}{s}, \quad (19)$$

$$\widetilde{R}_{3[2]}(s) = \frac{1 - \tilde{q}_{30}(s) - \tilde{q}_{31}(s) - \tilde{q}_{20}(s)\tilde{q}_{32}(s) - \tilde{q}_{21}(s)\tilde{q}_{32}(s)}{s}, \quad (20)$$

$$\widetilde{R}_{3[3]}(s) = \frac{1 - \tilde{q}_{30} - \tilde{q}_{31}(s) - \tilde{q}_{32}(s)}{s}. \quad (21)$$

The Laplace transform of the multi-state reliability function of that object is

$$R(s) = \left[\widetilde{R}_{3\,[0]}(s), \widetilde{R}_{3\,[1]}(s), \widetilde{R}_{3\,[2]}(s), \widetilde{R}_{3\,[3]}(s)\right].$$

4 Model of Two Kinds of Failures

We assume that the failures are caused of wear or some random events. With positive probabilities there are possible only the state changes from k to $k-1$ or from k to 0. Time of state change from a state k to $k-1$, $\quad k = 1, \ldots, n$ because of wear is assumed to be a nonnegative random variable η_k with a PDF $f_k(x)$, $x \geq 0$. Time to a total failure (state 0) of the object in the state k is a nonnegative random variable ζ_k exponentially distributed with parameter λ_k. Under those assumptions the stochastic process $\{X(t) : t \geq 0\}$ describing the reliability state of the component is the semi-Markov process with a state space $S = \{0, 1, \ldots, n\}$ and a kernel

$$Q(t) = \begin{bmatrix} Q_{00}(t) & 0 & 0 & \cdots & 0 \\ Q_{10}(t) & 0 & 0 & \cdots & 0 \\ Q_{20}(t) & Q_{21}(t) & 0 & \cdots & 0 \\ Q_{30}(t) & 0 & Q_{32}(t) & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & 0 \\ Q_{n0}(t) & 0 & \cdots & Q_{n\,n-1}(t) & 0 \end{bmatrix}, \quad (22)$$

where

$$Q_{k\,k-1}(t) = P(\eta_k \leq t, \zeta_k > \eta_k) = \int_0^t e^{-\lambda_k x} f_k(x) dx, \quad (23)$$

$$Q_{k\,0}(t) = P(\zeta_k \leq t, \eta_k > \zeta_k) = \int_0^t \lambda_k e^{-\lambda_k x} [1 - F_k(x)]\, dx \quad (24)$$

for $k = 1, \ldots, n$.

5 Example 2

To explain such model we take $n = 3$ and we suppose that random variables η_k, $k = 1, 2, 3$ have gamma distributions with parameters $\alpha_k = 1, 2, \ldots$ and $\beta_k > 0$ with PDF

$$f_k(x) = \frac{\beta_k^{\alpha_k} x^{\alpha_k - 1} e^{-\beta_k x}}{(\alpha_k - 1)!} \tag{25}$$

In this case the matrix (16) has the form:

$$Q(t) = \begin{bmatrix} Q_{00}(t) & 0 & 0 & 0 \\ Q_{10}(t) & 0 & 0 & 0 \\ Q_{20}(t) & Q_{21}(t) & 0 & 0 \\ Q_{30}(t) & 0 & Q_{32}(t) & 0 \end{bmatrix} \tag{26}$$

Let us notice that this matrix is equal to the matrix (18) from Example 1 with $Q_{31}(t) = 0$. Therefore we can apply equalities (19)–(21) to calculate components of multi-state reliability function. Finally we obtain the Laplace transforms:

$$\widetilde{R}_{3[1]}(s) = \frac{1 - \tilde{q}_{30}(s) - \tilde{q}_{20}(s)\tilde{q}_{32}(s) - \tilde{q}_{10}(s)\tilde{q}_{21}(s)\tilde{q}_{32}(s)}{s}, \tag{27}$$

$$\widetilde{R}_{3[2]}(s) = \frac{1 - \tilde{q}_{30}(s) - \tilde{q}_{20}(s)\tilde{q}_{32}(s) - \tilde{q}_{21}(s)\tilde{q}_{32}(s)}{s}, \tag{28}$$

and

$$\widetilde{R}_{3[3]}(s) = \frac{1 - \tilde{q}_{30} - \tilde{q}_{32}(s)}{s}, \tag{29}$$

where

$$\begin{aligned} \tilde{q}_{10}(s) &= \frac{\beta_1^2}{(s+\beta_1+\lambda_1)^2} + \frac{\lambda_1(s+2\beta_1+\lambda_1)}{(s+\beta_1+\lambda_1)^2}, \\ \tilde{q}_{21}(s) &= \frac{\beta_2^2}{(s+\beta_2+\lambda_2)^2}, \quad \tilde{q}_{20}(s) = \frac{\lambda_2(s+2\beta_2+\lambda_2)}{(s+\beta_2+\lambda_2)^2}, \\ \tilde{q}_{32}(s) &= \frac{\beta_3^2}{(s+\beta_3+\lambda_3)^2}, \quad \tilde{q}_{30}(s) = \frac{\lambda_3(s+2\beta_3+\lambda_3)}{(s+\beta_3+\lambda_3)^2}. \end{aligned} \tag{30}$$

For numerical example we fix

$$\begin{aligned} \alpha_1 &= 2, \quad \beta_1 = 0.04, \quad \lambda_1 = 0.004, \\ \alpha_2 &= 2, \quad \beta_2 = 0.03, \quad \lambda_2 = 0.002, \\ \alpha_3 &= 2, \quad \beta_3 = 0.02, \quad \lambda_3 = 0.001. \end{aligned}$$

Substituting functions (30) with above parameters to Eqs. (27)–(29) we compute transforms

$$\widetilde{R}_{3[1]}(s),\ \widetilde{R}_{3[2]}(s),\ \widetilde{R}_{3[3]}(s).$$

Using MATHEMATICA program we compute inverse Laplace transforms. Finally we obtain l-level reliability functions

$$\begin{aligned}
R_{3[1]}(t) &= e^{-0.044t}(1+0.04t),\\
R_{3[2]}(t) &= 47.9166\mathrm{e}^{-0.044t} - 46.9166\mathrm{e}^{-0.032t} + 0.2499\mathrm{e}^{-0.044t}t + 0.3550\mathrm{e}^{-0.032t}t\\
R_{3[3]}(t) &= 52.6697\mathrm{e}^{-0.044t} + 58.2769\mathrm{e}^{-0.032t} - 109.9466\mathrm{e}^{-0.021t}\\
&\quad + 0.189\mathrm{e}^{-0.044t}t + 1.1735\mathrm{e}^{-0.032t}t + 0.5098\mathrm{e}^{-0.021t}t
\end{aligned}$$

These reliability functions are shown in Fig. 1.

The corresponding expectations, second moments and standard deviations of l level system lifetime we calculate using the following formulae

$$\begin{aligned}
m_1[l] &= [E[T_{[l]}|X(0)=3] = \lim_{s\to 0}\widetilde{R}_{3[l]}(s), \quad l=1,2,3,\\
m_2[l] &= E[T^2_{[l]}|X(0)=3] = -2\lim_{s\to 0}[\widetilde{R}'_{3[l]}(s)],\\
\sigma[l] &= \sqrt{m_2[l]-[m_1[l]]^2}.
\end{aligned}$$

For given parameters we get

$$\begin{array}{lll}
m_1[1]=43.38, & m_1[2]=98.68 & m_1[3]=182.47,\\
m_2[1]=2911.34, & m_2[2]=12940.82, & m_2[3]=41960.11,\\
\sigma[1]=32.07, & \sigma[2]=56.59 & \sigma[3]=93.07.
\end{array}$$

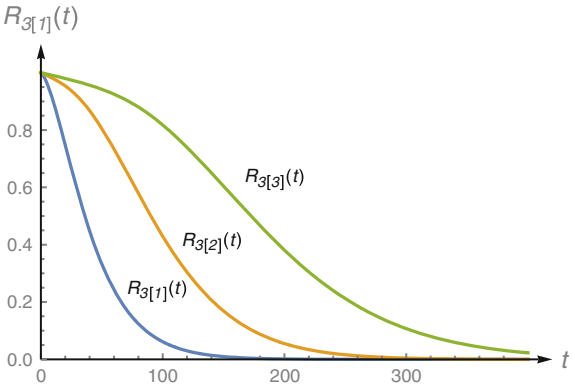


Fig. 1 The l level reliability functions

6 Simple Model of Damage Process

We assume that there are possible the state changes only from k to $k-1$, $k = 1, 2, \ldots, n$ with the positive probabilities. Now, the stochastic process $\{X(t) : t \geq 0\}$ describing reliability state of the object, is the semi-Markov process with a state space $S = \{0, 1, \ldots, n\}$ and a kernel

$$Q(t) = \begin{bmatrix} Q_{00}(t) & 0 & 0 & \cdots & 0 \\ Q_{10}(t) & 0 & 0 & \cdots & 0 \\ 0 & Q_{21}(t) & 0 & \cdots & 0 \\ 0 & 0 & Q_{32}(t) & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & 0 & \cdots & Q_{nn-1}(t) & 0 \end{bmatrix} \tag{31}$$

Age from Eq. (15) we obtain the Laplace transforms of the multi-state reliability function components:

$$\widetilde{R}_{n[k]}(s) = \frac{1 - \tilde{q}_{kk-1}(s)\tilde{q}_{k+1k}(s)\ldots\tilde{q}_{nn-1}(s)}{s}, \quad k = 1, 2, \ldots, n \tag{32}$$

Notice that for the kernel (31) a function $Q_{k\,k-1}(t)$, for $k = 1, 2, \ldots,$ n is equal to CDF of a waiting time in state k

$$Q_{k\,k-1}(t) = G_k(t) = P(T_k \leq t), \quad k = 1, 2, \ldots, n \tag{33}$$

Theorem *If*

$$Q_{k\,k-1}(t) = \begin{cases} 0 & \text{for} \quad t < 0 \\ 1 - \left(1 - \frac{\lambda_k}{\lambda_{k+1}}\right) e^{-\lambda_k t} & \text{for} \quad t \geq 0, \end{cases} \quad k = 1, \ldots, n-1 \tag{34}$$

and

$$Q_{nn-1}(t) = \begin{cases} 0 & \text{for} \quad t < 0 \\ 1 - e^{-\lambda_n t} & \text{for} \quad t \geq 0 \end{cases} \tag{35}$$

then the multi-state reliability function is

$$R(t) = \left[1, e^{-\lambda_1 t}, e^{-\lambda_2 t}, \ldots, e^{-\lambda_n t}\right],$$

where

$$0 < \lambda_1 < \lambda_2 < \ldots < \lambda_n.$$

Proof Notice that

$$\widetilde{Q}_{kk-1}(s) = L[Q_{kk-1}(t)] = \frac{\lambda_k(s+\lambda_{k+1})}{s\,(s+\lambda_k)\lambda_{k+1}} \quad \text{for} \quad \widetilde{Q}_{k\,k-1}(s) = \frac{\tilde{q}_{k\,k-1}(s)}{s},$$
$$k = 1, 2, \ldots, n.$$

Therefore, according to (32) we obtain

$$\begin{aligned}\widetilde{R}_{n[\mathrm{k}]}(s) &= \frac{1 - \tilde{q}_{\mathrm{kk-1}}(s)\tilde{q}_{\mathrm{k+1k}}(s)\ldots\tilde{q}_{n\,n-1}(s)}{s} \\ &= \frac{1}{s} - \frac{\lambda_k(s+\lambda_{k+1})}{s(s+\lambda_k)\lambda_{k+1}}\frac{\lambda_{k+1}(s+\lambda_{k+2})}{(s+\lambda_{k+1})\lambda_{k+2}}\cdots\frac{\lambda_{n-1}(s+\lambda_n)}{(s+\lambda_{n-1})\lambda_n}\frac{\lambda_n}{(s+\lambda_n)} = \frac{1}{s} - \frac{\lambda_k}{s(s+\lambda_k)} \\ &= \frac{(s+\lambda_k)-\lambda_k}{s(s+\lambda_k)} = \frac{1}{s+\lambda_k}, \quad k = 1, 2, \ldots, n \qquad \square\end{aligned}$$

It means that

$$R_{n[\mathrm{k}]}(\mathrm{t}) = L^{-1}\left[\frac{1}{s+\lambda_k}\right] = e^{-\lambda_k t}, t \geq 0, \quad k = 1, 2, \ldots, n.$$

Thus, the theorem has been proved.

We should note that from assumptions of that theorem it follows that the probability distributions the random variables $T_k, \quad k = 1, 2, \ldots, n-1$ are a mixture of a discrete and absolutely continuous distributions

$$G_k(t) = p\,G_k^{(d)}(t) + q\,G_k^{(c)}(t), \quad k = 1, \ldots n-1,$$

where

$$p = \frac{\lambda_k}{\lambda_{k+1}}, \quad q = 1 - \frac{\lambda_k}{\lambda_{k+1}},$$

$$G_k^{(d)}(t) = \begin{cases} 0 & \text{for} \quad t<0 \\ 1 & \text{for} \quad t \geq 0 \end{cases}, \quad G_n^{(c)}(t) = \begin{cases} 0 & \text{for} \quad t<0 \\ 1 - e^{-\lambda_k t} & \text{for} \quad t \geq 0 \end{cases}.$$

From the above it follows that

$$P(T_k = 0) = \frac{\lambda_k}{\lambda_{k+1}}, \quad k = 1, \ldots, n-1.$$

References

1. Aven, T.: Relibility evaluation of multistate systems with multistate components. IEEE Trans. Reliab. **34**(2), 463–472 (1985)
2. Block, H.W., Savits, T.H.: A decomposition for multistate monotone systems. J. Appl. Probab. **19**(2), 391–402 (1982)
3. Grabski, F.: Semi-Markov Models of Reliability and Operation. IBS PAN, Warsaw (2002). (in Polish)
4. Grabski, F.: Semi-Markov Processes: Applications in Systems Reliability and Maintenance. Elsevier, Amsterdam (2015)
5. Grabski, F., Kołowrocki. K.: Asymptotic reliability of multistate system with semi-markov states of components. In: Proceedings. of the European Conference on Safety and Reliability - ESREL'99, pp. 317–322 (1999)
6. Kołowrocki, K.: Reliability of Large Systems. Elsevier (2004)
7. Krolyuk, V.S., Turbin, A.F. (1976). Semi-Markov processes and their applications. Kiev Naukova Dumka (in Russian)
8. Korczak, E.: Reliability analisis of non-repaired multistste systems. In: Proceedings of the Advances in Safety and Reliability: ESREL'97, pp. 2213–2220 (1997)
9. Lisnianski, A., Frankel, I.: Non-homogeneous markov reward model for an ageing multi-state system under minimal repair. Int. J. Perform. Eng. **4**(5), 303–312 (2009)

The Problem of Tyre Footprint Width Estimation by Fibre Optic WIM Sensors in Condition of Geometric Complexity

Alexander Grakovski and Alexey Pilipovecs

Abstract The measuring vehicle's weigh-in-motion (WIM) is one of the actual research fields of transport telematics. It is important for vehicles classification and identification, as well as overload enforcement, road maintenance planning, cargo fleet managing in intelligent transport systems, control of the legal use of the transport infrastructure, road surface protection from the early destruction and for the safety on the roads. One of sufficient subtasks of WIM is the tyre footprint width estimation for correct contact area calculation. The problem of tyre width estimation by fibre optic sensors (FOS) measuring system is complicated by nonlinearity, and complex interaction dynamics of vehicle's tyre and the group of sensors. The graphical model of interaction with sloped (diagonal) sensor, its properties, and the tyre width estimation algorithm with correction of geometric interaction are discussed here.

Keywords Weigh-in-motion (WIM) · Fibre-optic sensor · Tyre footprint width estimation

1 Introduction

Weigh-in-motion (WIM) technologies offer a fast and accurate measurement of the actual weights of the trucks when entering and leaving the road infrastructure facilities. Unlike the static weighbridges, WIM systems are capable of measuring vehicles travelling at a reduced or normal traffic speeds and do not require the vehicle to come to a stop. This makes the weighing process more efficient, and in the case of the commercial vehicle allows the trucks under the weight limit to bypass the enforcement.

A. Grakovski (✉) · A. Pilipovecs
Transport and Telecommunication Institute, Lomonosova 1, Riga 1019, Latvia
e-mail: avg@tsi.lv

A. Pilipovecs
e-mail: pilipovecs@gmail.com

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_19

The fibre-optic sensors (FOS), whose working principle is based on the change of the optical signal parameters due to the optic fibre deformation under the weight of the crossing road vehicle [1–3], have gained popularity in the last decade.

A lot of recent investigations are devoted to the peculiarities of the construction and applications of the sensors, using different physical properties. The data presented in this publication have been received using PUR experimental sensors [2] based on the change of the transparency (the intensity of the light signal) during the deformation.

Fibre optic load-measuring cables are placed in the gap across the road, filled with resilient rubber (Fig. 1b). The gap width is 30 mm. Since the sensor width is smaller than the tyre footprint on the surface, the sensor takes only part of the axle weight. The following formula is used to calculate the total weight of the axis using the basic method [3]:

$$W_{ha} = \int_{t_{end}}^{t_{front}} (A_t(t) \cdot P_t(t))dt \tag{1}$$

where W_{ha}—weight on half-axle, $A_t(t)$—dynamic area of the tyre footprint, $P_t(t) \sim V(t)$—air pressure inside the tyre and, according to Newton's 3rd law, it is proportional to the axle weight.

The deflection of a light conductor and reflective coating occurs, that is why the conditions of light reflection inside are changed, and some of it escapes. The greater the load the less light comes from the second end of the light conductor. Therefore the sensor has the unusual characteristic for those, familiar with the strain gauges: the greater the load the lower the output is [4].

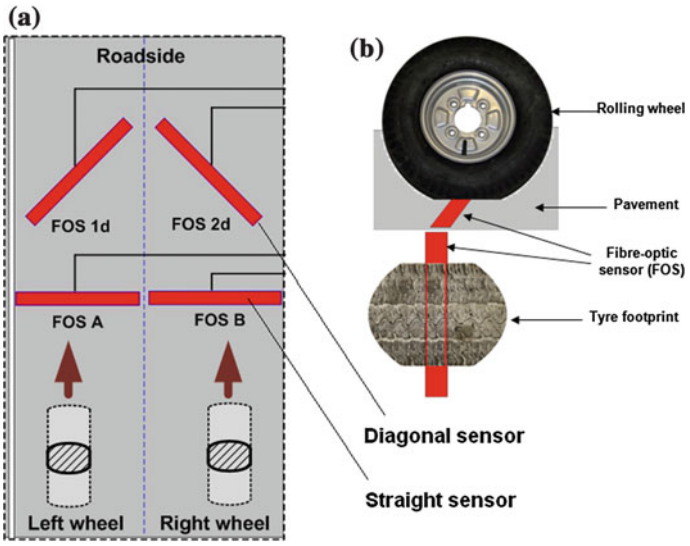


Fig. 1 **a** The system of weigh-in-motion (WIM) measurement, and **b** fibre optic sensors position against the wheel and tyre footprint

The area of the tyre footprint $A_r(t)$ from (1) is calculated roughly by the length of the output voltage impulse, which, in its turn, depends on the vehicle speed. The area method [3] uses the assumption that the area under the recorded impulse curve line, in other words—the integral, characterizes the load on the axle. To calculate the integral, the curve line is approximated by the trapezoid. In this case the smaller the integral then the greater the load. The algorithm of the estimation of each wheel and axle weigh, based on tyre footprint shape reconstruction [4, 5], required the information about tyres footprint width. In the experiments [4] it was known as the parameters of experimental vehicle (tyre R22.5, 1st and 2nd axles—250 mm, 3rd–5th axles—300 mm), but in common case, of course, it is unknown and requires the method of estimation. In our research we can use it as the reference values for comparison with the results of evaluation. As any another parameters of WIM measurement (speed, temperature, weigh) the width of tyre footprint sufficiently can exert influence on the results of WIM.

2 Vehicle Tyre Footprint Contact Width Evaluation

Usually for this purpose the diagonal fibre optic sensor in combination with perpendicular can be used for tyre contact patch (or footprint) evaluation [6]. Using the system (see Fig. 1a) of FOS A (FOS B) and sloped (diagonal) FOS 1d (FOS 2d) signals, which are shown in Fig. 2, it seems to be possible to calculate the width of each tyre. The truck speed v by calculating the average of the values of impulse peak time value shift of two straight sensors (when the distance between sensors is known) on each wheel can be fulfilled with enough accuracy [7].

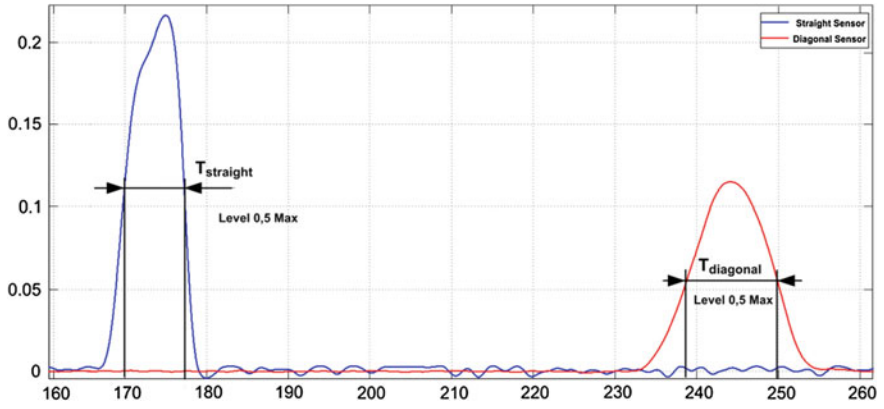


Fig. 2 Normalized and filtered signals of straight (FOS A) and diagonal (FOS 1d) sensors

When the speed of vehicle v is known, the evaluation of tyre footprint contact width W can be easy to estimate by the expression:

$$W = v \cdot \left(T_{diagonal} - T_{straight}\right), \tag{2}$$

where $T_{diagonal}$ is the duration of the pulse of diagonal (sloped) straight sensor on the level of half of the pulse amplitude, and $T_{straight}$ is the same for straight sensor (see Fig. 2). But the attempt to use this method in direct calculations allows us to significant errors in width on 30–50 % or more. The reason of low accuracy of calculations is in the nonlinear nature of fibre optic sensor [4] as well as the specific behaviour of contact area between the tyre and the surface of the diagonal sensor. Also we need to take into account the dual wheels of so called motor axle (the distance between two neighbourhoods dual wheels approximately is 100–150 mm and it cannot be measured exactly).

In order to avoid these effects, it is necessary at first to normalize the signals, filter out the noise and make linearization of the signals according to the pre-calculated axial velocity and temperature of the FOS in the same manner, as it is for pressure volume on sensor calculation [5]. The results of linearization of the pulses of voltage (see Fig. 2, signals from straight and diagonal sensors) are presented on Fig. 3.

It is clearly seen that linearized signal of pressure in range of magnitude (0–0.3) sufficiently distorted by amplified noise so, that the measurement of full duration of the pulses for time difference (2) is impossible.

In this situation the appropriate solution may be an evaluation of time difference between full durations of both pulses $T_{diagonal}$ and $T_{straight}$ respectively, where each of them defines by the crossing points of the tangent line, drawn through the points of front and back of the pulse on the level of 0.5 of maximum value, and the horizontal (time) axis (see Fig. 3a, b). The results of tyre footprint width estimation according to expression (2) where $T_{diagonal}$ and $T_{straight}$ are defined by mentioned

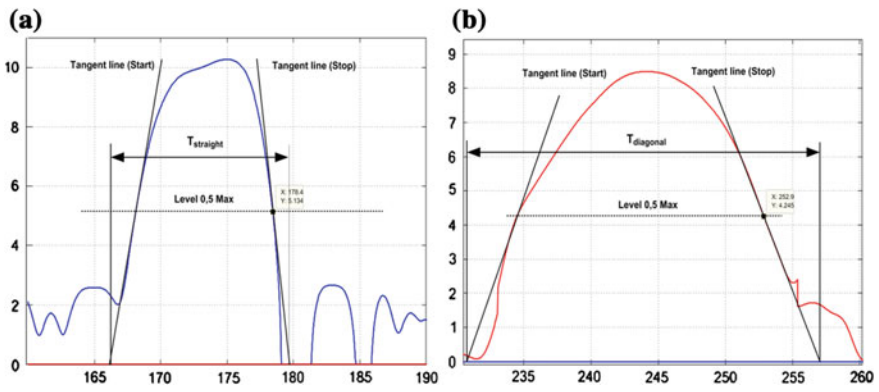


Fig. 3 Linearized signals of **a** straight (FOS A) and **b** diagonal (FOS 1d) sensors

Table 1 Evaluated tyre footprint width of the signal for the speed of 90 km

Reference tyre width (mm)			250	650	300	300	300
Speed	No	Axle	1st	2nd	3rd	4th	5th
90 km/h	1	Left	212.47	660.02	310.28	302.58	288.88
		Right	182.49	659.86	308.76	295.5	297.12
		Average	197.48	659.94	309.52	299.04	293.00
Error (%)			−21.01	1.53	3.17	−0.32	−2.33

Table 2 Tyre footprint length/width ratio for experimental vehicle wheels

Axle	1st	2nd	3rd	4th	5th
L/W ratio	0.721	0.275	0.533	0.533	0.533

above method, are presented in Table 1, where maximum error occurs the value of 21 % at the 1st axle wheels.

The explanation of the maximum error on 1st axle wheels can be provided by comparison of proportion of footprint (tyre contact patch)—ratio between length and width of it (see Table 2).

So, it is clearly seen that the geometry of the 1st axle differs from another axles. Let to obtain the nature of dynamics we need to consider the model of the process where the L/W ratio will be sufficient parameter.

3 Virtual Model of Interaction of Tyre and Diagonal Sensor

From the laboratory experiments with the fibre optic sensor of PUR type [2, 4, 5] it is known that the signal magnitude of FOS is sufficiently depended on active length of the sensor, situated under the pressure. Let to understand the law of active length changes we design the graphic model of moving wheel dynamics across the diagonal FOS (see Fig. 4).

The movement of the tyre footprint across the diagonal sensor leads to the gradual change of active length of the sensor (where the contact between tyre and sensor takes place). Measured active length is presented on the diagram as blue curve in dependence on the distance of wheel displacement (see Fig. 4).

For the understanding of footprint shape we select the group of the shapes of moving objects with common width and different length/width ration (see Fig. 5). The results of the modelling are reflected on the diagrams in Fig. 6.

Full distance of sensor activity in common case is equivalent the sum of so called effective length L_{ef} and width L_{ef} of the footprint form (see Fig. 4):

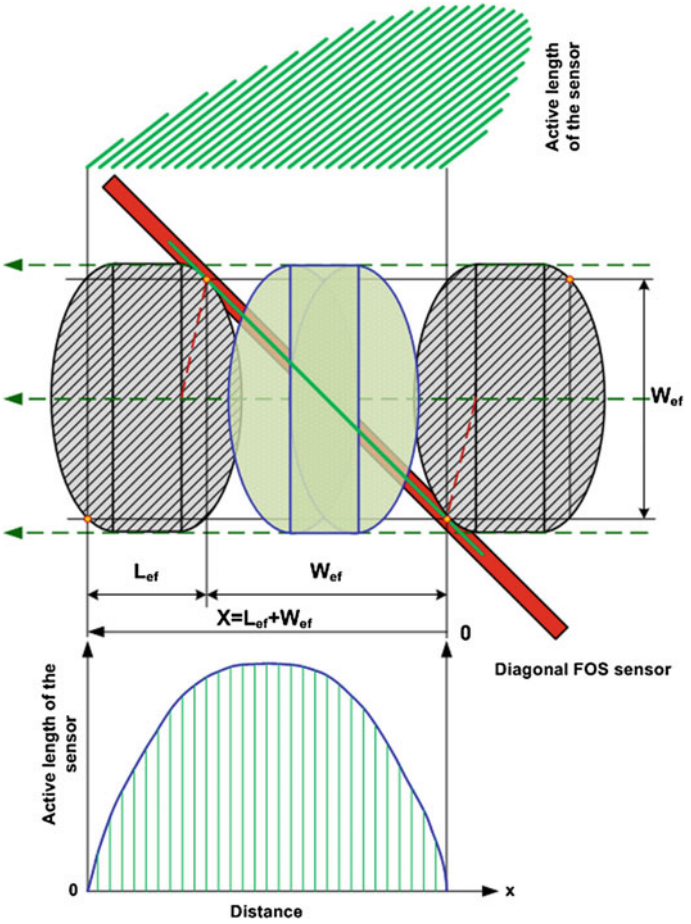


Fig. 4 Graphic model of moving wheel dynamics across the diagonal FOS

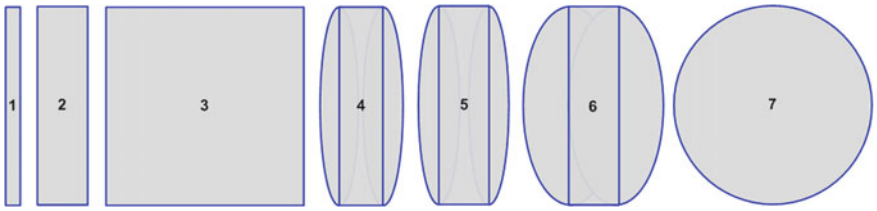


Fig. 5 Different shapes of footprint movement though the diagonal sensor (*1* rectangle narrow, *2* rectangle wide, *3* square rectangle, *4* tyre footprint narrow, *5* tyre footprint narrow with increased polygon shape, *6* tyre footprint wide, *7* spherical patch)

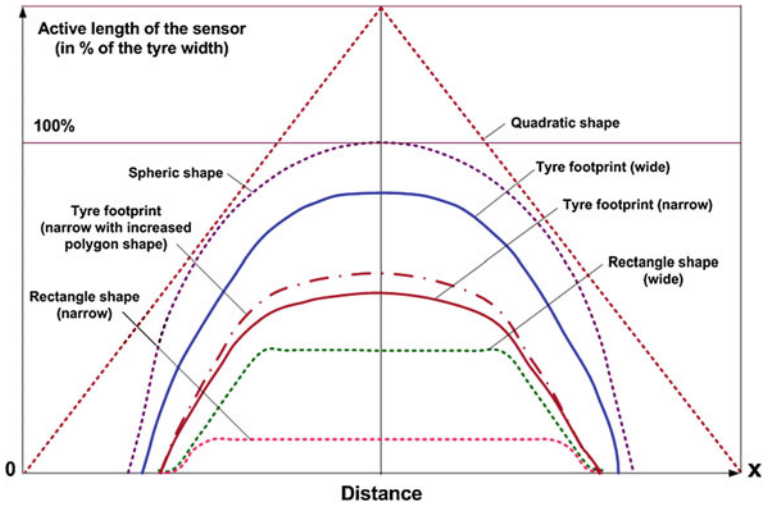


Fig. 6 Changes in active sensor's length in dependence on the distance according to the shape of the object (see Fig. 5)

$$X = L_{ef} + W_{ef}, \tag{3}$$

what fully responds to the algorithm (2) of tyre width calculation.

From the diagram on Fig. 6 is seen that the distortion of pressure value for 1st axle (L/W ratio 0.721) is more sufficient. But the increasing of active length of the sensor decreases the pressure value [5] (see Fig. 7a).

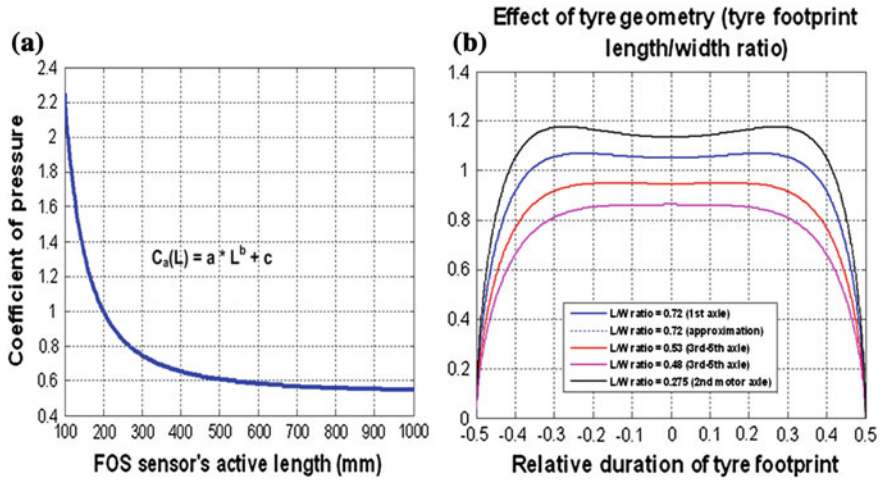


Fig. 7 Characteristics of FOS: **a** pressure correction coefficient as the function of active length of the sensor, **b** combined correction curves in dependence on the L/W ratio

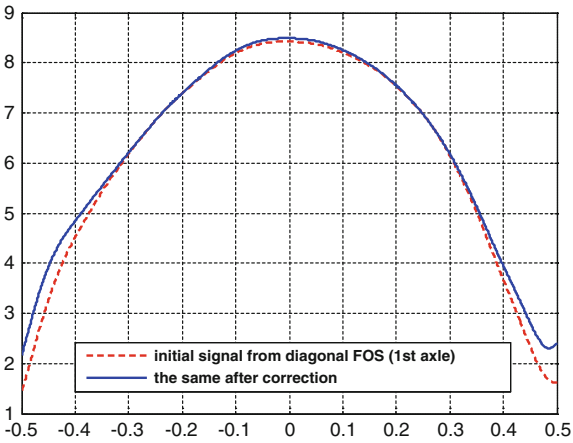


Fig. 8 Transformation of the form of the signal after correction

Common influence of both effects produces the combined correction characteristics for each wheel of the vehicle (see Fig. 7). Application of the correction curve to the pressure pulse form of 1st axle on diagonal sensor (Fig. 3b) leads the changes in the form of the signal (Fig. 8) and changes the slope of tangent lines in the algorithm of tyre width estimation. In this example the result leads to tyre width estimation increasing. For the same record (Table 1) the evaluation of the 1st axle tyre width changes from 197.48 to 237.90 mm which can consider as appropriate result (accuracy 4.85 %).

Table 3 Evaluated tyre footprint width of the signals for the speed of 30–90 km/h

Reference tyre width (mm)			250	650	300	300	300
Speed	No	Axle	1st	2nd	3rd	4th	5th
90 km/h	1	Left	261.73	660.02	310.28	302.58	288.88
		Right	214.07	659.86	308.76	295.5	297.12
		Average	237.90	659.94	309.52	299.04	293.00
Error (%)			−4.84	1.53	3.17	−0.32	−2.33
70 km/h	1	Left	268.26	669.85	286.78	272.25	296.64
		Right	266.84	644.94	286.34	281.26	294.06
		Average	267.55	657.395	286.56	276.755	295.35
Error (%)			7.02	1.14	−4.48	−7.75	−1.55
50 km/h	1	Left	230.1	620.26	298.67	302.81	310.91
		Right	258.49	655.11	288.9	314.06	316.96
		Average	244.295	637.685	293.785	308.435	313.935
Error (%)			−2.28	−1.89	−2.07	2.81	4.65
30 km/h	1	Left	253.03	675.31	319.28	324.47	348.56
		Right	189.19	646.56	294.27	288.69	298.84
		Average	221.11	660.935	306.775	306.58	323.7
Error (%)			−11.56	1.68	2.26	2.19	7.90

In the same manner it was processed the representative records with different speed of the reference vehicle (5-axle truck [4]) in range of 30–90 km/h.

The results of tyre width evaluation, presented in Table 3, demonstrate relatively good quality of measurement enough for tyre classification, and weight calculation.

4 Conclusions

Consider the WIM problem through the FOS sensor's accuracy point of view we can conclude that each component of input data, including the tyre footprint width, can significantly affects on the accuracy of measurement results, depending not only on sensor's geometric position and properties of components, but also on weather conditions.

The experimental data processing results show that the range of the vehicles velocity from 30 to 90 km/h is fully appropriate for WIM based on fibre-optic sensors method of tyre width estimation. Further researches direction can be connected with the analytic description of detected effect of diagonal sensor let to allow it compensation method on all axles of measured vehicles with different length/width ratio of tyre footprint (patch) and to increase the accuracy of evaluation respectively.

Acknowledgments This work was supported by Latvian state research program project "The next generation of information and communication technologies (Next IT)" (2014–2017).

References

1. Weight-in-motion of Axles and Vehicles for Europe (WAVE) General report, European Commission DG VII—Transport, 120 pp. (2011)
2. SENSORLINE GmbH: (© Sensor Line). SPT Short Feeder Spliceless Fiber Optic Traffic Sensor: product description (2010). http://sensorline.de/home/pages/downloads.php
3. Teral, S.: Fiber optic weigh-in-motion: looking back and ahead. Opt. Eng. **3326**, 129–137 (1998)
4. Grakovski, A., Pilipovecs, A., Kabashkin, I., Petersons, E.: Tyre Footprint Geometric Form Reconstruction by Fibre-Optic Sensor's Data in the Vehicle Weight-in-Motion Estimation Problem. Informatics in Control, Automation and Robotics, 10th International Conference, ICINCO 2013 Reykjavík, Iceland, July 29–31, 2013. Series: Lecture Notes in Electrical Engineering, vol. 325, pp. 123–137 (2015)
5. Grakovski, A., Pilipovecs, A., Kabashkin, I., Petersons, E.: Weight-in-motion estimation based on reconstruction of tyre footprint's geometry by group of fibre optic sensors. Transport Telecommun. **15**(2), 97–110 (2014)
6. Muhs, J., Jordan, J., Adams, K., Tobin, Jr., LaForge, J.: Apparatus for Weighing and Identifying Characteristics of a Moving Vehicle. US Patent 5.260.520, Nov. 9, 1993 (1993)
7. Kamenchenko, S., Grakovski, A.: Increased Safety of Data Transmission for "Smart" Applications in the Intelligent Transport Systems. Theory and Engineering of Complex Systems and Dependability. Series: Advances in Intelligent Systems and Computing, vol. 365, pp. 185–194 (2015)

Study of Dependencies Between Concrete Deterioration Parameters of Fly Ash-Based Specimens

Vlasta Ondrejka Harbuľáková, Adriana Eštoková and Alena Luptáková

Abstract Corrosion of concrete can lead to reduction of engineering properties such as strength, elastic modulus and durability and it may happen due to a number of various physical, chemical, and physic-chemical factors and their interactions reflecting peculiarities of both material aspects and external factors. In the paper, investigation of concrete specimens where the fly ash replacement was 10 wt% of cement content and they were exposed to aggressive environment of aluminium sulphate during the five 7-day cycles is presented. The leached-out masses of the Ca, Si, Fe, Al, Mn, and Cu, measured at the end of each cycle as well as pH values measured during different time period are published. Paper deals with the possibilities of investigation of deterioration parameters (chemical elements concentration, pH) of fly ash-based concrete specimens by correlation analysis. The correlation analysis confirmed the different leaching rate of calcium and silicon. However, the high mutual correlation was found between leaching of the silicon, iron, cuprum, aluminium and manganese.

Keywords Aluminium sulphate · Cement replacement · Chemical corrosion · Correlation

V.O. Harbuľáková (✉)
Faculty of Civil Engineering, Department of Environmental Engineering,
Technical University of Košice, Vysokoškolská 4, 04200 Košice, Slovakia
e-mail: vlasta.harbulakova@tuke.sk

A. Eštoková
Faculty of Civil Engineering, Department of Material Engineering,
Technical University of Košice, Vysokoškolská 4, 04200 Košice, Slovakia
e-mail: adriana.estokova@tuke.sk

A. Luptáková
Institute of Geotechnics, Slovak Academy of Science,
Watsonova 45, 04353 Košice, Slovakia
e-mail: luptakal@saske.sk

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_20

1 Introduction

In order to address a concrete dependability problem, important problems to be considered are: concrete composition mixture, water/cement ratio, composition of aggregates (fine and coarse) environment in which the building is placed and many more. A lot of structures are exposed environments which cause deleterious processes in concrete matrix: alkali aggregate reaction, chloride attack, effect of sea water, effect of de-icing salts, efflorescence and very aggressive sulphate and acid attack and that decrease its dependability. Sulphate attack is one of main factors causing deterioration of concrete durability. The deterioration of Portland cement concretes exposed to sulphates may be described by the following reactions [1]:

- The conversion of calcium hydroxide derived from cement hydration reactions to calcium sulphate (gypsum),
- The reaction of hydrated calcium aluminate of the Portland cement with calcium sulphate form calcium sulfoaluminate (ettringite).

Also the aquatic environment readily extract alkali compounds from the concrete pore fluid, lowering the pH and aggressively dissolves calcium hydroxide and decomposes calcium silicate hydrate which leads to gradually reduces the strength of the affected concrete until it disintegrates.

For example, in the United States and other parts of the world where U.S. standards have been adopted, the chemical part of the specification requires only a combined total of silica, alumina, and iron oxide. It does not specify the amount of silica that reacts with lime to produce added strength. The alumina content could be high in fly ash, which could be detrimental because more sulphate to control its reactivity might be required. Sulphate is added to the cement to control only the setting reactions of the aluminates and ferrites in the cement. However, the amount is limited because expansive reactions are possible after the concrete has set. This amount of sulphate does not take into account the extra aluminates that can be added when fly ash is used. Too much iron oxide will retard the setting time [2].

Previous studies have shown the positive influence of mineral additions, such as fly ash [3, 4], silica fume and blast furnace slag [5, 6] because of lower calcium hydroxide content, reduced Ca-to-Si (C/S) ratio in calcium silicate hydrates (C–S–H) and the refined pore structure they produce in concrete [3, 7, 8]. Different methods of maintaining the reliability of critical infrastructure were presented in [9–15].

The pozzolanic materials van be divided into two groups namely, natural pozzolanas and artifical pozzolanas. The typical examples of natural pozzolana are: clay, shales, opaline cherts, diatomacerous earth and volcanic tuffs and pumicites. The commonly used artificial pozzolanas are fly ash, blast furnace slag, silica fume, rice husk ash, metakaoline and surkhi. The pozzolanic materials when used as replacement are generally substituted for 10–50 % of cement. This substitution produces concrete that is more permeable but much more resistant to the action of salt, sulphate or acidic water [16]. Fly ash is by-product of burning pulverized coal in an electrical generating station [17]. The advantages of using fly ash far outweigh

the disadvantages. The most important benefit is reduced permeability to water and aggressive chemicals. Properly cured concrete made with fly ash creates a denser product because the size of the pores is reduced. The quality of fly ash is important–but it can vary. Poor-quality fly ash can have a negative effect on concrete. The principle advantage of fly ash is reduced permeability at a low cost, but fly ash of poor quality can actually increase permeability. Some fly ash, such as that produced in a power plant, is compatible with concrete. Other types of fly ash must be beneficiated, and some types cannot be improved sufficiently for use in concrete [2].

The paper presents the results of an investigation of concrete specimens with 10 wt% of cement replacement by fly ash influenced by an aggressive sulphate environment ($Al_2(SO_4)_3$). The correlations between the leached-out masses of the Ca, Si, Fe, Al, Mn, and Cu and pH values of leachates were analysed.

2 Materials and Methods

Concrete specimens were used for the experiments simulating the chemical corrosion of concrete. Set of concrete samples contained of 10 wt% of coal fly ash as cement replacement. The used coal fly ash with volumetric weight of 2381 kg/m^3 originated from black coal's burning process and was incorporated into cement composites without any modification. Concrete cylinder samples of a 32 mm diameter and 15 mm height were formed as a drilled core from concrete cube (150 × 150 × 150 mm), prepared according to a standard process, using drilling mechanism STAM. The cylinder concrete specimens were rid of impurity, dried and weighted, and exposed to a leaching medium represented by aluminium sulphate with a pH value of 3.22. Liquid medium used for experiment was prepared by dissolving 171.1 g of $Al_2(SO_4)_3$ in distilled water and filled into 1000 mL.

Chemical corrosion-simulating-experiments proceeded in five consecutive cycles. Each cycle consists of the following steps: a 7-day exposition of specimens to a liquid medium, a removal the specimens from the liquid, a 2-day drying of specimen at room temperature and afterwards a removing of precipitations by little brush, a re-immersion of specimen into the medium and finally adjustment of the pH value back to the initial value. Measurements of pH values were done at the beginning, after 30 min contact liquid media with concrete samples in each cycle and at the end of each cycle. The values of leachate´s pH were measured by pH meter PHH—3X Omega. Concentrations of elementary ions in leachates after each experimental cycle were determined using X-ray fluorescence analysis (XRF). The measured concentrations of dissolved ions and the pH values measured in leachates were used for the statistical investigation.

In statistics, dependence refers to any statistical relationship between two random variables or two sets of data. Correlation refers to any of a broad class of statistical relationships involving dependence. Descriptive statistics is the discipline of quantitatively describing the main features of a collection of data [18]. Increase of the absolute value of the R_{xy} (correlation coefficient) is proportional to linear

correlation. Information about two dimensional statistical data set gives correlation coefficient R_{xy} as is shown in Eq. (1).

$$R_{xy} = \frac{n\sum_{i=1}^{n} x_i y_i - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} y_i\right)}{\sqrt{\left[n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i^2\right)^2\right]\left[n\sum_{i=1}^{n} y_i^2 - \left(\sum_{i=1}^{n} y_i^2\right)^2\right]}} \quad (1)$$

R_{xy} values are from the interval <−1, 1>. If $R_{xy} = 1$, the correlation is full linear, if $R_{xy} = -1$, then the correlation is inversely linear and if $R_{xy} = 0$, the pairs of values are fully independent. Than degree of the correlative closeness is: medium, if $0.3 \leq |R_{xy}| < 0.5$; significant, if $0.5 \leq |R_{xy}| < 0.7$; high, if $0.7 \leq |R_{xy}| < 0.9$; and very high, if $0.9 \leq |R_{xy}|$.

Correlation coefficient was for the purposes of our assessment obtained by the function "Pearson" in Microsoft Excel.

3 Results and Discussion

Before applying the statistical approach using Pearson correlation coefficient, the results of the chemical corrosion experiment of concrete samples with different chemical composition are presented in this section. pH changes and leached-out concentrations dependence on time were studied with the aim to find out the dependencies between studied parameters and reveal the trend of the corrosion process. Consequently, for more precise conclusion, these results were used as input data for a dependency analysis between the measured parameters while statistical results are presented the next part of the paper.

Changes in pH values of aluminium sulphate solutions after 30 min of immersion of the concrete specimens into $Al_2(SO_4)_3$ are presented in Fig. 1. The differences in the values of pH were calculated by comparing the pH measured after 30 min of specimens´ immersion and the original ones (pH 3.22) in a particular cycle.

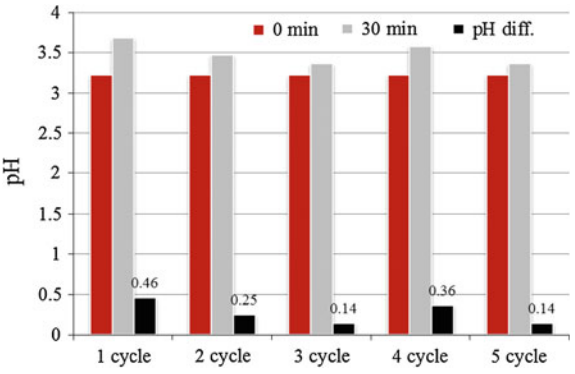


Fig. 1 Changes in pH after 30 min of samples' immersion

As seen in Fig. 1 the differences in pH values oscillated while ranging in an interval <0.14−0.46>. An increase of pH even after 30 min of the immersion of the samples was likely caused by instant leaching of the alkali compounds from the concrete materials. The difference between the original pH in the cycle and the pH measured after 30 min was getting lower up to the third cycle which points to the facts that the leaching of the alkali compounds got slower and the equilibrium between leached-out ions and the cement matrix was stated faster. After the third cycle, passing first 30 min in the next cycles, the differences in pH values oscillated while first increased up to 0.36 and further decreased to 0.14.

The original pH value of aluminium sulphate and pH measured in leachates at the end of each 7-day cycle are presented in Table 1.

Obviously, higher pH values were measured after each cycle when compared to the 30-min ones. Comparing the final pH of media in all cycles, the decreasing trend of pH measured at the end of cycles was observed. The final pH decreased form the value of 4.49 measured after the first cycle to the value 3.89 measured after the last fifth cycle (Table 1). The founding confirmed that the leaching rate of concrete materials decreased as mentioned above with the time of exposure.

For better understanding, statistical approach was used for a study of the dependability trend between pH of media and leached-out concentrations of selected ions. From the wide range of results, only leached-out concentrations of calcium, silicon, aluminium, iron, manganese and cuprum were presented in this paper. The leached-out masses of the Ca, Si, Fe, Al, Mn, and Cu, measured at the end of each cycle, were depicted to the graphs (Figs. 2, 3, 4, 5, 6 and 7).

Figures 1 and 2 present different trends in Ca^{2+} and Si^{4+} as concrete basic chemical elements leaching. In case of calcium the highest leached-out mass was measured at the end of the first cycle and then the leaching of calcium became slower. Different situation was observed for silicon leaching, where the maximum leached-out mass of Si was measured after the third cycle, so in the middle of the

Table 1 Final pH values of liquid media after each cycle

	Initial	Cycle 1	Cycle 2	Cycle 3	Cycle 4	Cycle 5
pH	3.22	4.49	4.23	4.18	3.91	3.89

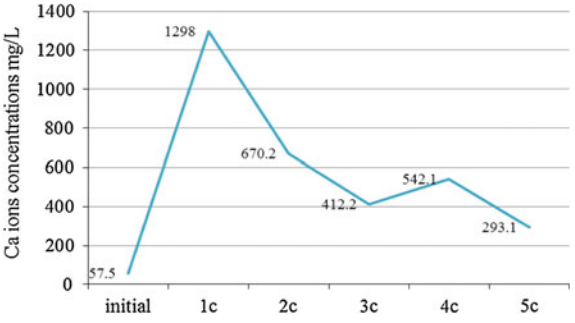


Fig. 2 Trend of Ca leaching in $Al_2(SO_4)_3$

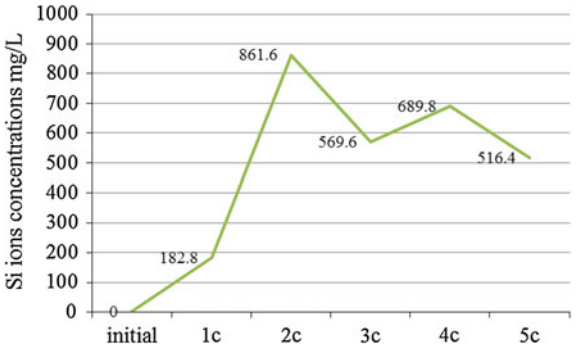


Fig. 3 Trend of Si leaching in $Al_2(SO_4)_3$

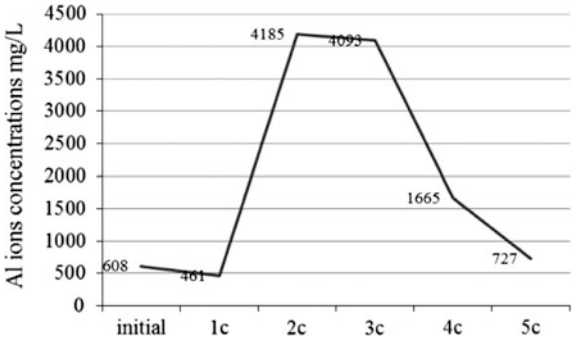


Fig. 4 Trend of Al leaching in $Al_2(SO_4)_3$

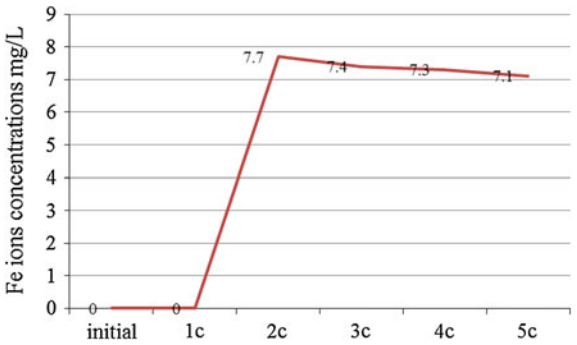


Fig. 5 Trend of Fe leaching in $Al_2(SO_4)_3$

experiment. This difference in maximum measured leached-out mass points to the different leaching rates of calcium and silicon in the aluminium sulphate solution during the experiment.

When comparing the leaching trends of the other elements, there was found a slightly similarity between aluminium, manganese and cuprum leaching as seen in Figs. 4, 6 and 7. The leaching curves reached a maximum after the second, third or fourth cycles for aluminium, cuprum and manganese, respectively. For iron, the

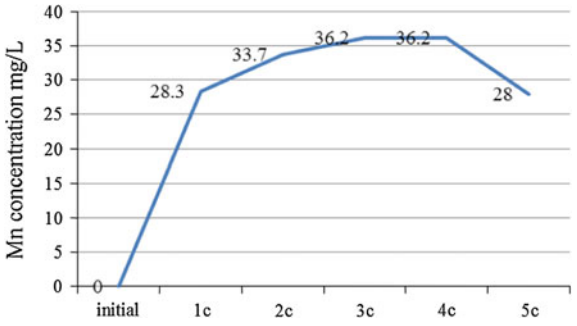


Fig. 6 Trend of Mn leaching in $Al_2(SO_4)_3$

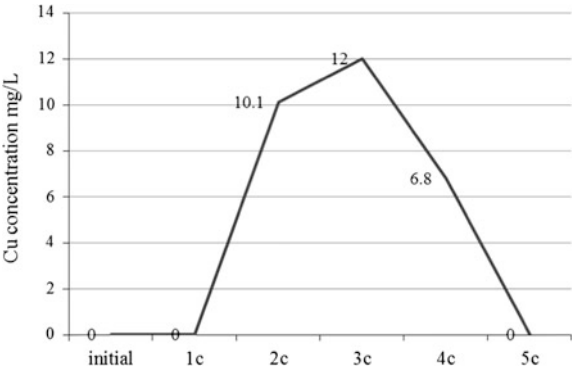


Fig. 7 Trend of Cu leaching in $Al_2(SO_4)_3$

leaching started in the second cycle as seen in Fig. 5 and proceeded at about the same leaching trend up to the last cycle. The leached-out mass of cuprum was extremely low and was close to zero at the end of the experiment. This could be caused by a reaction of the leached cuprum with the media by forming sulphates or by a low detection limit of analytical equipment.

As mentioned above, the presented results were applied as input data for the statistical analysis with the aim of better understanding of deterioration processes trend. Based on the concentrations of ions dissolved into the liquid phase of aluminium sulphate presented above (Figs. 2, 3, 4, 5, 6 and 7) and pH values (Table 1) the statistical evaluation was performed. Results of the calculated correlation coefficients (R_{xy}) for Ca/pH, Si/pH, Al/pH, Fe/pH, Cu/pH, and Mn/pH dependences for the studied concrete specimens are presented in Table 2.

The highest correlation coefficient was calculated for calcium leaching versus pH (R_{xy} = 0.84) with a correlation closeness defined as high. Correlation closeness was

Table 2 Correlation coefficients between leached-out masses of ions and pH of media

Correlated parameter	Ca/pH	Si/pH	Fe/pH	Al/pH	Mn/pH	Cu/pH
Correlation coefficient R_{xy}	0.84	0.44	0.25	0.36	0.27	0.35

found as medium for relation pH and silicon (R_{xy} = 0.44); pH and aluminium (R_{xy} = 0.36); and pH and cuprum (R_{xy} = 0.35). No correlation ($R_{xy} \leq 0.3$) was observed for iron and manganese leaching versus pH as is shown in Table 2.

The high found-out correlation between dissolved Ca^{2+} amount and pH of leachate resulted in the fact that the calcium is dominantly leached-out from the cement matrix in form of its alkali compounds. On the contrary, silicon´s compounds leaching was not observed to contribute so significant to the pH moving to the alkali region. The results of the correlation analysis of other elements studied (Fe, Cu, Mn, and Al) confirmed that the leaching of these is no based on the alkali compounds.

To study the similar or different trends in leaching of the analysed ions, a correlation analysis between leached-out concentrations of related ions dissolved from the concrete specimens to the liquid media was performed. The calculated correlation coefficients regarding the analysed ions are presented in Table 3.

The correlation found out between the leached-out amounts of calcium and the other elements in particular cycles during the experiment is fully independent. This confirms the different rates and mechanism of leaching when comparing calcium and silicon or calcium and iron, aluminium, cuprum and manganese what is in accordance with our previous results [19, 20].

On the other hand, a very high correlation was found between silicon and iron, cuprum and manganese, and cuprum and aluminium leaching, respectively, with the correlation coefficients $R_{xy} > 0.9$ (Table 3). A high correlation having $R_{xy} > 0.7$ was calculated for the couples Si/Al, Si/Mn, Si/Cu, Fe/Mn, and Fe/Cu. This means that the more massive was leaching of silicon the more massive was leaching of iron, aluminium, manganese and cuprum. The analogical relation was observed for iron and manganese and iron and cuprum, respectively.

Table 3 Correlation coefficients between leached-out concentrations of ions

Correlated ions	Correlation coefficient R_{xy}
Ca/Si	0.05
Ca/Fe	−0.23
Ca/Al	−0.04
Ca/Mn	−0.03
Ca/Cu	−0.03
Si/Fe	0.93
Si/Al	0.72
Si/Mn	0.72
Si/Cu	0.75
Fe/Al	0.67
Fe/Mn	0.70
Fe/Cu	0.70
Cu/Mn	0.92
Cu/Al	0.96
Al/Mn	0.67

4 Conclusion

Detail studies of behaviour of particular chemical components of concrete matrix and relation among them may be the key in the sphere of dependability, reliability and durability of concrete structure. The correlation analysis used for the evaluation of the results of the fly ash-based concrete's chemical corrosion modelling confirmed the different leaching rate of calcium and silicon. However, the high mutual correlation was found between leaching of the silicon, iron, cuprum, aluminium and manganese.

The statistical approach confirmed to be really helpful in understanding and evaluating the experimental results. The findings revealed that by monitoring one or only a few leached-out elements from the concrete composites in the leachate we could be able to predict the leaching of the other elements. This could be a very useful and simplistic approach in a process of concrete material's deterioration monitoring.

To conclude the findings in general, a more detailed research is needed in various sulphate media together with using more sophisticated mathematical methods.

Acknowledgments This research has been carried out within the Grants No. 2/0145/15 and 1/0481/13 of the Slovak Grant Agency for Science.

References

1. Winter, N.: Understanding Cement. WHD Microanalysis Consultants Ltd, United Kingdom (2009)
2. Rosenberg, A.: Using Fly Ash in Concrete, NPCA, Precast Magazines Archive 2009–2010
3. Bertron, A., Duchesne, J., Escadeillas, G.: Attack of cement pastes exposed to organic acids in manure. Cem. Conr. Compos. **27**, 898–909 (2005)
4. De Belie, N., Debruyckere, M., Van Nieuwenburg, D., De Blaere, B.: Attack of concrete floors in pig houses by feed acids: influence of fly ash addition and cement-bound surface layers. J. Agric. Eng. Res. **68**(2), 101–108 (1997)
5. Bertron, A., Escadeillas, G., Duchesne, J.: Cement pastes alteration by liquid manure organic acids: chemical and mineralogical characterization. Cem. Concr. Res. **34**(10), 1823–1835 (2004)
6. De Belie, N., Verschoore, R., Van Nieuwenburg, D.: Resistance of concrete with limestone sand or polymer additions to feed acids. Trans. ASAE **41**(1), 227–233 (1998)
7. Gruyaert, E., Van den Heede, P., Maes, M., De Belie, N.: Investigation of the influence of blast-furnace slag on the resistance of concrete against organic acid or sulphate attack by means of accelerated degradation tests. Cem. Concr. Res. **42**(1), 173–185 (2012)
8. De Belie, N., De Coster, V., Van Nieuwenburg. D.: Use of fly ash or silica fume to increase the resistance of concrete to feed acids. Mg. Concr. Res. **49**, 337–344 (1997)
9. Bary, B., Sellier, A.: Coupled moisture-carbon dioxide-calcium transfer model for carbonation of concrete. Cem. Concr. Res. **34**(10), 1859–1872 (2004)
10. Kong, J.S., Ayman, N., Ababneh, D.M., Frangopol, Y.X.: Reliability analysis of chloride penetration in saturated concrete. Probab. Eng. Mech. **17**(3), 305–315 (2002)

11. Bažant, Z.P., Steffens, A.: Mathematical model for kinetics of alkali–silica reaction in concrete. Cem. Concr. Res. **30**(3), 419–428 (2000)
12. Jonkers, H.M., Thijssen, A., Muyzer, G., Copuroglu, O., Schlangen, E.: Application of bacteria as self-healing agent for the development of sustainable concrete. Ecol. Eng. **36**(2), 230–235 (2010)
13. Berryman, Ch., Zhu, J., Jensen, W., Tadros, M.: High-percentage replacement of cement with fly ash for reinforced concrete pipe. Cem. Concr. Res. **35**(6), 1088–1091 (2005)
14. Pietrucha-Urbanik, K.: Failure prediction in water supply system—current issues, theory and engineering of complex systems and dependability. Adv. Intell. Syst. Comput. **365**, 351–358 (2015)
15. Pietrucha-Urbanik, K.: Assessment model application of water supply system management in crisis situations. Global NEST J. **16**(5), 893–900 (2014)
16. Thomas, M.: Optimizing the Use of Fly Ash in Concrete, Portland Cement Association, Publication IS 548 p. 24 (2007)
17. Gambhir, M.L.: Concrete Technology: Theory and Practice, McHill Education (India) Private Limited (2003)
18. Kreyszig E.: Advanced Engineering Mathematics, 10th edn. Wiley (2011)
19. Ondrejka Harbuľáková, V., Purcz, P., Eštoková, A., Luptáková, A., Repka, M.: Using a statistical method for the concrete deterioration assessment in sulphate environment. Chem. Eng. Trans. **43**, 2221–2226 (2015)
20. Ondrejka Harbuľáková, V., Eštoková, A., Purcz, P., Luptáková, A., Statistical dependence investigation of deleterious process on concrete samples exposed to different media. In: CEST 2015, Athens, CEST, pp. 1–5 (2015)

Influence of Data Uncertainty on the Optimum Inspection Period in a Multi-unit System Maintained According to the Block Inspection Policy

Anna Jodejko-Pietruczuk and Sylwia Werbińska-Wojciechowska

Abstract In the presented paper, authors focus on delay-time-based maintenance modelling issues. They present an analytical maintenance model for multi-unit systems. Implemented maintenance policy is the Block-Inspection Policy (BIP) that assumes performing inspection actions at regular time intervals of T. In the next step, the problem of data uncertainty is analysed. There is also investigated the sensitivity analysis of the proposed cost model due to probability distributions of system elements' initial and delay times parameters change.

Keywords Delay time approach · Block-inspection policy · Parameter estimation

1 Introduction

Maintenance problems of technical systems have been of significant importance for the past 50 years, especially because of the applications in industry, transportation, or civil engineering (see e.g. [2, 13, 18–20, 22, 23, 34]). Thus, the published papers which contribute to maintenance theory cover the full range of potential models for maintenance management problems.

When technical systems/elements are being inspected, potential defects may be identified and removed in order to prevent future failures occurrence. As a result, the determination of inspection intervals is one of the main decision problems of any maintenance managers [28]. However, following the literature (see e.g. [16, 27]), it may be stated that the interval between planned maintenance interventions is usually determined subjectively by manager's experience or is given as a manufacturer's recommendation (e.g. written in service manuals). This may be

A. Jodejko-Pietruczuk (✉) · S. Werbińska-Wojciechowska
Wroclaw University of Technology, 27 Wybrzeze Wyspianskiego Street, Wroclaw, Poland
e-mail: anna.jodejko@pwr.edu.pl

S. Werbińska-Wojciechowska
e-mail: sylwia.werbinska@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*, Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_21

connected with the lack of knowledge of maintenance managers who do not implement the known maintenance models (see e.g. [16]). On the other hand, the maintenance decision may have a multidimensional nature and depend on the problems connected with definition if and what kind of signal of forthcoming failure may occur. For example, the squealing of vehicle belt during its operation process performance indicates the need for its preventive replacement. However, the occurred noise during operation processes performance of construction machines (e.g. wheel loader) will only indicate a problem in the drive system, which may regard to e.g. differential problems or transmission problems (e.g. propeller shaft). Another problem is that some inspection (diagnostic) actions may be performed only during planned maintenance actions performance. A good example here is the planned replacement of engine oil in construction machines. During this maintenance operation performance, one can use a sample of used oil and examine it for the content of metallic elements. The increased level of their content may attest to the usage of certain engine system components, such as wear out of cylinders, piston rings or bearings. The mentioned problems may be solved with the use of delay-time based maintenance modelling.

The delay-time concept is developed by Christer in 1976 (see e.g. [3, 11]). The basic idea rests on an observation that a failure does not usually occur suddenly, but is preceded by a detectable fault for some time prior to actual failure—the delay time [7]. So, the delay time h is defined as the time lapsing from the moment when a fault could first be noticed till the moment when a subsequent failure occurs, if left unattended [4, 21]. For recent literature surveys of the analysed research area we recommend reading [4, 5, 29].

The successful implementation of developed delay-time model depends upon the estimation process of its parameters from available information sources. This problem includes the necessity to obtain failure times and repair times, the rate of defects occurrence at time u, the cumulative probability function of delay time h, and the probability of defects identified at PM [33]. Generally, there is no possibility to measure directly either the delay time associated with a defect, or the initial point u. There can be proven a possibility to estimate the delay time for a set of specific faults and failures, and from this deduce the location of the initial point and estimate the delay-time and initial-point distributions [7]. The complexity of analysed problem is confirmed by many previous studies that have paid attention to the issues of parameter estimation in delay-time modelling.

If there are available sufficient maintenance data about failures occurrence and maintenance interventions such as inspections, the delay time distribution can be estimated by classical statistical methods. The most often used are the method of maximum likelihood or statistical process control methods (see e.g. [30, 31]). On the other hand, there is used the method which bases on subjective data obtained from maintenance engineers' experience (see e.g. [25, 26]).

The earliest works in this area were Christer [3], Christer and Whitelaw [11], and Christer and Waller [8–10], where authors developed a method for using subjective opinions of experts in estimation of the delay time distribution. The subjective estimation of delay time is later discussed e.g. in [1, 6, 12]. Recently, a series of

research studies introduced the objective estimation of model's parameters: see e.g. [17, 24, 27, 32]. All of these researches introduced the new extended formula of the likelihood function of the delay time modelling adjusted to the analysed problems and case studies.

However, in these previously developed studies the parameter estimation of the delay time models was solved with the use of simulation data, subjective data, or combination of subjective and objective data. This situation was connected with not sufficient maintenance data which could allow the use of fully objective approach to solve the modelled problem [25]. On the other hand, the developed formula of the likelihood functions of the delay time models was usually quite complex to solve it by traditional derivation methods [24]. Following [29], the simple and robust estimating algorithm for delay-time model parameters is a task not yet accomplished but necessary to automate delay-time model applications. Thus, in this paper authors focus on the situation when reliable operational and maintenance data are not possible to obtain because of some technical or economic reasons and only some limited characteristics may be evaluated with the methods discussed above. In such a situation, it is extremely important to have the knowledge what consequences may result from a wrong parameters' estimation process performance and which of characteristics are really necessary to get to know in order to correctly decide on delay-time model decision variables. Following this, in this paper authors aim to answer the questions:

1. what is the importance of probability distribution parameters on the Block Inspection Policy cost in a multi-unit systems,
2. which of them should be accurately estimated on the base of objective data,
3. when simple decision rules are enough to apply when one wants to determine the best (or just profitable) inspection period?

The performed analysis is based on the analytical delay-time model developed and presented e.g. in [14].

2 Problem Description, Notations and Assumptions

To stage the problem, here we consider a multi-unit system subject to periodic inspection. An implemented maintenance policy is the Block Inspection policy—one of the most commonly used in practice. The performed inspections are carried out in order to check the working status of the system or its elements and take place at regular time intervals of T. Each inspection action requires a constant time and gives information whether the system is normally working, or is in a defective state, which is in need of an immediate attention. An example of this type of maintenance policy may regard to the periodic inspections of break pressure in e.g. wheel loaders. Such break pressure checking is performed in every 1000 machine working hours.

The inspections are assumed to be perfect. As a result, system defect which occurs till the moment of inspection will be identified and replaced within the inspection period. Moreover, failure is observed immediately and the system is replaced at a given cost and downtime. The main decision variable in the model is the inspection interval/cycle. For a modelling purpose and additional parameters' estimation procedure, we propose the following notation.

2.1 Notation

c_i unit inspection cost
c_f unit consequence costs incurred in the case of system failure
c_r unit replacement cost of an element
$C_I(T)$ expected cost of inspection action performance per a single inspection cycle with the length T
$C_R(T)$ expected cost of preventive replacement performance per a single inspection cycle with the length T
$C_F(T)$ expected cost of corrective replacement performance per a single inspection cycle with the length T
$C(T)$ expected system maintenance costs per an unit of its operating time
$K(T)$ cost coefficient used to determine the best inspection interval
$T_M(T)$ total expected length of operating time assuming a single inspection cycle has the length T
h delay time of the defect
u initial time of the defect
$G_j(u)$ Cumulative distribution function of the initial time u of jth component
$F(x)$ Cumulative distribution function of a system's time to failure
$R(x)$ Reliability function of a system
k minimal number of up-stated elements for having system in an operational state
n number of elements working in a system.

2.2 Assumptions

Based on the problem description, the following main modelling assumptions are proposed to characterize the operation and maintenance of a technical system:

- the system is a three state system where, over its service life, it can be either operating, operating acceptably or down for necessary repair or planned maintenance,
- inspection action performance begins the new inspection cycle for the analysed system and inspections are carried out periodically on the system,

- the system can remain functioning in an acceptable manner until breakdown (despite having defects),
- the breakdown will be assumed to have been caused by $n - k + 1$ of the defects which have deteriorated sufficiently to affect the operating performance of the system as a whole,
- components are prone to become defective independently of each other when the system is in operating,
- defects which may have arisen in the system, deteriorate over an operating time,
- all elements with identified defects will be replaced within the inspection performance,
- components have their own *pdfs* of initial and delay time,
- the component replacement costs are the same for all elements.

3 Delay-Time Model for Multi-unit System

In this section, there is presented the cost delay-time-based maintenance model for multi-unit system performing in *k-out-of-n* reliability structure and a simple method of determining the near-optimum inspection period length T. The structure of a modelled system has been chosen for further analysis because it is the most general case of a reliability structure of a system and makes it possible to model great majority systems existing in practice. The examples of series systems may regard e.g. to vehicles fuel supply system with one common rail unit in a three-cylinder engine performance (3-*out-of* 3) Series structures are commonly used also e.g. in construction equipment. A systems with parallel structure are commonly found in NPP or aviation and in lighting systems. Figure 1 shows a scenario of defect arrivals, failures and inspections of an exemplary system performing in 2-*out-of*-3 reliability structure and using the BIP.

Following Fig. 1, it is assumed that system elements work independently under the same conditions. In the analysed system may be performed one of the two maintenance operations: failure repair or inspection together with possible replacement of elements with defects. Following this, it is assumed that when a system failure occurs, there is only performed replacement of failed components without additional inspection action performance. However, in the case of planned inspection action performance, the replacement will be performed only for those elements with visible symptoms of their future damage.

3.1 Expected Maintenance Costs Model

For a multi-unit system, the long-term expected maintenance costs per unit of operating time may be estimated according to the formulae [14]:

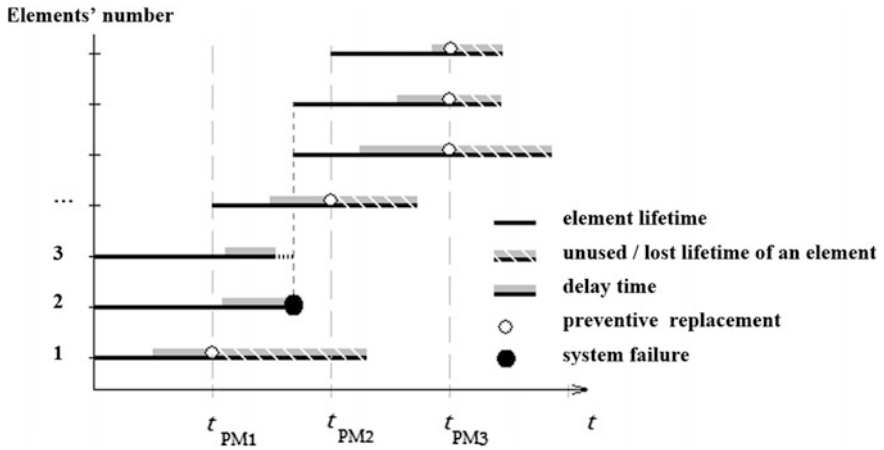


Fig. 1 The example of block inspection policy implementation in the case of a three-element system working in 2-*out-of*-3 reliability structure

$$C(T) = \frac{C_I(T) + C_R(T) + C_F(T)}{T_M(T)} \tag{1}$$

where

$$C_I(T) = c_i(1 - F(T)) \tag{2}$$

$$C_R(T) = c_r(1 - F(T))\left(\sum_{j=1}^{n} G_j(T)\right) \tag{3}$$

$$C_F(T) = c_f F(T) \tag{4}$$

The maintenance cost expressed in the Eq. (1) represents the sum of possible cost: of a system failure, replacement cost of working elements with observable defects and inspection costs per a single inspection period. There should be underlined here, that the developed mathematical model gives the possibility for estimation of expected maintenance costs for multi-unit system, whose all elements are as good as new at the beginning of an inspection cycle of length T (e.g. first inspection cycle performance).

The total expected cycle length, T_M may be given by the following formulae:

$$T_M(T) = \int_0^T R(u)du \tag{5}$$

The reliability function, presented in the Eq. (5) is defined in more detail e.g. in works [14, 15] for systems with *k-out-of-n* reliability structure. The convergence of

the presented analytical delay-time model results with the simulation ones was presented in [14].

3.2 The Use of the Costs Model to Determine the Best Inspection Time Interval

The cost models presented in Sect. 3.1 and work [psam] may be used to support the process of searching of the best length of inspection period for given reliability features of a system. According to the proposed method a set of calculations of the BIP policy cost should be executed for a range of possible values of inspection periods T. The calculations (according to Eqs. (1–4)) should be conducted taking into account a reliability structure of a system and assuming that all its elements are *as good as new* at the beginning of an inspection period. The results may be used while determining the best (or near the best) inspection period for a system operating in a long time horizon, when the assumption about initial elements' state is not satisfied. To set a value of the best inspection period one should consider the relation of the two most important cost components of the BIP policy: the expected failure and replacement costs, and calculate their difference ($K(T)$):

$$K(T) = C_F(T) - C_R(T) \tag{6}$$

The K coefficient makes it possible to find its minimum value and in this way—to indicate periods T, for which a system yields the best cost results of the BIP policy. In a period pointed out by the minimum of the K coefficient, independently on a system reliability structure, the probability of elements' preventive replacement is high (what means the defects have arisen in majority of system elements) whereas the probability of a system failure is still low (there is a chance to prevent a system failure). This simple method is discussed in details in [14], where also its convergence with a simulation research has been presented.

Exemplary graphs of the K coefficient for parallel (1-*out-of*-3) and series (3-*out-of*-3) systems are shown in the Fig. 2 and corresponding costs of C_I, C_R, C_F for a series system are presented in the Fig. 3.

The literature review presented in the first section of the paper emphases the problem that often arises in practice due to the lack of precise maintenance and operational data. The authors use the proposed cost model to examine what is the importance of various model parameters for the BIP costs and for right determination of an inspection period T in a *k-out-of-n* system. The examination should be the foundation to answer the questions asked in the first paragraph of the paper. For this reason, the next section contains the sensitivity analysis of the proposed cost model due to change of probability distributions characteristics of system elements' initial and their delay times.

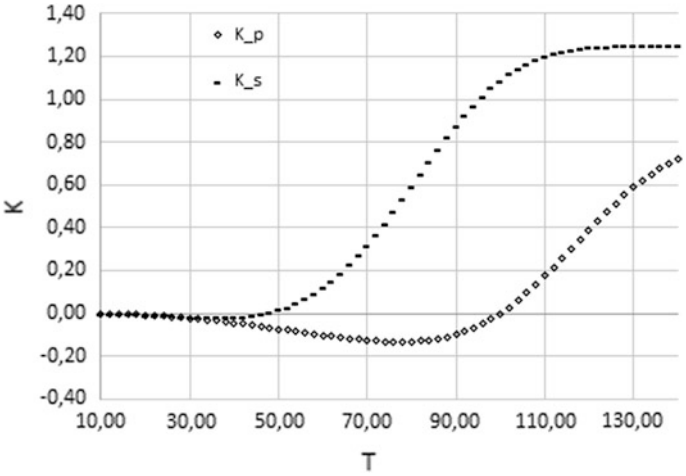


Fig. 2 The *K* coefficient for 1-*out-of*-3 ("p" index) and 3-*out-of*-3 ("s" index) systems

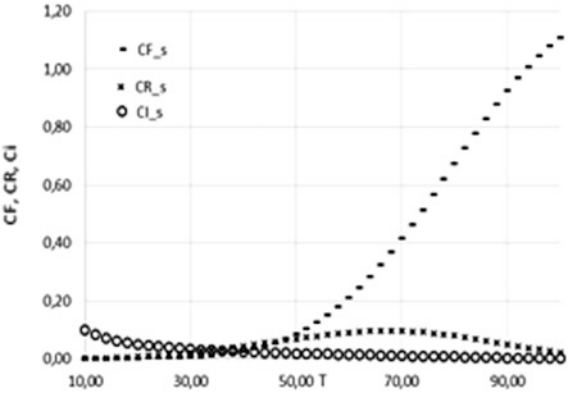


Fig. 3 The expected costs of: a system failure (*CF_s*), elements' preventive replacement (*CR_s*) and inspection (*CI_s*) for 3-*out-of*-3 system

4 The Influence of Probability Distribution Parameters on Cost Model Results

The BIP may be implemented in technical systems when some information about reliability characteristics is known. The basic reliability parameters that have to be specified in such systems are: an estimation of system components' time to signal of a future failure appearance (time to their defect) and delay time characteristics. These two times, together with other data (e.g. a system reliability structure, unit costs, etc.) give the base to optimize an interval between inspections that may provide the best economical results. During the process of delay time characteristics estimation, the perfect case is to gather enough objective data to specify the

probability distribution of this random variable. However, maintenance engineers usually are able to assess (more or less precisely) only the basic expected times u and h, and sometimes to enrich it with the standard deviation estimator. Thus the following analysis aims at comparison of being found the best inspection intervals for various detailed parameters of the probability distributions (and in this way—their various forms of probability densities), assuming that the expected initial time and the expected delay time are constant. The assumption of constant expected times while changing other probability characteristics reflects the mentioned fact that engineers much more often are able to estimate mean time to signal of future failure appearance and mean delay time, than full characteristics of times' probability distributions.

Assuming that an evaluations of expected initial time u and delay time h are known, the analysis of system costs results has been done for a chosen range of parameters of times u and h probability distributions. Changes in distributions' parameters with constant expected values of distributions have resulted in changes of standard deviations and hazard rate of the distributions.

The authors use Weibull probability distribution with its parameters yielding the same expected times u and h but different hazard rate. The expected initial time of a system element is assumed to be 65, while its expected delay time is equal 35 in further analysis. A few examples of a hazard rate resulting from various values of shape and scale parameters of the chosen Weibull distribution with constant delay time are presented in the Fig. 4.

Figures 5, 6, 7 and 8, illustrate the influence of probability parameters of times u and h upon the system reliability level R at the end of a single inspection cycle of length T. The studied range of shape parameters au and ah is between 0.5 ÷ 15. This is mostly connected with having no practical application of the values au, $ah \leq 1$, when a hazard rate of such distribution is decreasing or constant. The hazard rate of defect initial time u or delay time h is usually strictly increasing due to the fact that defects occur as a result of technical system deteriorating. System reliability that

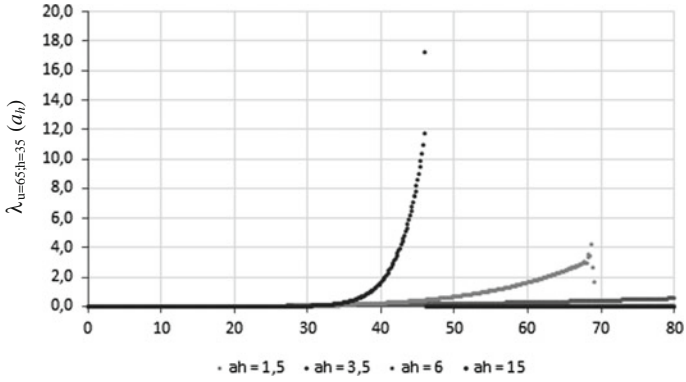


Fig. 4 Hazard rate of system elements' delay times assuming various values of a shape parameter "ah" in the Weibull probability distribution

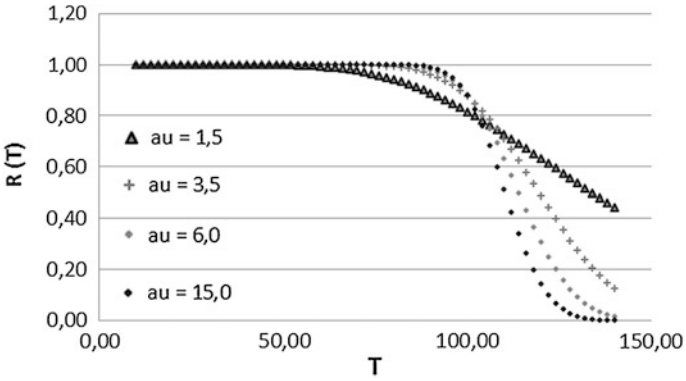


Fig. 5 Reliability of 1-*out-of*-3 (parallel) system in the inspection moment assuming various shape parameters "au" in the Weibull probability distribution of system elements' initial times *u*

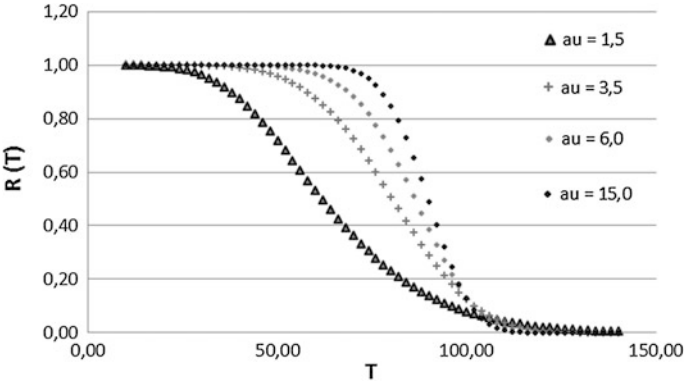


Fig. 6 Reliability of 3-*out-of*-3 (series) system in the inspection moment assuming various shape parameters "au" in the Weibull probability distribution of system elements' initial times *u*

directly affects all the costs components of the BIP, relatively strongly depends on probability characteristics of initial time *u* in a series system (Fig. 6) but when cases of strictly increasing hazard rate are considered (au ≥ 3.5), the dependency becomes much slighter. The dependency is also not very strong when probability distribution of delay time is analysed (Fig. 8) and in the parallel case (Figs. 5 and 7), especially when it comes to very long inspection intervals ($T \geq 50$).

The dependency of hazard rates of elements' times *u* and *h* upon the system reliability may be observed in some part of the tested inspection intervals *T*. In order to examine the direct effect of probability characteristics on cost results of BIP application in a multi-unit system, their influence on every kind of cost component has been tested. A few exemplary results obtained for a series system (more sensitive to hazard rate resulting from probability distributions) and for various forms of initial time probability distribution are presented in Figs. 9, 10, 11 and 12.

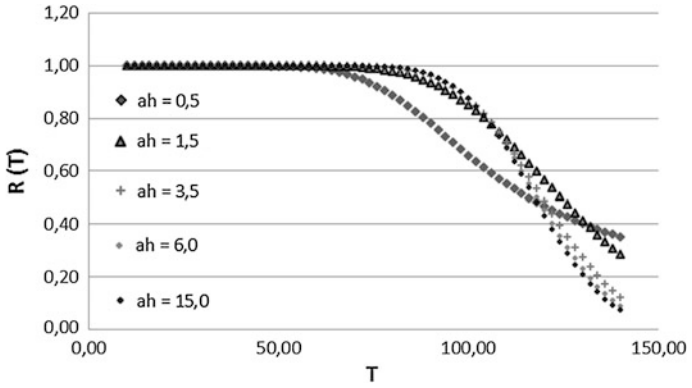


Fig. 7 Reliability of 1-*out-of*-3 (parallel) system in the inspection moment assuming various shape parameters "ah" in the Weibull probability distribution of system elements' delay times *h*

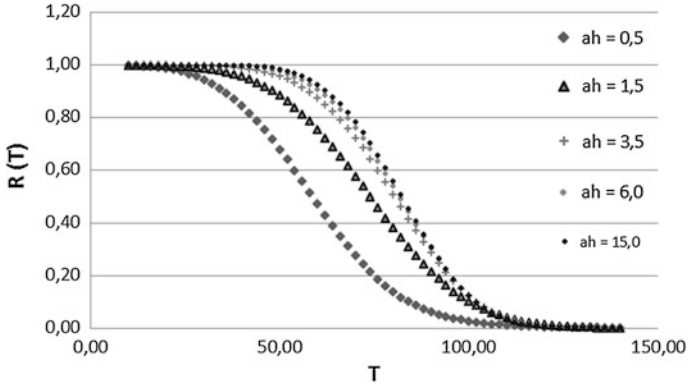


Fig. 8 Reliability of 3-*out-of*-3 (series) system in the inspection moment assuming various shape parameters "ah" in the Weibull probability distribution of system elements' delay times *h*

The presented cost results confirm the necessity of precise estimation of these parameters, if the costly-optimum inspection interval is to be taken in a multi-unit system with a series reliability structure. The influence of probability characteristic tested in the research upon system costs is much slighter, especially in a parallel system. In order to illustrate the relation between tested parameters and optimum cost results for systems with both reliability structures, the additional research has been conducted:

- all the tested periods T has been ranged for the sake of obtained cost results,
- the ith (i = 1, 5, 10, 60) cheapest period T found during the research have been compared among others and chosen results (for *au*) are presented in Figs. 13 and 14, while Figs. 15 and 16 depict coefficients K related to the ith best period.

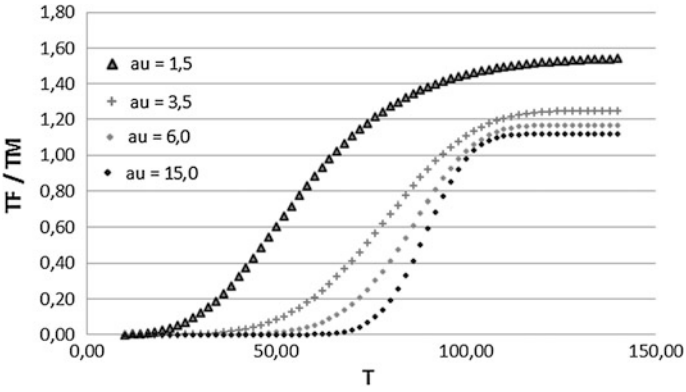


Fig. 9 The expected failure cost per unit of operating time in 3-*out-of*-3 system assuming various shape parameters "au" in the Weibull probability distribution of system elements' initial times *u*

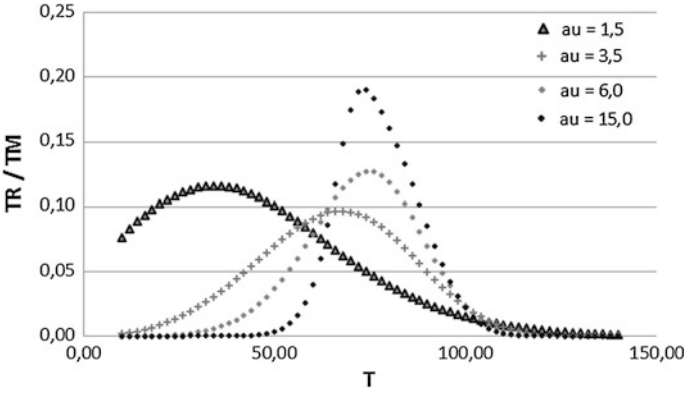


Fig. 10 The expected replacement cost per unit of operating time in 3-*out-of*-3 system in the inspection moment assuming various shape parameters "au" in the Weibull probability distribution of system elements' initial times *u*

As it may be seen in Figs. 13, 14, 15 and 16, the hazard rate of initial time (determined by a shape parameter *au* of the Weibull probability distributions) has a strong influence on the best length of an inspection cycle *T* only for the cases when the rate is decreasing ($au < 1$), constant ($au = 1$), or lightly increasing ($au < 3.5$). Thus, maintenance managers of technical systems (especially parallel ones), composed of elements with such characteristic of initial and/or delay times (similar effect has been observed when delay time parameters were studied), should pay a great attention to precise estimation of probability parameters of times *u* and *h*. The

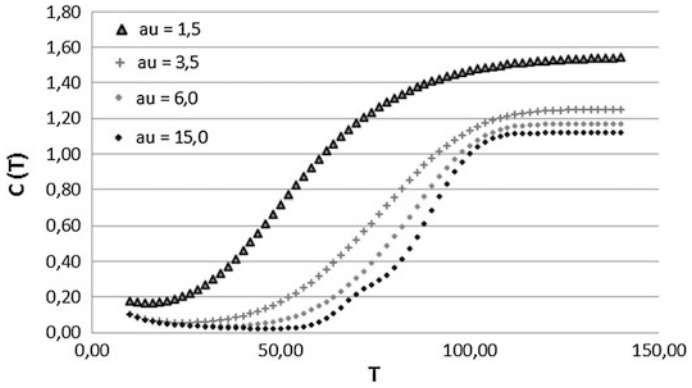


Fig. 11 The expected BIP cost per unit of operating time in 3-*out-of*-3 system in the inspection moment assuming various shape parameters "ah" in the Weibull probability distribution of system elements' delay times *u*

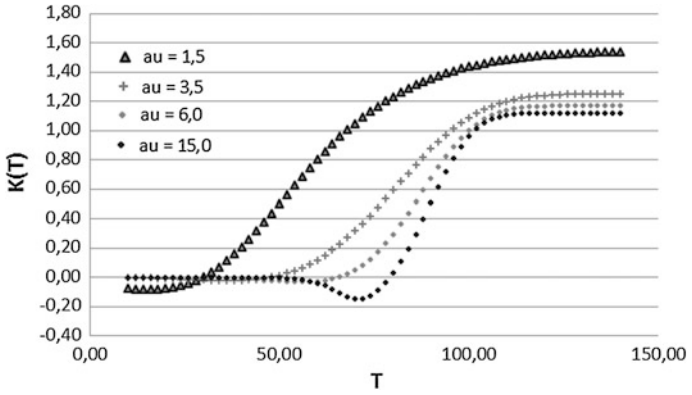


Fig. 12 *K* coefficient values in 3-*out-of*-3 system in the inspection moment assuming various shape parameters "ah" in the Weibull probability distribution of system elements' delay times *u*

optimum *T* found for a parallel system with decreasing hazard rate of initial time and delay time differs meaningfully from the results found for systems with the increasing hazard rate (Fig. 16).

On the other hand, if there is a justification to state that the hazard rates of times *u* and *h* are strictly increasing, the cost consequences of imperfect estimation of probability distribution parameters should not be severe. This situation is mainly connected with the wide flat region of "similarly good" *T* intervals (e.g. Fig. 2).

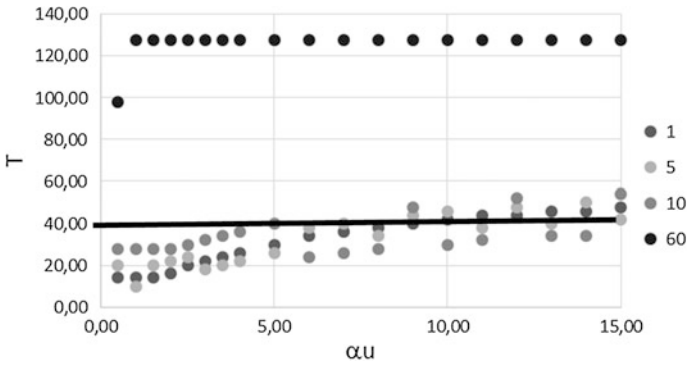


Fig. 13 The *i*th (i = 1, 5, 10, 60) best interval between inspections in 3-*out-of*-3 (parallel) system for various shape parameters "au" in the Weibull probability distribution of system elements' initial times u

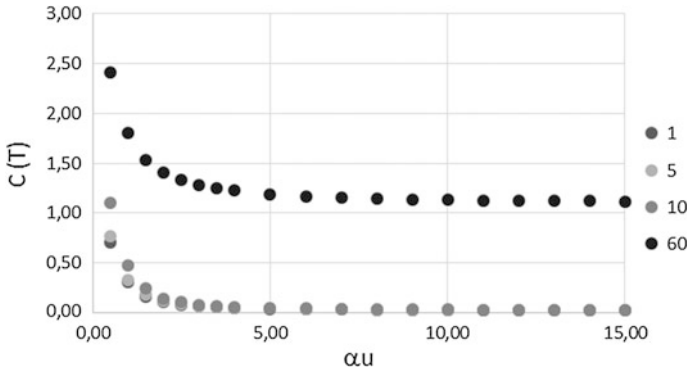


Fig. 14 The total expected cost of the BIP policy obtained for the *i*th (i = 1, 5, 10, 60) best interval between inspections in 3-*out-of*-3 (series) system for various shape parameters "au" in the Weibull probability distribution of system elements' initial times u

The above mentioned fact may be of utmost importance from the practical point of view, when there is not enough operational and maintenance data to use complicated analytical models requiring precise information on probability distributions forms and their parameters. In such cases, on the base of expert estimation of the expected initial and delay times, the parameters may be assessed and the proposed model may be used to determine the profitable (not necessary the best) time interval between inspections without a great risk of severe cost consequences. An exemplary interval T that yields cost results close to the best found solution, is marked by the black horizontal line in Fig. 13 and may be determined independently on precise parameters of probability distribution of initial time u.

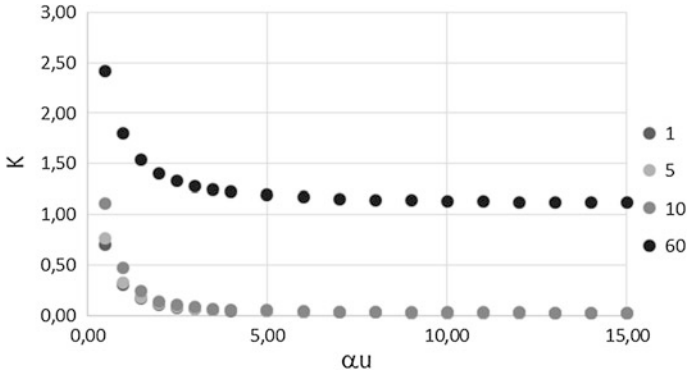


Fig. 15 *K* coefficient corresponding to the *i*th (i = 1, 5, 10, 60) best interval between inspections in 3-*out-of*-3 (parallel) system for various shape parameters "au" in the Weibull probability distribution of system elements' initial times *u*

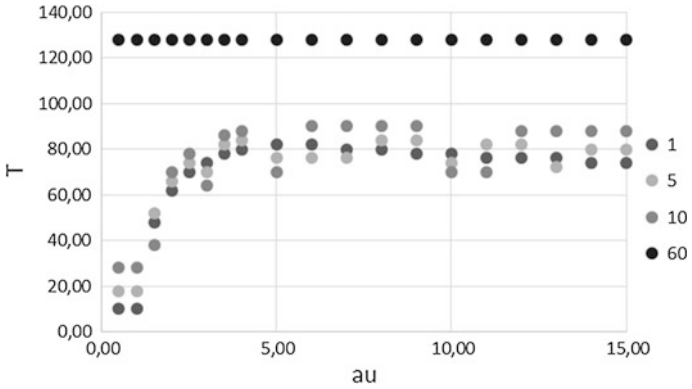


Fig. 16 The *i*th (i = 1, 5, 10, 60) best interval between inspections in 3-*out-of*-3 (parallel) system for various shape parameters "au" in the Weibull probability distribution of system elements' initial times *u*

5 Summary

In the paper, authors focus on the problem of delay-time model parameters estimation. Even the best maintenance model cannot give the right advice for maintenance managers, if there is no possibility to obtain and estimate the necessary operational and maintenance data. The basic reliability parameters that have to be specified in such systems are: an estimation of system components' time to signal of a future failure and some delay time characteristics. These two times, together with other data (e.g. a system reliability structure, unit costs, etc.) give the base to optimize the time between inspections that may provide the best economical and/or

availability results. In the situation, when are sufficient and reliable maintenance and operational data, the known parameter estimation methods may be applied. However, maintenance engineers usually are able to assess (more or less precisely) only the expected delay time and sometimes may enrich it with the standard deviation estimator. Thus, authors investigate what cost consequences may result from imprecise estimation of initial and delay time probability distributions.

On the basis of given research, it is possible to answer the questions asked in the first subsection. The most important parameters of initial and delay times probability distribution are: their expected values of initial and delay time as well as "a shape" of initial time hazard rate. Those three are usually enough to estimate in order to determine near-optimum inspection interval for a multi-unit system with *k-out-of-n* reliability structure. They should be estimated as precisely as it is possible. If the times and hazard rate are evaluated on the base of some objective data one can apply the proposed method, which allows for effortless finding of "a good" solution for most of systems built of components with increasing hazard rates of times u and h. Nonetheless if the hazard rates of the considered times are close to constant, the estimation of probability characteristics should be made in details.

To sum up the paper, the research results may be used by engineers to define what kind of operational and maintenance information should be gathered in order to define proper inspection policy for technical systems. First, there are given some guidelines for the best length of time period T estimation—the practical cost coefficient K is defined and investigated. It indicates the period T for which the probability of a system failure is still low whereas probability of elements' preventive replacement is high—what is intuitive. In the second step, there was performed testing process of the developed cost model assuming various forms of probability distributions describing the basic variables—initial time u and delay time h of system elements. The carried out research gives the answer, when the exact estimation of probability distributions of random variables u and h is necessary in order to obtain the reasonable economical results in the maintenance decision-making process. It also gives some tips when such a risk of severe cost consequences may be avoided. As one might expect, in the situation when hazard rates of times u and h are strictly increasing (mostly occurred in practice), the cost consequences of imperfect estimation of probability distribution parameters should not be severe. Following this, the presented may contribute to a first step of considerations about development of decision support system focused on maintenance decision-making processes.

References

1. Akbarov, A., Christer, A.H., Wang, W.: Problem identification in maintenance modeling: a case study. Int. J. Prod. Res. **46**(4), 1031–1046 (2008)
2. Babiarz, B., Chudy-Laskowska, K.: Forecasting of failures in district heating systems. Eng. Fail. Anal. **56**, 384–395 (2015)

3. Christer, A.H.: Modelling inspection policies for building maintenance. J. Opl. Res. Soc. **33**, 723–732 (1982)
4. Christer, A.H.: Developments in delay time analysis for modelling plant maintenance. J. Opl. Res. Soc. **50**, 1120–1137 (1999)
5. Christer, A.H.: A review of delay time analysis for modelling plant maintenance. In: Osaki, S. (ed.) Stochastic Models in Reliability and Maintenance. Springer (2002)
6. Christer, A.H., Redmond, D.F.: A recent mathematical development in maintenance theory. IMA J. Math. Appl. Bus. Ind. **2**, 97–108 (1990)
7. Christer, A.H., Redmond, D.F.: Revising models of maintenance and inspection. Int. J. Prod. Econ. **24**, 227–234 (1992)
8. Christer, A.H., Waller, W.M.: An operational research approach to planned maintenance: modelling P.M. for a vehicle fleet. J. Opl. Res. Soc. **35**(11), 967–984 (1984)
9. Christer, A.H., Waller, W.M.: Reducing production downtime using delay-time analysis. J. Opl. Res. Soc. **35**(6), 499–512 (1984)
10. Christer, A.H., Waller, W.M.: Delay time models of industrial inspection maintenance problems. J. Opl. Res. Soc. **35**(5), 401–406 (1984)
11. Christer, A.H., Whitelaw, J.: An operational research approach to breakdown maintenance: problem recognition. J. Opl. Res. Soc. **34**(11), 1041–1052 (1983)
12. Christer, A.H., Wang, W., Sharp, J., Baker, R.: A case study of modelling preventive maintenance of production plant using subjective data. J. Opl. Res. Soc. **49**, 210–219 (1998)
13. Giel, R., Młyńczak, M., Plewa, M.: Logistic support model for the sorting process of selectively collected municipal waste. In: Theory and Engineering of Complex Systems and Dependability: Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, June 29–July 3, 2015, Brunów, Poland, pp. 369–380. Springer (2015)
14. Jodejko-Pietruczuk, A., Werbińska-Wojciechowska, S.: Expected maintenance costs model for time-delayed technical systems in various reliability structures. In: Proceedings of Probabilistic Safety Assessment and Management, PSAM 12: Honolulu, Hawaii, USA, 22–27 June 2014 (IAPSAM), pp. 1–8 (2014)
15. Jodejko-Pietruczuk, A., Werbińska-Wojciechowska, S.: Model of expected maintenance costs for multi-unit technical objects with time delay (in Polish). In: Siergiejczyk, M. (ed.) Maintenance Problems of Technical Systems, pp. 79–92. Publ. House of Warsaw University of Technology, Warsaw (2014)
16. Jodejko-Pietruczuk, A., Młyńczak, M., Zając, M.: Assessment of economical lifetime of heavy-duty machines: case study. In: Briš, R., Guedes Soares, C., Martorell, S. (eds.) Reliability, Risk and Safety: Theory and Applications, vol. 1, pp. 531–534. Taylor & Francis, London (2010)
17. Jones, B., Jenkinson, I., Wang, J.: Methodology of using delay-time analysis for a manufacturing industry. Rel. Eng. & Sys. Safety **94**, 111–124 (2009)
18. Kierzkowski, A., Kisiel, T.: Functional readiness of the check-in desk system at an airport. Theory and Engineering of Complex Systems and Dependability: Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, June 29–July 3, 2015, Brunów, Poland, pp. 223–233. Springer (2015). DOI:10.1007/978-3-319-19216-1_21
19. Kierzkowski, A., Kisiel, T.: Functional readiness of the security control system at an airport with single-report streams. In: Theory and Engineering of Complex Systems and Dependability: Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, June 29–July 3, 2015, Brunów, Poland, pp. 211–221. Springer (2015). DOI:10.1007/978-3-319-19216-1_20
20. Lubkowski, P., Laskowski, D.: Selected issues of reliable identification of object in transport systems using video monitoring services. In: Mikulski, J. (ed.) Telematics—support for transport. Book Series: Communications in Computer and Information Science, vol. 471, pp. 59–68 (2014)

21. Nowakowski, T., Werbińska-Wojciechowska, S.: Developments of time dependencies modeling concepts. Advances in safety, reliability and risk management. In: Proceedings of the European Safety and Reliability Conference, ESREL 2011, Troyes, France, 18–22 September 2011, pp. 832–838. CRC Press/Balkema, Leiden (2011)
22. Pokoradi, L.: Sensitivity analysis of reliability of systems with complex interconnections. J. Loss Prev. Process Ind. **32**, 436–442 (2014)
23. Restel, F.J.: Measures of reliability and safety of rail transportation system. In: Advances in Safety, Reliability and Risk Management—Proceedings of the European Safety and Reliability Conference, ESREL 2011, pp. 2714–2719 (2012)
24. Tang, Y., Jing, J.J., Yang, Y., Xie, C.: Parameter estimation of a delay time model of wearing parts based on objective data. Math. Probl. Eng. Article ID 419280, 1–8 (2015)
25. Van Noortwijk, T.M., Dekker, R., Cooke, R.M., Mazzuchi, T.A.: Expert judgement in maintenance optimization. IEEE Trans. Reliab. **41**, 427–432 (1992)
26. Wang, W.: Subjective estimation of delay-time distribution in maintenance modelling. Eur. J. Oper. Res. **99**, 511–529 (1997)
27. Wang, W.: Modeling planned maintenance with non-homogeneous defect arrivals and variable probability of defect identification. Ekspl Niez - Maint Reliab **2**, 73–78 (2010)
28. Wang, W.: An inspection model based on a three-stage failure process. Reliab. Eng. Syst. Safety **96**, 838–848 (2011)
29. Wang, W.: An overview of the recent advances in delay-time-based maintenance modelling. Reliab. Eng. Syst. Safety **106**, 165–178 (2012)
30. Wang, W., Jia, X.: An empirical Bayesian based approach to delay time inspection model parameters estimation using both subjective and objective data. Qual. Reliab. Eng. Int. **23**, 95–105 (2007)
31. Wang, W., Zhang, W.: Early defect identification: application of statistical process control methods. J. Qual. Mainten. Eng. **14**(3), 225–236 (2008)
32. Wang, L., Hu, H., Wang, Y., Wu, W., He, P.: The availability model and parameters estimation method for the delay time model with imperfect maintenance at inspection. Appl. Math. Model. **35**, 2855–2863 (2011)
33. Wen-yuan, L.V., Wang, W.: Modelling preventive maintenance of production plant given estimated PM data and actual failure times. In: Proceedings of International Conference on Management Science and Engineering ICMSE'06, Lille (2006)
34. Zajac, M., Swieboda, J.: Process hazard analysis of the selected process in intermodal transport. In: 2015 International Conference on Military Technologies (ICMT), pp. 1–7. IEEE (2015)

Effectiveness of Redundancy in Communication Network of Air Traffic Management System

Igor Kabashkin

Abstract Air Traffic Management (ATM) systems represent essential infrastructure that is critical for flight safety. Communication is a key element in the present ATM system. Communication between air traffic controllers and pilots remains a vital part of air traffic control operations, and communication problems can result in hazardous situations. The modern ATM system has independent direct communication channels (CC) for each controllers operating at different radio frequencies. Currently, the main method of improving the reliability of controller's CC is duplication of equipment to provide communications on each frequency of interaction ground-to-air channel. Unfortunately the economic efficiency of used fault-tolerance approach is low. In the paper the reliability of selected communication channels with common set of standby radio station in the system with periodical sessions of communications for real conditions of ATM is discussed. Mathematical model of the channel reliability is developed. Comparative analysis of redundancy effectiveness for developed and used structure of communication network is performed.

Keywords Reliability · Redundancy · Air traffic management · Controller · Communication network

1 Introduction

Air Traffic Management (ATM) systems represent essential infrastructure that is critical for flight safety. Communication is a key element in the present ATM system. Communication between air traffic controllers and pilots remains a vital part of air traffic control operations, and communication problems can result in hazardous situations. Analysis of aviation accidents has identified that a breakdown

I. Kabashkin (✉)
Transport and Telecommunication Institute, Lomonosova 1, Riga 1019, Latvia
e-mail: kiv@tsi.lv

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_22

in effective human communication has been a causal or contributing factor in the majority of accidents [1].

There are different types of air traffic controllers:

- Tower controllers direct the movement of vehicles on runways and taxiways. Most work from control towers, watching the traffic they control.
- Approach and departure controllers ensure that aircraft traveling within an airport's airspace maintain minimum separation for safety.
- En route controllers monitor aircraft once they leave an airport's airspace.

The modern ATM system has independent direct communication channels (CC) for each controllers operating at different radio frequencies f_i, $i = \overline{1,m}$, where m is number of CC. The amount of the CC is determined by the structure of ATM in the area of a specific airport and provides independent interaction with the aircrafts for all controllers. Technical support of controller-pilot communication carried out by means of radio stations (RS). Interoperability of technical means and controllers in ATM communication network is provided by voice communications system (VCS) which is a state-of-art solution for air traffic control communication. The modern approach to system design focuses on providing high-availability solutions that are based on reliable equipment and on redundancy strategies tailored to customers' needs and requirements.

Currently, the main method of improving the reliability of controller's CC is to duplicate equipment to provide communications on each frequency of interaction ground-to-air channel (Fig. 1). Each of the m controllers communicates with

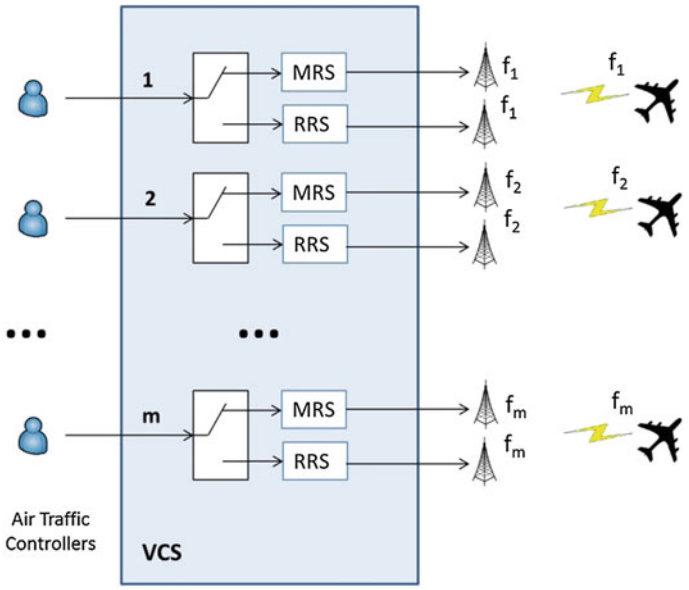


Fig. 1 Redundant communication network of ATM system

aircraft using the main radio station (MRS) as basic hardware. After the failure of MRS, he switches into a work with the backup (redundant radio station—RRS).

Unfortunately the economic efficiency of used fault-tolerance approach is low. In the paper another redundant method for communication network of ATM system is discussed.

The rest of this paper is organized as follows. In Sect. 2 some important works in the area of reliability with redundancy are reviewed. In Sect. 3 the main definitions and assumptions are presented and a models of ATM communication network reliability with common set of standby radio stations for reservation of different ATM communication channels are proposed. In Sect. 4 the conclusions are presented.

2 Related Works

The reliability of duplicated channels is well studied in the literature [2, 3].

One of the methods to increase efficiency of redundancy in the structures with identical elements is the allocation of the common group of reserve elements. The *k-out-of-n* system structure is a very popular type of redundancy in fault tolerant systems. The term *k-out-of-n* system is used to indicate an *n*-component system that works (or is "good") if and only if at least *k* of the *n* components work (or are good). This system is called a *k-out-of-n:G* system. The works [4–6] provide improved versions of the method for reliability evaluation of the *k-out-of-n:G* system. Liu [7] provides an analysis of the *k-out-of-n:G* system with components whose lifetime distributions are not necessarily exponential.

An *n*-component system that fails if and only if at least *k* of the *n* components fail is called a *k-out-of-n:F* system [8]. The term *k-out-of-n* system is often used to indicate either a G system or an F system or both. The *k-out-of-n* system structure is a very popular type of redundancy in fault-tolerant systems. It finds wide applications in telecommunication systems [9, 10]. This model can be used for analyse of reliability of ATM communication network with *k* controllers and *n* RS provided availability of CC.

In real conditions it is important to know not the reliability of communication network at whole but each selected CC for controller. The channel reliability problem in standby system consisting of independent elements with some units used as a universal component standby pool is investigated in [11].

In this paper we investigate the reliability of selected communication channel with common set of standby radio station in the system with periodical sessions of communications for real conditions of ATM communication network.

3 Model Formulation and Solution

The following symbols have been used to develop equations for the models:

λ Failure Rate for MRS and RRS
μ Repair Rate for MRS and RRS
m Number of communication channels and number of MRS
n Number of RRS in common set of redundant radio stations
A Channel Availability
A_0 Required Availability of the CC
A_d Availability of the system with duplicate RS
l Number of repair bodies
t_s Mean time of failed RS switching on a reserve one and the frequency tuning in the communication channels
$\upsilon = 1/t_s$ Parameter of exponential distribution of t_s
T_c Periodicity of communication demands with parameter of Poisson's flow $\varphi = 1/T_c$

In this paper we investigate a repairable redundant communication network of ATM system with $N = m + n$ radio stations, m of which are MRS and n radio stations are used as a universal component standby pool (Fig. 2). All switching operations and the restructuring of reserve radio frequencies are carried out by

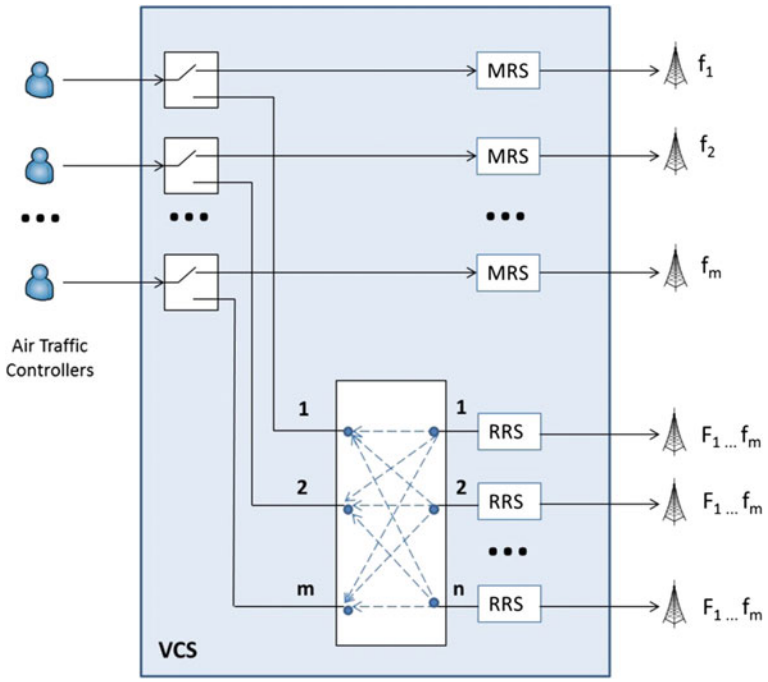


Fig. 2 Communication network of ATM system with common set of standby elements

VCS. The channel reliability for each controller (reliability of selected communication channel) must satisfy the following requirements

$$A \geq A_0 \tag{1}$$

The behaviour of the examined system is described by the Markov Chain state transition diagram (Fig. 3), where: H_i—state with *i* failed RS, but in the selected channel there is a workable RS; H_{iI}—state with *i + 1* failed RS, in the selected channel there is no a workable RS; H_{Ti}—state with *i* failed RS, *i − 1* RS are under repaired, failure of *i-th* RS is not detected, but a demand on communication is in this channel, in the selected channel there is a workable RS; $H_{\pi i}$—state with *i* failed RS, *i − 1* RS are under repaired, the RRS is beginning switch to frequency of failed MRS, in the selected channel there is a workable RS; H_{ITi}—state with *i* failed RS, one of failed RS is in the selected channel, this failure is not detected, a demand on communication is in this channel, the rest *i − 1* failed RS are under repaired; $H_{I\pi i}$—

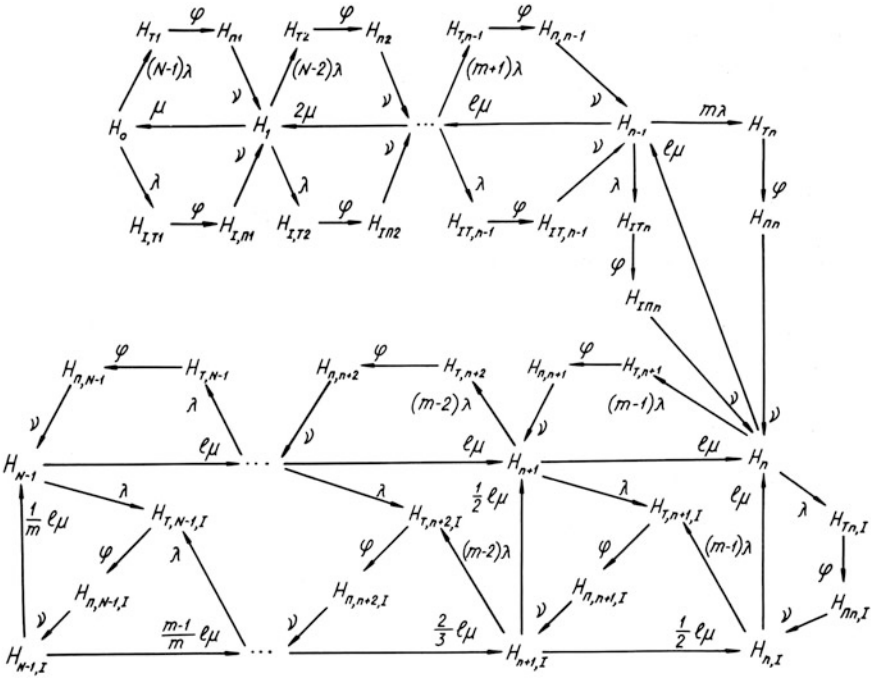


Fig. 3 Markov Chain state transition diagram

state with i failed RS, one of them is in selected channel, the RRS is beginning switch to frequency of failed RS from selected channel, $i - 1$ RS are under repaired.

For active backup mode of redundant elements and for the system with $1 \le l \le n$ number of repair bodies the Markov Chain state transition diagram of the system is shown at the Fig. 3. On the base of this diagram the system of Chapman–Kolmogorov's equations can be writing in accordance with the general rules [12].

By solving the resulting system of equations, we can obtain an expression for availability of selected communication channel:

$$A = 1 - \sum_{\forall i,j} P_{ijI} = \frac{a_1 + a_2}{a_1 + a_2 + a_3},$$

where

$$a_1 = \sum_{i=0}^{l} \binom{i}{N} \gamma^i \left[1 + \lambda\left(\frac{1}{\varphi} + \frac{1}{\upsilon}\right)(N - i - 1)\right] + \frac{N! l^l}{l!} \sum_{i=l+1}^{n-1} \frac{\omega^i}{(N-i)!} \left[1 + \lambda\left(\frac{1}{\varphi} + \frac{1}{\upsilon}\right)(N - i - 1)\right],$$

$$a_2 = \frac{N! l^l \omega^n}{ml!} \sum_{i=0}^{m-1} \frac{\omega^i}{(m-i-1)!} \left[1 + \lambda\left(\frac{1}{\varphi} + \frac{1}{\upsilon}\right)(m - i - 1)\right],$$

$$a_3 = \frac{N! l^l \omega^n}{ml!} \sum_{i=0}^{m-1} \frac{(i+1)\omega^{i+1}}{(m-i-1)!} \left[1 + \lambda\left(\frac{1}{\varphi} + \frac{1}{\upsilon}\right)\left(m - i - 1 + \frac{1}{i+1}\right)\right] + \lambda\left(\frac{1}{\varphi} + \frac{1}{\upsilon}\right)\left[\sum_{i=0}^{l} \binom{i}{N} \gamma^i + \frac{N! l^l}{l!} \sum_{i=l+1}^{n-1} \frac{\omega^i}{(N-i)!}\right],$$

$$\gamma = \frac{\lambda}{\mu}, \omega = \frac{\gamma}{l}.$$

The communication network of ATM system is highly reliable system. For such systems condition $N\omega \ll 1$ is satisfied. In this case the equation for availability A with sufficient accuracy for practical purposes can be written as

$$A = 1 - \lambda\left(\frac{1}{\varphi} + \frac{1}{\upsilon}\right)a, \tag{2}$$

where

$$a = \sum_{i=0}^{l} \binom{i}{N} \gamma^i + \frac{N! l^l}{l!} \sum_{i=l+1}^{n} \frac{\omega^i}{(N-i)!}.$$

Limitation on switch time $t_s = 1/\upsilon$ during reservation process can be determined by substituting the expression (2) in the expression (1):

$$t_s \leq \frac{1 - A_0}{a\lambda} - \frac{1}{\varphi} \tag{3}$$

The similar analysis for the system with $n \leq l \leq m$ number of repair bodies is shown that if condition $N\omega \ll 1$ is satisfied the limitation on switch time $t_s = 1/\upsilon$ during reservation process can be determined by the same expression (3), where

$$a = \sum_{i=0}^{n} \binom{i}{N} \gamma^i$$

3.1 Numerical Example

Let's evaluate the availability of selected channel in redundant communication network of ATM system with operation sessions in the random moments of time.

It is possible to evaluate of decreasing of the reliability in the proposed model of redundancy with common set of standby radio stations in comparison with the standard model of duplication RS in communication channel with the help of the reliability degradation factor

$$V = (1 - A_d)/(1 - A),$$

At the Fig. 4 the reliability degradation factor

$$V = \frac{1 - A_d}{1 - A}$$

shown as function of number m of communication channels with different number n of standby radio stations in common redundant set of RS for Mean Time Between

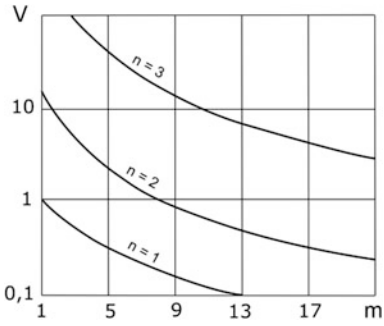


Fig. 4 The reliability degradation factor

Failures of each RS $MTBF = 1/\lambda = 3000$ h, 1 = 1 and $T_c = 30$ min. The value of A is determined in accordance with the expression (2), equation for the A_d availability of the system with duplicate RS in each channel was determined in [2].

Analysis of the curves at Fig. 4 shows that for communication network with small number of channels ($m \leq 8$) in proposed model of redundancy channel reliability is not less than in the model with the duplication of RS and is achieved by using only two redundant stations. For communication network with $m \geq 9$ channels in proposed model of redundancy only three RRS are needed for the same channel reliability. For example, the economic effectiveness for communication system with $m = 8$ channels will be six free radio stations, and for $m = 21$ channels this effect will be 18 RRS.

4 Conclusions

Mathematical model of the channel reliability of the communication network of Air Traffic Management system in the real conditions of operation is developed. It is shown that there is a finite switching time on a reserve, for which the reliability of the proposed model of reservation will be higher than using duplication of radio stations in the channels for any number of common set of redundant elements $n \geq 2$.

Expressions for channel availability of the ATM communication network with common set of redundant radio stations for all controllers are developed.

It is shown, that for real ATM communication network only $n = 2$ backup radio stations for network with $m \leq 8$ channels and only $n = 3$ backup radio stations for network with $m \geq 9$ channels are needed to provide the channel reliability not less than in the model with duplication of communication equipment in each channel.

High economic efficiency of the proposed method of backup is shown.

Acknowledgments This work was supported by Latvian state research programme project "The Next Generation of Information and Communication Technologies (Next IT)" (2014–2017).

References

1. Wiegmann, D., Shappell, S.: Human error perspectives in aviation. Int. J. Aviat. Psychol. **11** (4), 341–357 (2001)
2. Ireson, W., Coombs, C., Moss, R.: Handbook of Reliability Engineering and Management 2/E, 2nd edn, 816 p. McGraw-Hill Education (1995)
3. Modarres, M., Kaminskiy, M., Krivtsov, V.: Reliability Engineering and Risk Analysis: A Practical Guide, 2nd edn. (Quality and Reliability), 470 p. CRC Press (2009)
4. Barlow, R., Heidtmann, K.: On the reliability computation of a k-out-of-n system. Microelectron. Reliab. Elsevier **33**(2), 267–269 (1993)
5. Misra, K.:. Handbook of Performability Engineering, 1315 p. Springer (2008)
6. McGrady, P.: The availability of a k-out-of-n:G network. IEEE Trans. Reliab. R-34(5), 451–452 (Dec. 1985)

7. Liu, H.: Reliability of a load-sharing k-out-of-n: G system: non-iid components with arbitrary distributions. IEEE Trans. Reliab. **47**(3), 279–284 (1998)
8. Rushdi, A.: A switching-algebraic analysis of consecutive-k-out-of-n: F systems. Microelectron. Reliab. Elsevier **27**(1), 171–174 (1987)
9. Ayers, M.: Telecommunications System Reliability Engineering, Theory, and Practice, 256 p. Wiley-IEEE Press (2012)
10. Chatwattanasiri, N., Coit, D., Wattanapongsakorn, N., Sooktip, T.: Dynamic k-out-of-n system reliability for redundant local area networks. In: 2012 9th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), pp. 1–4, 16–18 (May 2012)
11. Kozlov, B., Ushakov, I.: Reliability Handbook (International series in decision processes), 416 p. Holt, Rinehart & Winston of Canada Ltd (1970)
12. Rubino, G., Sericola, B.: Markov Chains and Dependability Theory, 284 p. Cambridge University Press (2014)

Resilience Assurance for Software-Based Space Systems with Online Patching: Two Cases

Vyacheslav Kharchenko, Yuriy Ponochovnyi, Artem Boyarchuk and Eugene Brezhnev

Abstract The paper discusses the problems of resilient software engineering for unmanned software-based space systems. Resilience is achieved by online patching of software upon emergence of defects providing a stable link to the ground control center. Based on the specifics of satellite orbits it offers two case models: a multifragment one—for systems with a continuous link from geostationary orbits; a multiphase one—for recurrent link from elliptic orbits. The results of the modeling offer the possibility to plan the values of the software initial failure rate and the period of preventive tests that would ensure required reliability and availability.

Keywords Reliability and availability of software-based space systems · Online patching · Markov's multifragment and multiphase models

1 Introduction

The rocket and space industry is one of the most important sectors of the world economy. The opportunities offered by rocket and space technology, its reliability and safety depend heavily on the characteristics of Software-Based Space Systems (SBSpS), their hardware and software quality. This paper considers unmanned space crafts (SC), which do not provide for the restoration of the SBSpS hardware.

V. Kharchenko (✉) · A. Boyarchuk
National Aerospace University KhAI, Kharkiv, Ukraine
e-mail: V.Kharchenko@khai.edu

A. Boyarchuk
e-mail: a.boyarchuk@mail.ru

V. Kharchenko · E. Brezhnev
Research and Production Company Radiy, Kirovograd, Ukraine
e-mail: e.brezhnev@csis.org.ua

Y. Ponochovnyi
Poltava National Technical University, Poltava, Ukraine
e-mail: pnch1@rambler.ru

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_23

In contrast to hardware, software of unmanned SC's can be restored and modified provided stable link to the ground control center [1]. This ensures software resilience, that is its resistance to changes in the requirements and conditions of the external environment as well as to occurrence of unspecified defects [2].

Development of software is an expensive process, with the major costs coming not from creation of the code, but from its qualification testing. According to standards accepted in the space industry the term "qualification testing" [3] is used to denote "the whole set of actions for verification and validation of critical software". The high cost of testing is due to the need to simulate outer space environment conditions in terrestrial conditions. The use of resilient upgradable software in SC's provides for a more flexible distribution of verification stages and elimination of detected defects during SC operation. The choice of software architecture must first be validated using mathematical models, which should take into account software modifications and re-engineering in the process of operation a space complex.

Existing models of systems with variable parameters use simulation methods [4], Bayesian analysis [5] and the most preferred method of Markov's and semi-Markov's processes [6]. In [7], a system approach is developed to the construction of multifragment models, but it does not provide modeling of procedures related to software online patching.

This research is aimed at the development and analysis of SBSpS availability models, which architecture allows for periodic or online patching of the code of software functions. The paper is structured as follows: the second section is devoted to the availability assessment of onboard software-based space systems and researched space computer system (Sect. 2.1); description of initial model SBSpS, which includes two duplicated hardware channels (Sect. 2.2). The third section offers an approach to developing availability model of geostationary software-based space systems with online patching and continuous link. The fourth chapter contains the detailed model presentation and research of software-based space systems with elliptical orbit, online patching and recurrent link. In the last chapter the conclusions are made, directions for future works are outlined.

2 Availability Assessment of Onboard Software-Based Space Systems

2.1 Researched Space Computer System

The need for software verification and updates of SBSpS requires a hardware architecture as in Fig. 1 [8].

The ground system sends the initiation commands to start verification procedures, which are processed by a special input data processing and decision unit (cancel module).

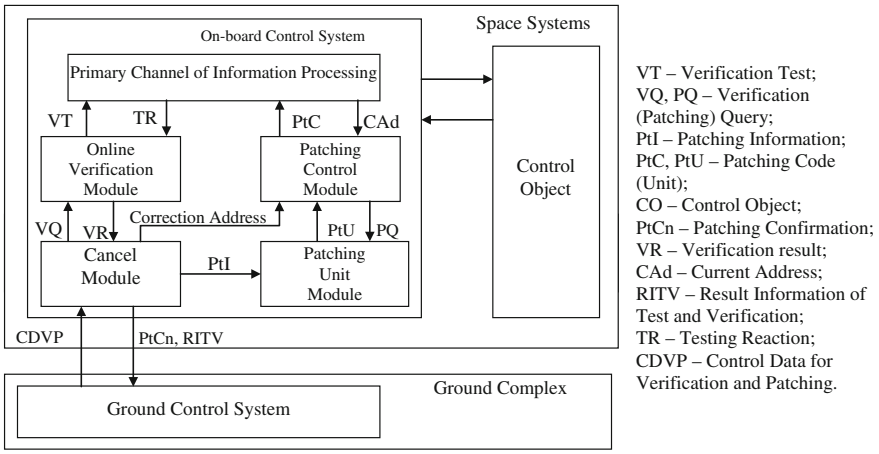


Fig. 1 Hardware SBSpS with online and periodic software patching

The sets of verification tests are stored in online verification module. In case a defective section of the code is found, information about it is sent to the ground control center, where a decision is made to eliminate the detected defect(s) and the software code replacement section is generated, which is transmitted via the cancellation block to the block of patching modules. In the patching control module, based on information on the current address of the element being executed, it is then followed by control takeover from the main information processing channel and replacement of the defective section by patched software code from the block of patching modules.

The software architecture must include the following functional blocks (Fig. 2).

Automatic spaceship can be used in different types of orbits (elliptic and geostationary). Provided continuous link (geostationary satellites) it will be more appropriate to use the strategy of online patching upon detection of a software defect. Satellites with elliptical orbits are characterized by short periodic intervals of

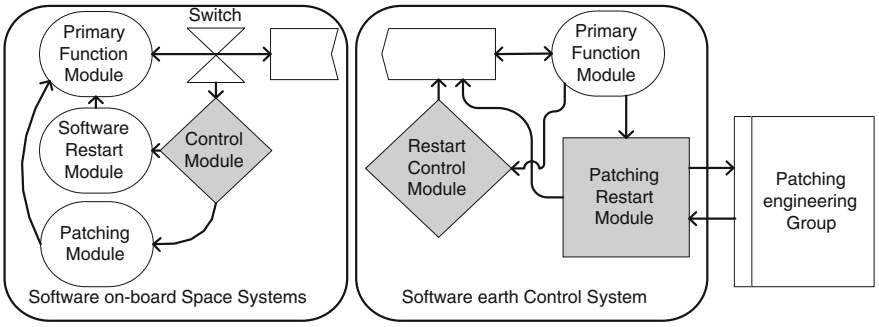


Fig. 2 Software architecture for online or periodic software patching

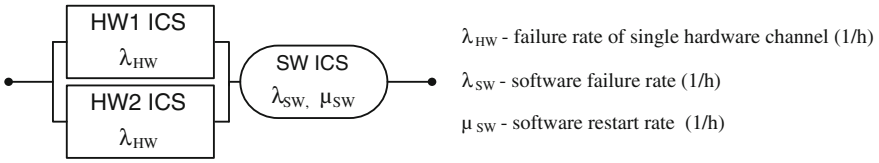


Fig. 3 Reliability block diagram of two-channel one-version SBSpS

stable link. In this case a more appropriate strategy will be periodic software code patching during time intervals that are multiples of the period of stable link spans.

2.2 Initial Model

In this paper we examine unmanned spacecraft with the most common architecture of SBSpS, which includes two duplicated channels; each of them operates with the same software version (Fig. 3). It is foreseen to restore the functional state of SBSpS by spacecraft crew [9]. Software of such spacecraft allows such modifications and software reengineering during spacecraft maintenance period.

The probability of failure-free operation of the hardware component of such SBSpS is a decreasing function $R_{HW}(t)$. If one is to consider a hypothetical situation when SBSpS software (and hardware) do not allow for recovery, the probability of failure-free operation of the SBSpS is defined by $R_{HW+SW}(t)$.

$$R_{HW}(t) = 2e^{-\lambda_{HW}\cdot t} - e^{-2\lambda_{HW}\cdot t};\ R_{HW+SW}(t) = \left(2e^{-\lambda_{HW}\cdot t} - e^{-2\lambda_{HW}\cdot t}\right)\cdot e^{-\lambda_{SW}\cdot t} \quad (1)$$

Introduce another assumption. In order to ensure trouble-free execution of program functions that allow for patching of software defects, the hardware component shall provide the probability of failure-free operation of at least 0.999 for 10 years (t = 87,658 ≈ 90,000 h) of spacecraft operation. The SBSpS complying with this requirement will have $\lambda_{HW} = 3e{-}7\ h^{-1}$. Moreover, at such value of λ_{HW}, a reliability of 0.999 is maintained for 993,243 h (11.3 years).

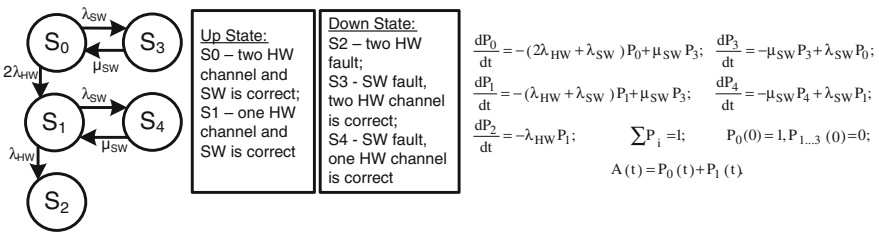


Fig. 4 Markovian graph of SBSpS with software restart and Kolmogorov-Chapman system

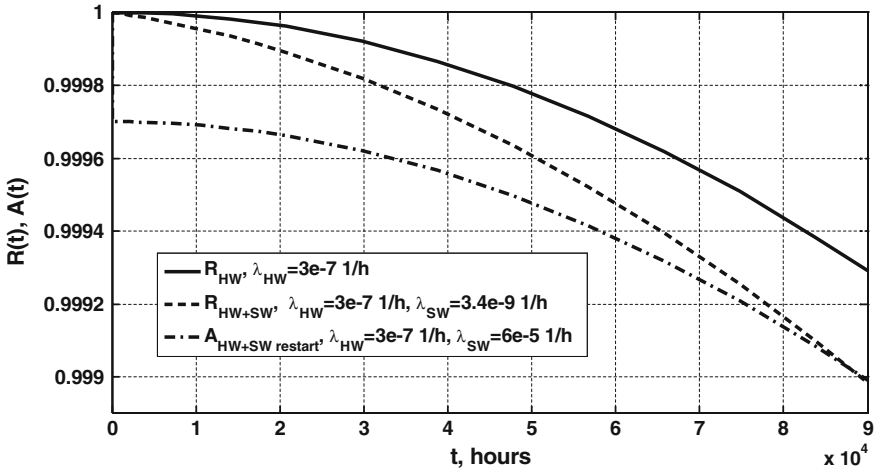


Fig. 5 Reliability of hardware and hardware-software two-channel SBSpS

Then the reliability requirement of 0.999 for 10 years of operation is attainable at λ_{SW} = 3.4e−9 h^{-1}. It should be noted that for software such value of λ_{SW} is practically unachievable.

A simple Markovian model that provides for software restart upon failure is presented in Fig. 4. The required availability value of 0.999 is achieved (provided that software restart time $T_{RESTART}$ = 5 h) at the software failure rate of λ_{SW} = 6e−5 h^{-1} (Fig. 5).

3 Case 1: Availability Model of Geostationary Software-Based Space Systems with Online Patching and Continuous Link

3.1 Description of Availability Model

In order to build the SBSpS availability model the following assumptions were made:

- at any given moment of time the SBSpS may be either in an operable or non-operable mode, and the flows of events switching the system from one functional state to another are simple;
- the SBSpS is recovered after a failure due to a software defect by software restart;
- elimination of software defects occurs after they have manifested themselves in the form of system failures with the detection probability D;

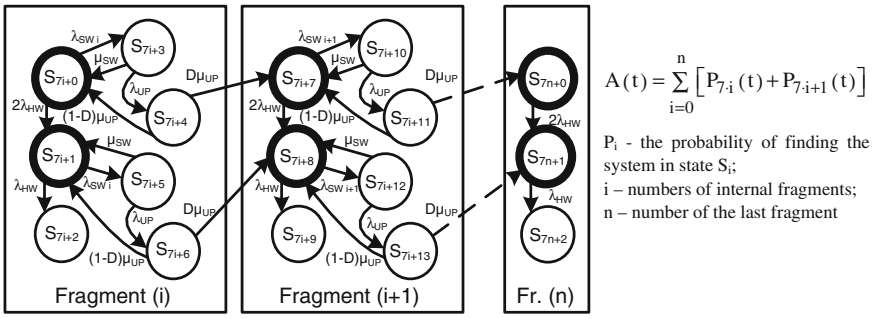


Fig. 6 Markovian graph of SBSpS functioning with software patches and availability function

- in order to eliminate a software defect the SBSpS switches to the patching mode and this mode is non-operable;
- patching of software defects does not cause new defects;
- all undetected failures can be completely eliminated.

Considering the assumptions made, Markovian analysis is adopted as the research method, and changes in the failure rate λ_{SW} are recorded using regular multifragment models [7, 9]. Therefore, the model, which graph is presented in Fig. 6 was selected as the reference model.

The SBSpS operation process is as follows. At start time the system implements all the specified functions and is in state S_0 (for the initial fragment i = 0). During operation hardware defects show up resulting in the system sequentially changing to state S_1 (failure of one of the hardware channels, with the system being in operable condition), S_2 (simultaneous failure of two hardware channels, with the system being in non-operable condition). After some time interval the system fails due to a software defect and its changes to state S_3 from state S_0 (or to state S_5 from state S_1). Upon emergence of the software defect it is, of course, detected and eliminated with probability D, and as a result after recovery the system passes to the next fragment (to state S_{7i+7} from state S_4 or to state S_{7i+8} from state S_6), which is characterized by a new parameter of $\lambda_{SW\ i+1}$. With the probability of (1–D) the defect is not eliminated, this is simulated by return to states S_0 and S_1. In the last fragment of the model the software defects are eliminated, and the systems suffers hardware failures only.

3.2 *Justification of Model Input Parameters*

The objective of the research is the analysis of variation of availability function. Input parameters' values used for modeling were obtained from statistical data on exploitation of SBSpS [8, 9] in Table 1.

Table 1 Values of input parameters

Symbol	Illustration	Value	Unit
λ_{HW}	Failure rate of single hardware channel	3e−7	1/h
λ_{SW0}	Initial software failure rate	[1e−4… 1e−3]	1/h
μ_{SW}	System recovery rate after occurrence of software fault	0.2	1/h
$\lambda_{SW\ k}$	Software failure rate after fixing of all faults is zero	0	1/h
μ_{UP}	Rate, matches to average patch installation time including transmission of the software code to the spacecraft is 5 h	0.2	1/h
λ_{UP}	Rate, matches to average time of localization of the software defect and development of a patch including collection of data for testing and its transmission from the spacecraft to the ground control center is 10 h.	0.1	1/h
D	Probability of complete elimination of the software defect by the patch	0.95	
$\Delta\lambda_{SW}$	Additional limitation presents elimination of 10 undetected software defects and uniformity of load for their localization and elimination	λ_{SW0}/ 10	1/h
T	Time interval for analysis of availability function behavior	90,000	hours

The task of analysis of the model is limited to finding the value of software failure rate $\lambda_{SW\ 0}$, at which the requirement to ensure the system availability function of not less than 0.998 for 2 years (20,000 h) in operation is met.

The results of calculations are presented as the graphical dependency of availability function on system functioning time in Fig. 7. The result of the modeling has shown that in this model the availability level of 0.998 for the first 2 years of operation is achieved at the software failure rate λ_{SW} = 5e−4 (1/h).

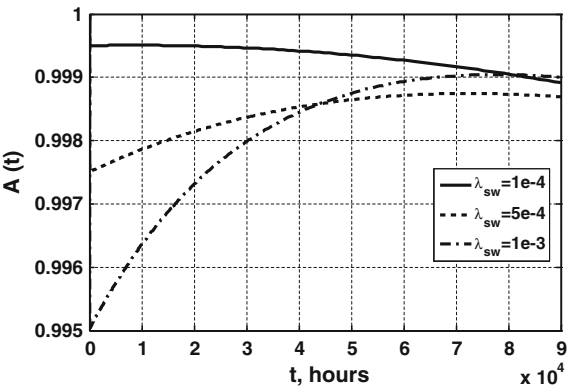


Fig. 7 Dependency of SBSpS availability function with different values of λ_{SW}

4 Case 2: Availability Model of Software-Based Space Systems with Elliptical Orbit (Recurrent Link)

4.1 Development of the Model

Availability calculations are related to SBSpS working in low demand mode and periodically (proof) verification. For such systems software repairs are initiated only when verification are performed. The verification are singular points along the time but this is not a problem as a multiphase Markovian approach may be used to deal with. Multiphase models with periodic checks are described in detail in [10]. Considering the practice of construction of such models, this paper hereinafter discusses the system unavailability parameter $U(t) = 1 - A(t)$. For the sake of analysis, the unavailability function is averaged out on the interval between the checks as Uavg(t).

SBSpS made of one periodically tested single component has 7 states as shown in Fig. 8: working (S_0 and S_1—with one hardware failure detected), nonserviceable (S_2—with two hardware failure detected), software failure detected (S_3 and S_5) and under software repair with software verification and online patching (S_4 and S_6).

Between verification its behavior is modeled by the Markovian process on the upper part of Fig. 8: it can hardware fail ($S_0 \rightarrow S_1$ and $S_1 \rightarrow S_2$), software fail ($S_0 \rightarrow S_3$ and $S_1 \rightarrow S_4$), software restart ($S_3 \rightarrow S_0$ and $S_4 \rightarrow S_1$), or under repair ($S_4 \rightarrow S_0$ and $S_6 \rightarrow S_1$). As no software verification and repair may be started within a verification interval, there is no transition from $S_3 \rightarrow S_4$ and $S_5 \rightarrow S_6$. Because the verification of the software failure has been performed before entering state S_4 and S_6, μ_{VER} is the repair rate of the component in Fig. 8.

When a verification is performed (see linking matrix on Fig. 8), a repair is started if a failure has occurred ($S_3 \rightarrow S_4$ and $S_5 \rightarrow S_6$), the component remains working if it was in a good functioning state ($S_0 \rightarrow S_0$, $S_1 \rightarrow S_1$), nonserviceable state ($S_2 \rightarrow S_2$), and in the very hypothetical case that a repair started at the previous verification is not finished, remains under repair ($S_4 \rightarrow S_4$, $S_6 \rightarrow S_6$). A linking matrix [L] may by used to calculate the initial conditions at the beginning of state i + 1 from the probabilities of the states at the end of test i. This gives the following equation:

$$\begin{bmatrix} P_0(0) \\ P_1(0) \\ P_2(0) \\ P_3(0) \\ P_4(0) \\ P_5(0) \\ P_6(0) \end{bmatrix}_{i+1} = \begin{bmatrix} 1000000 \\ 0100000 \\ 0010000 \\ 0000100 \\ 0000100 \\ 0000001 \\ 0000001 \end{bmatrix} \cdot \begin{bmatrix} P_0(\tau) \\ P_1(\tau) \\ P_2(\tau) \\ P_3(\tau) \\ P_4(\tau) \\ P_5(\tau) \\ P_6(\tau) \end{bmatrix} = \overrightarrow{P_{i+1}}(0) = [L]\overrightarrow{P_i}(\tau) \qquad (2)$$

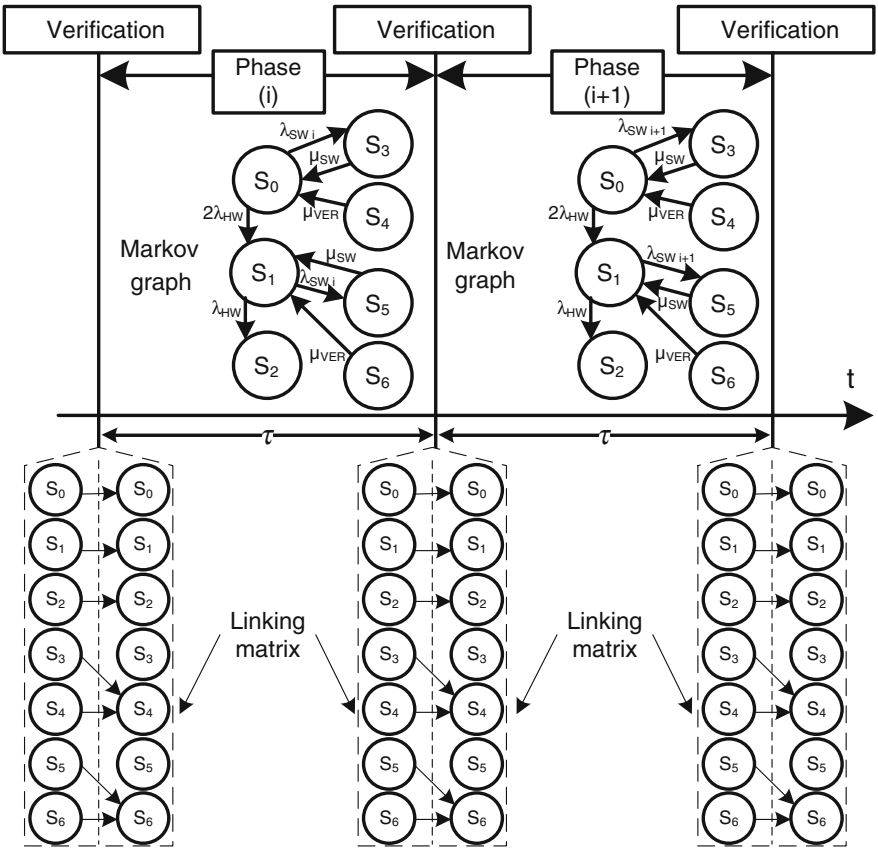


Fig. 8 Principle of the multiphase Markovian modeling of SBSpS with recurrent patches

Replacing Pi(τ) by its value, leads to an equation of recurrence which allows to calculate the initial conditions at the beginning of each verification intervals. Obtaining the instantaneous unavailability is straightforward by summing the probabilities of the states where the system is unavailable. A line vector (q_k) is helpful to express that (3) and Uavg is calculated in the way previously described through the mean down times which in turn is easy to calculate from the mean cumulated times spent in the states:

$$U(t) = \sum_{k=1}^{n} q_k P_k(t) = 1 - P_1(t) - P_2(t),\ U_{avg}(\tau) = \int_0^{\tau} U(t)dt. \qquad (3)$$

where $q_k = 1$ if the system is unavailable in state k, and $q_k = 0$ otherwise.

While building the model it is necessary to take account of changes in parameter λ_{SW} when eliminating a software defect. The defect is eliminated after its

emergence and when the next time for verification comes. But this event is probable, and one cannot tell for sure at which time interval λ_{SW} will decrease by $\Delta\lambda_{SW}$. At the onset of the next phase the probability of occurrence of the software defect is determined as the sum of probabilities P4(τ) + P6(τ) of the previous phase. Then the change of the software failure rate in the next phase is determined from the formula:

$$\Delta\lambda_{SW}(\tau+1) = \Delta\lambda_{SW} \cdot [P_4(\tau) + P_6(\tau)] \quad (4)$$

4.2 Justification of Model Input Parameters

All operations on localization of software defects, data collection, development of patches and their installation are carried out after the next scheduled test. Consequently, the length of verification increases as compared to parameter T_{UP} in the previous model: $\mu_{VER} = [1/(1/\lambda_{UP} + 1/\mu_{UP}] = 1/15\ h^{-1}$.

The values of the rest of parameters are the same as in the previous model (Table 1). Also, unlike the previous model, it assumes that detected defects are reliably eliminated (D = 1). Analysis of the model at $\lambda_{SW\ 0} = 5e{-}3\ h^{-1}$ (the value by an order of magnitude higher than that in the previous model) is of interest.

The task to investigate the model is reduced to finding value ΔT_{VER}–the frequency of preventive checks that ensures system availability function requirement of at least 0.998 for 2 years (20,000 h) of operation. One should also take into account the limitation: the frequency of preventive checks must be a multiple of the period of stable satellite link within its visual range: $\Delta T_{VER} = n\Delta T_{LINK}$, n = [1…N].

The chart in Fig. 9a presents the time interval that was specially scaled down t = [0…5000] to make unavailability curve characteristic waves clearly visible. It is

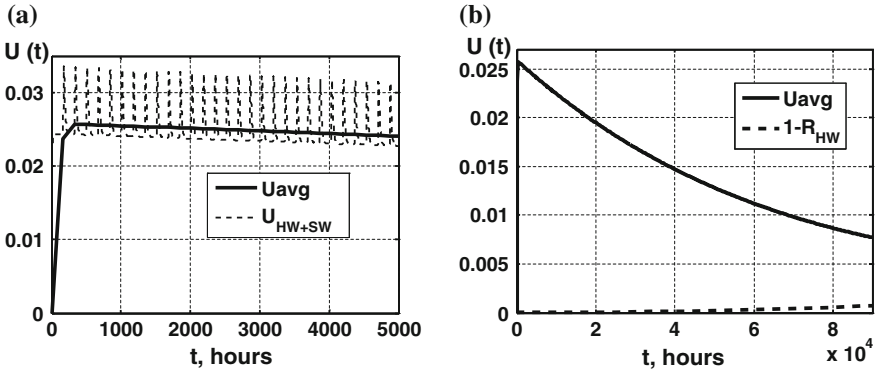


Fig. 9 Unavailability function of SBSpS with recurrent software verification and patches (**a**), the same function and unreliability function of two hardware channel (**b**)

clearly seen that curve U_{avg} slopes down with time. This demonstrates the process of software defects eliminations.

The value of the parameter ΔT_{VER} = 7 days (168 h) that ensures the value of the unavailability function of not more than 0.002 for 2 years (20,000 h) of operation (Fig. 9b) was determined by the trial-and-error method.

5 Conclusions

The paper analyzes the SBSpS as an object of patching, the regulatory framework in this area and the adequacy of existing patching and verification methods for software specifics of space systems.

The results of analysis of the models showed that a hardware reliability of 0.999 would require the failure rate of $\lambda_{HW} = 3e{-}7\ h^{-1}$. For hardware these are quite high, though realistic figures (such reliability is declared, for example, by some hard disk manufactures). If we are to consider non-recoverable hardware, then for a reliability of 0.999 we would need to ensure $\lambda_{SW} = 3.4e{-}9\ h^{-1}$. It is infeasible to ensure such reliability where it is impossible to simulate mission conditions for software testing. Introduction in the software architecture of the functional block responsible for restart allows to "mitigate" the failure rate requirements by 4 orders ($\lambda_{SW} = 6e{-}5\ h^{-1}$), however, in practice these figures are hardly achievable.

Introduction to the software architecture of additional patching blocks allows to still further reduce requirements to initial software failure rate down to $\lambda_{SW} = 5e{-}4 \ldots 5e{-}3\ h^{-1}$. In this case resilience requirements to achieve the availability level of 0.998 during the first 2 years of operation are met.

Besides, it is interesting to research such systems considering other dependability attributes, first of all, integrity and confidentiality [6]. In this case the proposed models should be enhanced taking into account results of SBSpS software component vulnerability analysis and different types of attack rate.

References

1. Risk Analysis of Rocket Space System Emergencies: 2000 Years. In: Kharchenko, V. (ed.). Safety Critical Infrastructures. Mathematical and Engineering Methods of Assessment and Ensuring, pp. 551–573. National Aerospace University "KhAI", Kharkiv (2011)
2. Pietravalle, R., Lanz, D.: Resiliency Research Snapot. The MITRE Corporation, USA (2011)
3. ECSS-E-ST-40C6-2009. Space Engineering. Software Requirements & Standards Division, Noordwijk, The Netherlands (2009)
4. Lintelman, S., Robinson, R., Mingyan, L., Bushnell, L., Poovendran, R., Sampigethaya, K.: Secure wireless collection and distribution of commercial airplane health data. IEEE Aerosp. Electron. Syst. Mag. **24**, 14–20 (2009)
5. Gashi, I., Popov, P., Stankovic, V.: Uncertainty explicit assessment of off-the-shelf software: a bayesian approach. Elsevier J. Inform. Softw. Technol. **51**(2), 497–511 (2009)

6. Trivedi, K.S. Kim, D.S., Roy, A., Medhi, D.: Dependability and security models. In: Proceedings 7th International Workshop on the Design of Reliable Communication Networks (DRCN 2009), pp. 11–20. Washington, DC, USA (2009)
7. Kharchenko, V., Odarushchenko, O., Odarushchenko, V., Popov, P.: Selecting mathematical software for dependability assessment of computer systems described by Stiff Markov Chains. In: Ermolayev, V., Mayr, H.C., Nikitchenko, M., Spivakovsky, A., Zholtkevych, G. (eds.) ICTERI-2013, CCIS, vol. 1000, pp. 146–162. Springer, Heidelberg (2013)
8. Kulba, V., Mykryn, E., Pavlov, B.: Design of Information Control Systems for Orbital Stations. Nauka, Moscow (2002)
9. Kharchenko, V., Ponochovnyi, Y., Boyarchuk, A.: Availability assessment of information and control systems with online software update and verification. In: Ermolayev, V., Mayr, H.C., Nikitchenko, M., Spivakovsky, A., Zholtkevych, G. (eds.) ICTERI-2014, CCIS, vol. 469, pp. 300–324. Springer, Heidelberg (2014)
10. IEC 61508-6:2010 Functional safety of electrical/electronic/programmable electronic safetyrelated systems. Part 6: Guidelines on the application of IEC 61508-2 and IEC 61508-3. European Committee for Electrotechnical Standardization. Brussels (2010)

A Mathematical Model to Regulate Roads Traffic in Order to Decongest the Urban Areas of Constantine City

Mouloud Khelf, Salim Boukebbab and Mohamed Salah Boulahlib

Abstract The most critical phenomenon in the road traffic is the congestion. Although the technical and technological progress realized by the humans in all domains. The road traffic remains a victim of increasing congestion when demand exceeds the capacity of the road infrastructure. In this case, the vehicle will slow down to the entrance of the road infrastructure, thereby forming a bottling in traffic. In the present work, the first step is to considering the different variables that characterize the progressive movement of vehicles on a road. In the aim to give a mathematical formulation which links the number of vehicles present at time "T" over a length "L" of the road. This enabled us to develop a mathematical model to regulate the traffic speed in real time. To validate the model, a real application is presented, who treat the congestion problem in Constantine city.

Keywords Traffic road · Congestion · Flow · Mathematical model · Measurement · Regulation · Optimization

1 Introduction

Today everyone agrees that, the transports systems will have to face a major challenge in the coming decades, to ensure sustainable mobility and people movement. For if half of the world's population (7 billion) now lives in cities (Fig. 1), the forecasts point that in 2050 over 2/3 of the world population (9 billion) will live in the city [1]. So it is imperative today that the transport infrastructures definitively making the switch to sustainable construction and less energy costly with less environmental pollution [2].

M. Khelf (✉) · S. Boukebbab · M.S. Boulahlib
Laboratoire Ingénierie des Transports et Environnement,
Faculté des Sciences de la Technologie, Université des Mentouri Constantine 1,
Campus Universitaire Zarzara, Constantine 25017, Algeria
e-mail: mouloudkhelf@yahoo.fr

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_24

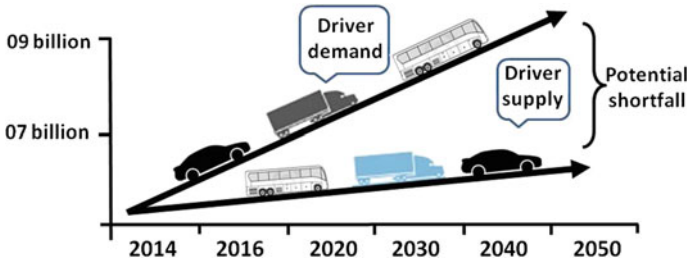


Fig. 1 Demand transport evolution

To achieve this, opinions differ and we can affirm that there exist two lines of thought. The first line of thought says that the road is a major source of pollution and, in particular, impacts on the environment: it is therefore necessary to curb the expansion of the road network to other modes of transport. The second line of thought says that the road is essential to the reliable and efficient transport system. It proposes to find the best strategy to make the organizational structure of all reliable transport infrastructures, flexible and optimized. A rational and dynamic solution but requires a clear planning, significant resources and rigor without failure.

Adopt the first line of thought can maintain the state of congestion. Worse, it can lead to aggravation, especially when one takes into account the forecasts of demographic change and urban sprawl. Adopt the second line of thought as surprising as it may seem, requires the construction of new road infrastructure. However, calculations show that to decongest the roads must be balanced: make savings at the community, reduce the time of inconvenience for users, as well as minimizing impacts of pollution on the environmental.

To develop the argument further, it's not the construction of roads which can solves the congestion problem on these last, but use them by optimal rationalization [3]. For this, the present research focuses on the notion of optimal rationalization in road traffic in urban areas to resolve this problem. It schooled be noted here that, the Algerian state has made great effort to develop public transportation (subways, trams, cable cars, trains, buses, planes, etc.). Nevertheless, individual transports such as car and motorcycle, present an evident flexibility comparatively to the public transport and took these last year's a significant market share in Constantine city.

2 The Congestion Phenomenon

The congestion of a road network is the condition where an increase in vehicle traffic, causes an overall slowdown of the latter. The term congestion is the degradation of quality of service when the number of users increases. This phenomenon is characterized by the occurrence of delays and even strangulation in

peak traffic periods, that is to say when the infrastructure capacity becomes insufficient to regulate the flow.

The problem is particularly common in large cities because the demand exceeds the capacity, then the vehicle will slowed to the entrance of the infrastructure thus forming a bottling. As each vehicle occupies a certain length of path, the length of the queue will only grow in proportion to the number of vehicles in this queued. Because of the character mesh of the road infrastructure in Constantine city, the lengthening of the queue can lead to congested parts of the road network that are auxiliary. In addition, several events may cause or aggravate congestion as: accidents, work, car breakdowns, nuisance parking, bad weather, etc. The road traffic congestion consequences are many and can be classified into three categories: economic, social and environmental [4–6].

3 The Solutions for Road Traffic Decongest

A first solution to decongest the traffic is to increase road capacity. This solution, certainly expensive, but has the enormous advantage of sustainably improve traffic flow, thereby generating fuel consumption in terms of savings and a significant reduction in environmental impacts. However, this solution unfortunately is not current at the public authorities rather favor the development of alternative modes of transport to the road. This situation, let's say that failing to increase the reception roads capacity. Another solution would be to develop regulation tool and management of the road traffic. This means, modeless the evolution of cars density in time and space. To establish models, some researchers use partial differential equations, that describe physical phenomena like fluid mechanics [7, 8]; which informs us that the flow debit is equal to the product of the concentration with the speed of the flow Q = K*V. The flow speed V is given by the equation: V = Q/K, This relationship is widely used in traffic theory for analyzing unidirectional flow [9, 10] (Fig. 2).

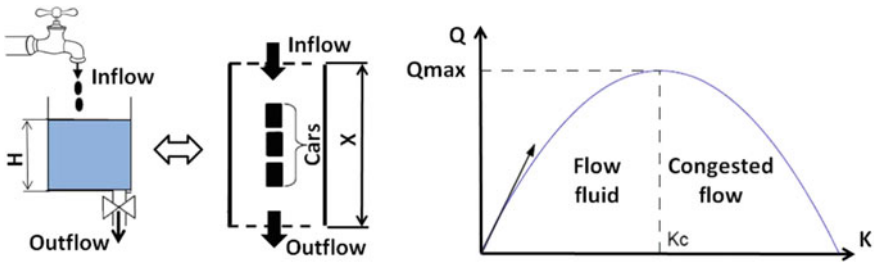


Fig. 2 The fundamental diagram (Q,K)

Although all efforts in research and development have resulted in the development of countless tools for control and management of road traffic, the problem of traffic congestion is not resolved completely [11, 12]. Personally, I think that the solution is not only technological but also and above human. It's finally time to be concerned with education, awareness and training of drivers to become real actors for better fluidity of traffic [13]. These recommendations are simple and are common sense. They are the result of a thorough analysis and a simplified and idealized mathematical model.

4 Proposition of Mathematical Model

The approach developed in the mathematical model will enable us to relate $\mathbf{Q_{max}}$ traffic flow and optimal speed $\mathbf{V_{opt}}$ that should be allowed to ensure fluidity in traffic. The steps are to study the ideal case at first, in the second step the semi-realistic case and highlighted and at the end the realistic model will be established.

4.1 Ideal Case Study

Consider a test road with single traffic lane and having a sufficient length rectilinear section. We put in this rectilinear section N identical cars of length L and whose conductors are assumed to be perfectly synchronized robots. The first car is positioned such that its front bumper is in alignment with the start line. The other cars (N − 1) are positioned one behind the other (bumper against bumper). The number of vehicles N, which passes through the starting line for the time **T** can be calculated by the following equation:

$$N = \frac{X}{L} \text{and } X = V^{*}T \Rightarrow N = \frac{V^{*}T}{L} \tag{1}$$

The flow **Q** of trail section is the number of vehicles N which passes at a constant speed **V** the line departure, divided per time **T**:

$$Q\,(\text{cars/hour})T = \frac{N(\text{cars})}{T(\text{hour})} = \left(\frac{V^{*}T}{L}\right) * \left(\frac{1}{T}\right) = \left(\frac{1}{L}\right) * V \tag{2}$$

with **Q**: numbers of vehicles per hour, **V**: speed in (m/s) and **L**: length of the car in (m).

4.2 A Semi-idealist World Study

Maintaining the same assumption of vehicles: same age, same size, same weight, same braking system, etc., however, the conductors are human. Considering the differences between the different car driver (especially reaction time), he was asked, this time, each driver to observe a safe distance from the vehicle that precedes it and which corresponds to the distance "**Lr**" would have the vehicle traveled during the reaction time "**tr**" medium of the conductors (Fig. 3).

In this case, If we admit that the reaction time is 02 (s) then we have: Lr (m) = V (m/s) * 2 (s). However the speed indicated on the car dashboard is given in (km/h). Then the following transformation 1 (m/s) = 3.6 (km/h) is necessary to calculate:

$$L_r = t_r(s) * V\left(\frac{m}{s}\right) = \left(\frac{2}{3,6}\right) * V\left(\frac{km}{h}\right) = \left(\frac{20}{36}\right) * V\left(\frac{km}{h}\right) = \left(\frac{5}{9}\right) * V\left(\frac{km}{h}\right) \tag{3}$$

Let us now give the start and ask drivers to speed up gradually to reach, after a complete turn (return to the original position) the speed **V** and keep it constant. All the cars are in motion with delay times because drivers are not synchronized. Taking into account the instant effect of the brakes, these vehicles can roll at a constant speed **V** on the trail but maintaining a safe distance "$L_r = t_r * V$". The number of vehicles **N** in this case, which passes through the starting line for the time **T** can be calculated by the formula:

$$N = \frac{X}{L + t_r * V} = \frac{V * T}{L + t_r * V} \tag{4}$$

The Eqs. (2) and (4) permit to determine the expression of the flow Q of a road straight section in semi-idealized case by the following formulation:

$$Q = N * \frac{1}{(T)} = \frac{V}{L + t_r * V} \tag{5}$$

One realizes that the flow Q, in sutured situation varies depending on the speed **V** as a hyperbolic function. For a vehicle length and a given reaction time, when the speed **V** increases, the flow Q tends to converge to a constant value equal to $\frac{1}{t_r}$. The maximum flow $\mathbf{Q_{max}} = \frac{1}{t_r}$, depend only by medium time reaction of drivers. This

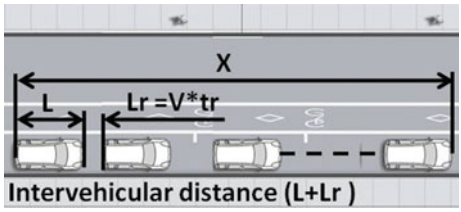


Fig. 3 The semi-idealized case

approach clearly shows that to contribute to the decongestion of roads, it's no use to drive fast. In fact, above a certain speed, the flow **Q** do not increases and remains constant.

4.3 *Case of a Semi-realistic World*

Considering the same experience on the test trail, in addition to the differences between the driver (especially reaction time), the cars do not have a braking system with immediate or instant effect. To stand still under the effect of the brake, the car will need a certain distance to dissipate the kinetic energy it has stored. The brakes need an average distance "Lc" proportional to the kinetic energy ($Lc = k*Ec = k*\left(\frac{1}{2}\right) * m*V^2$) for stopping the vehicles. Each driver must maintain a safe distance "ds" from the vehicle which precedes it. This distance is equal to "**ds = Lc + Lr**" that are travels the vehicle during a medium time "**t = tr + tc**" (Fig. 4).

Give now the start and ordain the car drivers to speed up gradually to reach after one full the speed V and keep it constant. The number of vehicles in this case, which passes through the starting line for the time T can be calculated by the following equation:

$$N = \frac{X}{L + L_r + L_c} = \frac{V * T}{L + t_r * V + \left(\frac{1}{2}\right) * k * m * V^2} \tag{6}$$

The Eqs. (2) and (6) give the expression of the flow Q of a straight section of a road in semi-realistic case:

$$Q = \frac{N}{T} = \frac{V}{L + t_r * V + \left(\frac{1}{2}\right) * k * m * V^2} \tag{7}$$

According to the Eq. (7) the flow Q, in situation of traffic saturation varies depending on the speed V as a hyperbolic function (Fig. 5).

For a vehicle length and a reaction time, the maximum flow Q is obtained when the derived function Q = f(V) is equal to zero $\left(\frac{dQ}{dV} = 0\right)$:

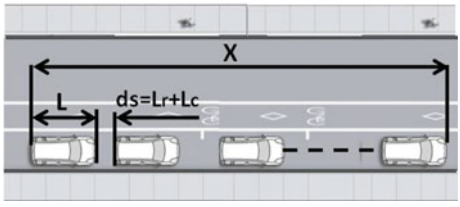


Fig. 4 The semi-realistic case

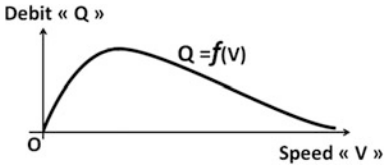


Fig. 5 The semi-realistic case

$$\frac{dQ}{dV} = \frac{\left(L - \left(\frac{1}{2}\right)k * m * V^2\right)}{\left(L + t * V + \left(\frac{1}{2}\right)k * m * V^2\right)^2} = 0,\ \left(L - \left(\frac{1}{2}\right)k * m * V^2\right) = 0 \rightarrow V_{opt} = \sqrt{\frac{2 * L}{k * m}} \tag{8}$$

$$Q_{max} = f(V_{opt}) = \frac{\sqrt{\frac{2*L}{k*m}}}{2 * L + t * \sqrt{\frac{2*L}{k*m}}} \tag{9}$$

Equation (9) give a proof that at optimum speed (V_{opt}), the flow Q is maximum and inversely proportional to the car driver reaction time. If the trail includes other traffic lanes, the method applies and the reasoning is similar, provided that drivers do not change lanes.

5 Numerical Applications: Case of Constantine City

In this application we will evaluate the section of the road linking the communes Hamma-Bouziane, El-Khroub and Aïn-Samara to the Constantine city. We have available for each day four periods and three measuring points of average traffic volume per hour and by region as illustrate in the following (Table 1):

Figure 6 illustrate clearly that the road linking Constantine city to Hamma Bouziane commune is the most dense between 09h00–16h00 with a flow Q equal 21458 (Cars/h). The average speed measured in this road is equal to 40 (km/h) and

Table 1 The data flow Q in Constantine city

Periods of day	El-Khroub-Constantine (Cars/h)	Ain Smara-Constantine (Cars/h)	Hamma Bouziane-Constantine (Cars/h)
6h00–9h00	3000	1700	4100
9h00–16h00	7042	10,254	21,458
16h00–19h00	4753	2531	5123
19h00–6h00	503	852	1325

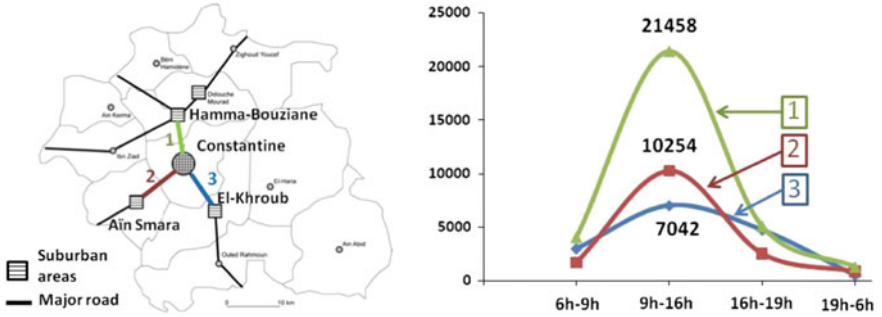


Fig. 6 The circulation cars flow in/out Constantine city

the permitted traffic speed is 60 (km/h). An impressive congestion is observed almost every day of week.

To solve this problem, we found it useful to review the speed limit authorized by the municipality to regulate traffic, in the aim to decongest this important road section for the economic exchange of the Constantine city. For this, the models developed by the Eq. (10) calculate the optimal speed of traffic in a situation of saturation. For this following assumptions are applied:

- The medium length of vehicles: L = 4 (m)
- The braking constant $C_f = \left(\frac{1}{2}\right) * k * m$ (coefficient of proportionality between the braking distance Lc and the square of the speed V^2, $L_c = C_f * V^2$). According to the physical laws [14], the braking distance is calculated by the following equation:

$$L_C = \frac{V^2}{2 * G * a} \tag{10}$$

with V = speed in (m/s), G = terrestrial gravitational acceleration equal 9.81 (m/s^2) and a = coefficient of adhesion. The coefficient of adhesion is related to the quality of the road. This coefficient varies between 0 and 1. It is generally considered that a dry road with a good surface takes a coefficient value equal 0.65 or 0.8. In our case the coefficient of adhesion "**a**" is equal to 0.8 in view of the quality of the road between Constantine city and commune of Hamma-bouziane. After calculate we obtain the results below (Fig. 7) (Table 2):

Before substituting the values of (L) and (k*m) into the Eq. (9), we obtain:

$$V_{opt} = \sqrt{\frac{2 \times L}{K \times m}} = \sqrt{\frac{2 \times 4}{2 \times 0{,}06371}} = 7{,}9236\left(\frac{m}{s}\right) = 28{,}52(km/h) \tag{11}$$

Considering the assumptions used, vehicles must observe a speed limit of 30 (km/h) to reduce the effects of traffic congestion.

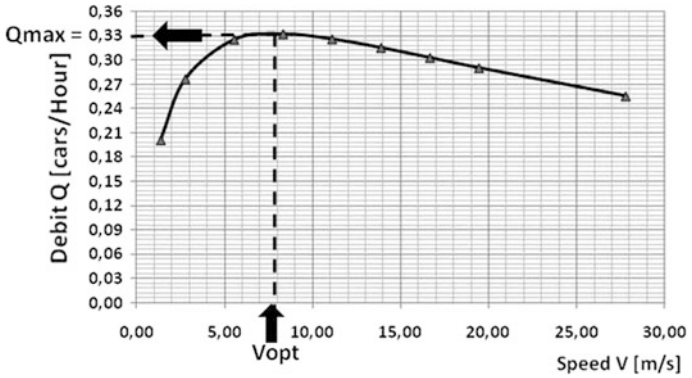


Fig. 7 The calculation results

Table 2 Results of computations

V(km/h)	V(m/s)	V^2 (m^2/s^2)	Lc = V^2/2*G*a	Cf = 1/2*k*m	Lr = (5/9)*V	Q = f(V)
5	**1.39**	1.93	0	0.063710	3	**0.20**
10	**2.78**	7.72	0	0.063710	6	**0.28**
20	**5.56**	30.86	2	0.063710	11	**0.33**
30	**8.33**	69.44	4	0.063710	17	**0.33**
40	**11.11**	123.46	8	0.063710	22	**0.33**
50	**13.89**	192.90	12	0.063710	28	**0.32**
60	**16.67**	277.78	18	0.063710	33	**0.30**
70	**19.44**	378.09	24	0.063710	39	**0.29**
100	**27.78**	771.60	49	0.063710	56	**0.26**

6 Conclusion

The last years, an enormous increase is observed in the car fleet in Algeria, causing several problems of traffic in most Algerian towns: traffic jam, congestion, accidents, pollution etc. The traditional road infrastructure (National road "RN") did not know a development, and their capacity to absorb traffic road are arrived to their limits. Added to this, the large Algerian cities are drowning in the weight of the "population concentration" and the anarchy of the urban development.

Road networks remained unchanged and generally inherited from the colonial period, while the number of inhabitants in these cities has doubled and in certain cases quadrupled. The attempts at solutions to the "baffling problem" of urban residents are introduced gradually with the entry into operation of the Algiers metro, tram of the capital, the tram of Oran and Constantine city and similar projects in other major cities including south of Algeria in the Ouargla city. The strategy of development of the transport of passengers, planified by the public authority, appears to favor public transport, and it was high time.

Modeling traffic congestion in a real world, we quickly confronted with the complexity of the task and the necessity to use powerful computers. In this framework, mathematical models have been proposed for the control and supervision traffic urban which must meet various objectives like: minimizing wait times for vehicles at intersections, optimization of traffic flows on the road network. Thus implies a considerable minimization of energy consumption, and consequently, the reduction of the air pollution.

The model developed in this article and present above gives a semi-realistic representation of road traffic in urban areas, but has the merit of being simple and educational. But, not to do many illusions, because the recommendations resulting from the model cannot, solve the problem of traffic congestion. But it's not the construction of road infrastructures, that solves the problem of congestion but use them with optimal rationalization. For this, the present research studied the notion of optimal rationalization of traffic in urban areas to solve the problem in question.

References

1. Forstall, R.L., Greene, R.P., Pick, J.B.: Which are the largest? Why lists of major urban areas vary so greatly. Tijdschrift voor economische en sociale geografie 100(3), 277–297 (2009) doi:10.1111/j.1467-9663.2009.00537.x
2. Greene, R.P., Pick, J.B.: Exploring the Urban Community: A GIS Approach. 2nd edn. Prentice Hall, Upper Saddle River, New Jersey (2012)
3. Pick, J.B., Azari, R.: Global digital divide: Influence of socioeconomic, governmental, and accessibility factors on information technology. Inform. Technol. Dev **14**(2), 91–115 (2008)
4. Weisbrod, Glen, Treyz, Don Vary George: Measuring Economic Costs of Urban Traffic Congestion to Business, Transportation Research Record. J. Transp. Res. Board (2003). doi:10.3141/1839-10
5. Helbing, D., Herrmann, H.J., Schreckenberg, M., Wolf, D.E.: Traffic and Granular Flow 99: Social, Traffic, and Granular Dynamics. Springer (2012)
6. Apte, J.S., Bombrun, E., Marshall, J.D., Nazaroff, W.W.: Global intraurban intake fractions for primary air pollutants from vehicles and other distributed sources. Int. J. Environ. Sci. Technol. **46**(6), 3415–3423 (2012). doi:10.1021/es204021h
7. Lighthill, M.J., Whitham, G.B.: On kinematic waves. II a theory of traffic flow on long crowded roads. In: Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 229(1178), pp. 317–345, May (1955)
8. Richards P.I: Shock waves on a highway. Oper. Res. **4**(1) 1956
9. Greenshields, B.D.: A study of highway capacity. In: Proceedings Highway Research Record, Washington Volume 14, pp. 448–477 (1935)
10. Payne,H.J.: Models of freeway traffic and control. In: Simulation Council Proceedings, Vol. 1, ch. 6. Mathematical Models of Public System, pp. 51–61 (1971)
11. Bouriachi Fares, Kechida Sihem et ChemssEnnahar Bencheriet, Modélisation d'un réseau du trafic urbain par Automate hybride, International conference on systems and information processing, May 12–14, Guelma, Algeria (2013)
12. Geroliminis, N., Sunb, J.: Properties of a well-defined macroscopic fundamental diagram for urban traffic. Int. J. Transp. Res. Part B: Methodol. Elsevier **45**(3), 605–617 (2011). doi:10.1016/j.trb.2010.11.004

13. Treiber, M., Kesting, A., Helbingb, D.: Three-phase traffic theory and two-phase models with a fundamental diagram in the light of empirical stylized facts. Int. J. Transp. Res. Part B: Methodological, Elsevier **44**(8–9), 983–1000, September–November 2010. doi:10.1016/j.trb.2010.03.004
14. http://alainfrancis.free.fr/exercicesetqcm/calculfreinage

The Use of a Simulation Model of the Passenger Boarding Process to Estimate the Time of Its Implementation Using Various Strategies

Artur Kierzkowski

Abstract Boarding is a significant element of the aircraft ground handling process. Its timeliness influences the planned execution of the connection network. The chapter shows the significance of the boarding process and pays attention to the importance of boarding process strategies when it is performed using a jetway. A simulation model of the passenger boarding process implemented with the use of a jetway has been presented. The model was implemented in the Flexsim software. Using this model, the boarding time for an Airbus 320 plane was estimated when it was implemented using various strategies which are presented in detail in the literature overview. The functioning of the simulation model is based on time characteristics determined on the basis of research conducted in the real system. The universality of the simulation model makes it possible to use it for any aircraft. The model also allows users to propose their own boarding strategies.

Keywords Aircraft · Boarding · Jetway · Simulation model

1 Introduction

The aircraft ground handling process is the key element of the aircraft operation process. Four operational states can be distinguished within aircraft operation process: the state of waiting for the performance of transport tasks, the state of flying, the state of ground handling and the state of technical servicing.

As shown by the research conducted [5], the critical path of the ground handling process includes processes such as: deboarding, cabin cleaning, boarding. Passenger deboarding is a process with a duration which can be foreseen to a considerable degree. Cabin cleaning is performed by the cabin crew or a third-party company which provides this service for the carrier within pre-defined time limits.

A. Kierzkowski (✉)
Faculty of Mechanical Engineering, Wroclaw University of Technology, 27 Wybrzeże Wyspiańskiego St, 50-370 Wrocław, Poland
e-mail: artur.kierzkowski@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_25

Therefore, the boarding process is of key importance for the timeliness of the ground handling process. It can be conducted in the following manner: from the airport jetway or using boarding stairs. If a jetway is used, passengers move directly from the gate to the aircraft. An advantage of this solution is the fact that in most cases, it is not necessary to change the level of passenger movement, which reduces the possibility of delays (e.g. for passengers with reduced mobility). Additionally, weather conditions (rain, snow) do not affect passengers. A disadvantage of this boarding method is the fact that passengers usually board the plane through one door (the front door), which may extend their duration. If, however, the boarding process takes place using boarding stairs, two scenarios are possible: passengers walk from the gate to the plane or passengers are taken to the plane by an airport bus. An advantage of passengers walking to the plane is the lack of necessity of paying for the bus handling service by the carrier, while the influence of weather conditions on the passenger is a disadvantage. An advantage of the strategy in which passengers are taken to the plane on an airport bus is a very short time during which the passenger is influenced by weather conditions, and a disadvantage is the use of an airport bus and also a necessity to change levels by passengers.

In the chapter, the author's attention is devoted only to the passenger boarding model using a jetway as its implementation has a greater influence on the ground handling time than in the case of boarding with the use of boarding stairs.

2 Overview of Literature on the Boarding Process

According to [5] boarding time starts when the first passenger enters the plane and ends when the last passenger is seated in his assigned seat. In the analysis of the boarding process, special attention must be paid to two notions: seat interference and aisle interferences. Seat interferences are events when a passenger wants to take his/her assigned seat (e.g. a window seat), despite the fact that one of the seats between the aisle and his/her seat is occupied (e.g. the aisle seat). Aisle interferences are events when the passenger wants to move to a row in a further part of the aircraft and is blocked by a passenger taking a seat in a closer row. A broad overview of literature concerning boarding process strategies was performed in [5]. The authors presented nine of the most frequently considered boarding strategies which are the subject of research: *random*, *back-to-front*, *by half-block*, *by row*, *by, half row*, *outside-in*, *reverse-pyramid*, *Steffen method*, *modified optimal method*.

A diagram of the *random*, *back-to-front*, *by half-block* boarding strategies is presented in Fig. 1.

The *random* method was the subject of research in [2]. According to this method, passengers stand in one queue to the gate and are allowed on board on a 'first come first served' basis. Each passenger is assigned to his/her seat.

Boarding in accordance with the *back-to-front* front method involves dividing passengers into a few groups during the boarding at the gate. Passengers with seats situated the furthest back are served first. They are followed by the next group and

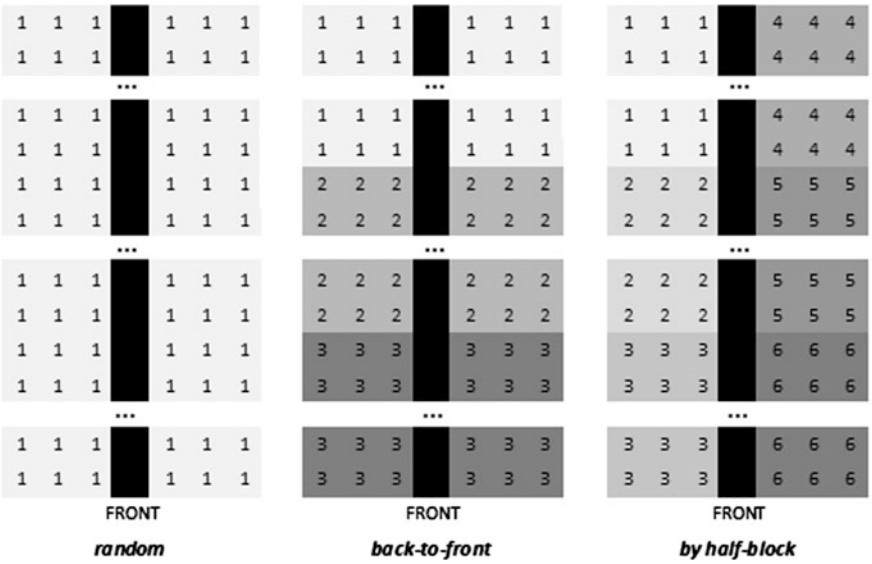


Fig. 1 Schematic representation of boarding strategies: *random*, *back-to-front*, *by half-block*

the next group and, finally, passengers seated in the first rows. In each group, passengers take their seats in accordance with the *random* strategy. Figure 1 presents a division into three groups, each of a similar size. The strategy was the subject of research by [1].

Boarding using the *by half-block* method is a certain modification of the *back-to-front* method. This strategy assumes the division of passengers into several groups [24]. The board is divided in a preliminary manner along the aisle. Next, in each subgroup, passengers are served according to the *back-to-front* strategy (each subgroup, one by one). Figure 1 presents a division into six groups. The board was divided into the left and right part and next into three groups in accordance with the *back-to-front* strategy.

A diagram of the *by row*, *by half-row*, *outside-in* boarding strategies is presented in Fig. 2.

The use of the *by row* boarding strategy takes place by rows from the last row to the first one [13]. An unquestionable problem of this strategy is the division of passengers into as many groups as there are rows on the aircraft. To configure seats on the aircraft in Fig. 2, it is necessary to divide passengers during the check-in at the gate into 30 groups.

The *half-row* strategy is a modification of the *by row* method. The board is divided in a similar manner as in the *by half-block* strategy towards the aisle [23]. Next, in each of the designated parts, the check-in is performed in accordance with the *by row* strategy. Just like for the *by row* strategy, division of passengers into groups at the gate is a problem. For the by half row strategy and distribution of seats

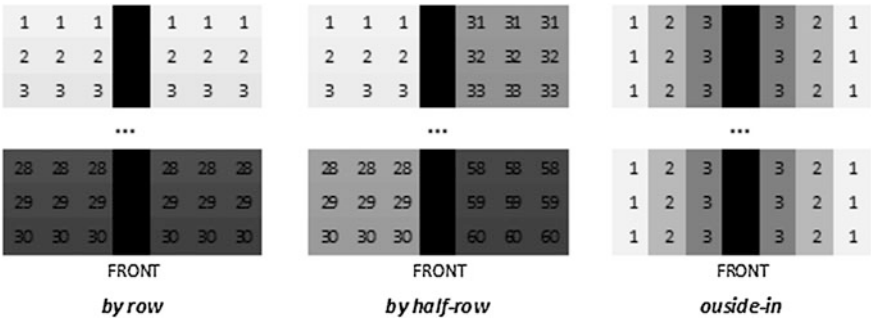


Fig. 2 Schematic representation of boarding strategies: *by row*, *by half-row*, *outside-in*

in Fig. 2, it is necessary to divide passengers into 40 groups. In each group, passengers are served in accordance with the *random* strategy.

The *outside-in* boarding strategy involves dividing the board into groups. The first group includes passengers who take window seats and the last one passengers who take aisle seats [2]. In accordance with the seat configuration in the aircraft in Fig. 2, passengers at the gate are divided into three groups. The lack of seat interference is an advantage of this strategy. A disadvantage, on the other hand, is the necessity of separating people travelling in groups (families) as they usually take seats in one row.

A diagram of the *reverse-pyramid*, *Steffen method*, *modified optimal method* boarding strategies is presented in Fig. 3.

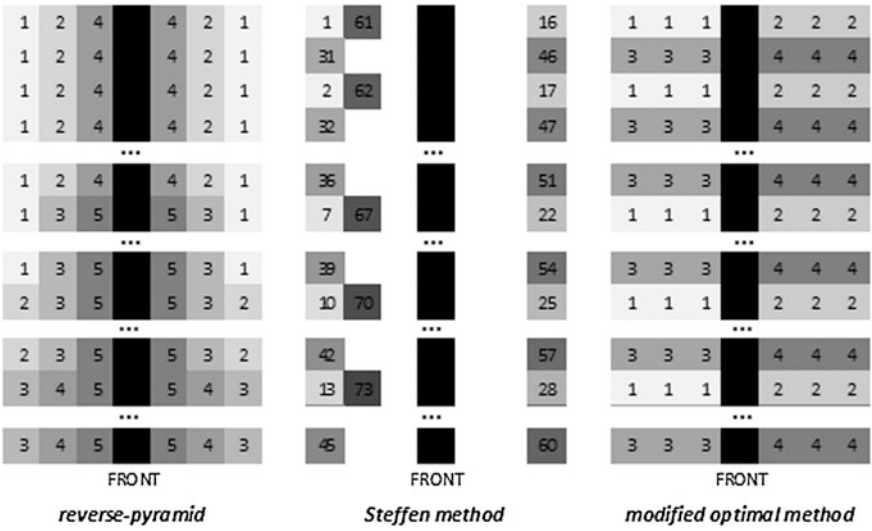


Fig. 3 Schematic representation of boarding strategies: reverse-pyramid, Steffen method, modified optimal method

The *reverse pyramid* is a modification of the *outside-in* method. The modification involves division of window seats into groups [1]. In accordance with the seat configuration in the aircraft in Fig. 3, passengers at the gate are divided into five groups. Advantages and disadvantages of this method are the same for the *outside-in* strategy.

The *Steffen method* involves boarding in a strictly specified sequence of seat occupation. Window seats in odd rows are taken first, next in odd rows on the other side [19]. Then, even rows are taken until aisle seats are reached.

The *modified optimal method* is a combination of the *by half-row* and *by half-block* boarding strategies. The board is divided along the aisle. The designated areas are divided into two subgroups: even and odd rows. Boarding takes place in the following manner: first, passengers from even rows in each of subzones and next from odd rows. In accordance with the seat configuration in Fig. 3, the strategy assumes: first, a group consisting of people in even rows on the right side of an aircraft and next another group of passengers form even rows on the left side of the aisle. Next, passengers from the group taking seats in odd rows on the right side of the aisle and the last group consists of passengers taking seats in odd rows on the left side of the aisle.

The transport system should also take into consideration aspects connected with reliability of the technical system [6–9, 14–16, 22], the effectiveness of its operation [3, 4, 17, 18, 20, 21, 25–29] and vulnerability aspects [10–12].

The studies mentioned in the literature overview are limited to the discussion and analysis of certain strategies. Studies also adopt a range of assumptions, e.g. aisle interferences are equal to 2.4 s, while seat interferences amount to 3.6 s [1]. Such assumptions may lead to considerable computational errors.

3 A Simulation Model of the Boarding Process with the Use of a Jetway

The following indices are a measure of the assessment of the implementation of the boarding process: minimum, maximum and average time of passenger boarding, the quantile of the first and third order. The simulation mode was developed using FlexSim—a computer simulation tool in the dedicated programming language—flexscript. A comparison of the implementation of the boarding strategy takes place on the basis of the average passenger boarding time, taking into account the other parameters.

For the purposes of the boarding simulation process model, the following variables were adopted:

- t_{BPAX}—time between the arrival of subsequent passengers. The random time variable between the arrival of subsequent passengers will be marked as $f(t_{BPAX})$;
- t_{BPCC}—time for checking the passenger's boarding card. The random variable of checking the boarding card by a crew member will be marked as $f(t_{BCC})$;

- v_{PAX}—the passenger's walking speed. In the developed model, it will be a constant value;
- p_{AI}—the unit probability of overtaking the passengers at the aisle;
- t_{UNPAX}—time for putting away the baggage by the passenger. The random time variable for putting away the baggage by the passenger will be marked as $f(t_{BPAX})$;
- t_{BSI}—seat interferences time. The random time variable of the seat interferences time will be marked as $f(t_{BSI})$;
- t_S—time for taking the seat by the passenger. The random time variable for taking the seat by the passenger will be marked as $f(t_S)$;
- Q_E—a queue of passengers reporting for the aircraft;
- Q_{AI}—a queue of passengers in the aisle.

The diagram of the boarding process is presented in Fig. 4. The model of the boarding process is presented in Fig. 5.

Entering input data is the first step of the passenger boarding model. All random variables are pre-set $f(t_{BPAX})$, $f(t_{BCC})$, $f(t_{BPAX})$, $f(t_{UNPAX})$, $f(t_{BSI})$, $f(t_S)$. The other values are also specified: v_{PAX}, p_{AI}. The seat and isle distribution on the aircraft is implemented. Passengers are generated in accordance with the pre-set distribution between reports $f(t_{BPAX})$. Then, the passenger is placed in the queue Q_E. Also information about the passenger's seat is also assigned to him/her. At the passenger generation moment, the passenger's boarding strategy is also taken into account. If the passenger is first in the queue for checking the boarding card, the service time is generated (boarding card check) t_{BCC} and next the passenger is placed in the queue Q_{AI}. The passengers moves at a speed of v_{PAX} from the preceding passenger to the place to be occupied. If the passenger has moved to the next passenger, it is checked

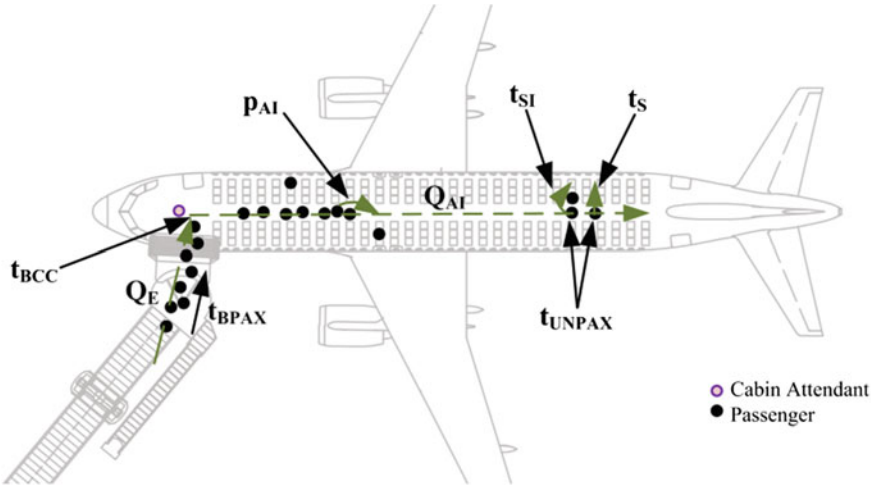


Fig. 4 Schematic representation of the boarding process

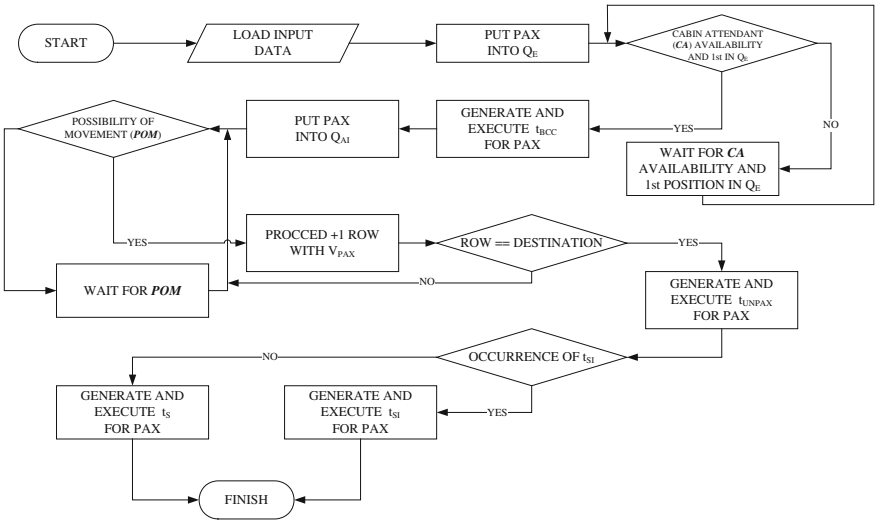


Fig. 5 Model of the boarding passengers

whether it is possible to overtake them. Overtaking between such passengers takes place with probability p_{AI}. Next, the passengers moves at a speed of v_{PAX} from the preceding passenger to the place to be occupied. Such events are performed alternately (moving and overtaking) until the passengers comes to the designated row in which he/she will occupy his/her seat. The t_{UNPAX} time is generated for them during which the passenger puts away his/her baggage and clothes into the overhead compartment. During the t_{UNPAX} time, the passenger blocks the aisle. If a seat interference occurs, the aisle is block for the replacing time t_{BSI}. If the passenger can sit in his/her seat freely, the seating time t_S is generated and the passenger does not block the aisle.

4 Application of the Model: Boarding of an Airbus 320 Aircraft

The developed model was used to estimate the boarding time for passengers of an A320 aircraft with 180 seats on board. The aircraft has thirty rows, six seats in a row with an aisle at the centre of the board.

On the basis of the research conducted, distributions of random variables were determined which are presented in (1–5):

$$f(t_{\mathrm{BPAX}}) = 1 - e^{-\frac{t_{BPAX}}{4.75}} \tag{1}$$

$$F(t_{BCC}) = \left(1 + \left(\frac{3.31}{t_{BCC}}\right)^{2.74}\right)^{-1} \tag{2}$$

$$F(t_{UNPAX}) = \frac{\Gamma_{t_{UNPAX}/38.35}(0.8)}{\Gamma(0.8)} \tag{3}$$

$$F(t_{BSI}) = \frac{\Gamma_{t_{BSI}/6.45}(0.9)}{\Gamma(0.9)} \tag{4}$$

$$f(t_S) = 1 - e^{-\frac{t_S}{3.46}} \tag{5}$$

$$v_{\mathrm{PAX}} = 1.4\left[\frac{\mathrm{m}}{\mathrm{s}}\right] \tag{6}$$

$$p_{\mathrm{AI}} = 0.24 \tag{7}$$

The consistence of empirical and theoretical distributions was verified using the Kolmogorov consistency test at a significance level of α = 0.05. In all cases, the distributions were found to be consistent with the ones proposed above (1–5) (values lower than the limit value $\lambda_{0.05} = 1.36$).

The developed model was used to estimate boarding times of passengers in accordance with the following strategies: *random*, *back-to-front*, *by half-block*, *by row*, *by half-row*, *outside-in*, *reverse-pyramid*, *Steffen method*, *modified optimal method*. Passengers were divided into individual groups as shown in Figs. 1, 2 and 3. The obtained boarding results for 180 passengers are presented in Fig. 6.

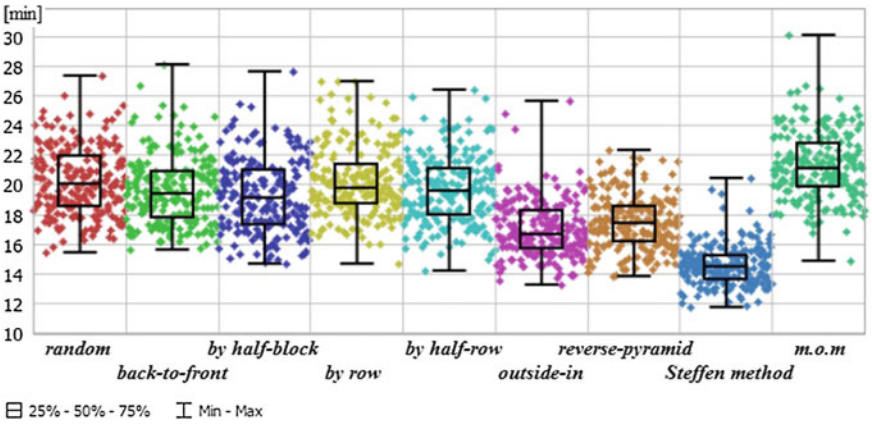


Fig. 6 Estimation of the boarding time depending on the assumed strategy

For the following strategies: *random*, *back-to-front*, *by half-block*, *by row*, *by half-row*, the median, mean boarding times of passengers are similar—they differ by no more than 1 min. The situation is similar for quantiles and minimum and maximum values. The adoption of any of the passenger boarding strategies allows the execution of the boarding process from 15 to 28 min. For the following strategies: *outside-in*, *reverse-pyramid*, the values of the median, mean and quantiles are by over 10 % lower than in the previous methods. The maximum boarding time is reduced to 22 min. The best results are obtained for the *Steffen method*, where the total boarding time ranges from 12 to 17 min with single cases reaching 20 min. The *modified optimal method* (marked as m.o.m. in Fig. 6) renders the worst results. It is caused by numerous seat and aisle interferences.

5 Summary

The study presents a simulation model for the boarding process executed via a jetway. The universality of the developed model makes it possible to use it both in the planning and optimisation of the boarding process. The developed model was used for the purposes of estimating the boarding time depending on the strategy adopted for Airbus 320. The adopted passenger divisions into groups according to the *back-to-front*, *by half-block*, *reverse-pyramid* strategies influence the results obtained. At the following stages of the research, the influence of the division into groups on the total boarding time for all of the aforementioned strategies will be checked.

Further work will be performed to analyse the sensitivity of the model. Variables which have a significant influence on the boarding time will be identified. Modification of the developed model will be introduced which are aimed at taking into account the influence of filling of individual areas of the aircraft on the time needed to put away the baggage. Moreover, a model of two dependent systems of the gate-boarding check-in will be developed, which will also allow an assessment of the influence of the passenger service process at the gate during the boarding time.

Acknowledgments The project is co-financed by the National Research and Development Centre under the Applied Research Program. This publication presents the results of research conducted in the project: "Model of logistical support for the functioning of the Wrocław Airport" realized by the Wrocław University of Technology and Wrocław Airport consortium.

References

1. Bazargan, M.: A linear programming approach for aircraft boarding strategy. Eur. J. Oper. Res. **183**(1), 394–411 (2007)

2. Ferrari, P., Nagel, K.: Robustness of efficient passenger boarding strategies for airplanes. Transp. Res. Rec. **1915**(1), 44–54 (2005)
3. Giel, R., Plewa, M.: The evaluation method of the process of municipal waste collection. In: CLC 2015: Carpathian Logistics Congress—Congress Proceedings (2015)
4. Giel, R., Młyńczak, M., Plewa, M.: Logistic support model for the sorting process of selectively collected municipal waste. In: Theory and engineering of complex systems and dependability: proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, June 29—July 3, 2015, Brunów, Poland. Springer, cop. 2015. s. 369–380 (2015)
5. Jaehn, F., Neumann, S.: Airplane boarding. Eur. J. Oper. Res. **244**, 339–359 (2015)
6. Jodejko-Pietruczuk, A., Werbińska-Wojciechowska, S.: Block inspection policy for non-series technical objects. Safety, reliability and risk analysis: beyond the horizon. In: Steenbergen, R. D.J.M., VanGelder, P.H.A.J.M., Miraglia, S. et al. (eds.) Proceedings of 22nd Annual Conference on European Safety and Reliability (ESREL) 2013, Amsterdam, Netherlands, Sep 29–Oct 02, 889–898 (2014)
7. Jodejko-Pietruczuk, A., Mlynczak, M., Zajac, M.: Assessment of economical lifetime of heavy-duty machines. Case study. In: Reliability, Risk and Safety: Theory and Applications Vols. 1–3, 531–534 (2010)
8. Kisiel, T., Valis, D., Zak, L.: Application of regression function—two areas for technical system operation assessment. In: Clc 2013: Carpathian Logistics Congress—Congress Proceedings, pp. 500–505 (2013)
9. Koucky, M., Valis, D.: Reliability of sequential systems with a restricted number of renewals. Risk Reliab. Soc. Saf. **1**(3), 1845–1849 (2007)
10. Kowalski, M., Magott, J., Nowakowski, T., Werbińska-Wojciechowska, S.: Exact and approximation methods for dependability assessment of tram systems with time window. Eur. J. Oper. Res. **235**(3), 671–686 (2014)
11. Kwasniowski, S., Zajac, M., Zajac, P.: Telematic Problems of Unmanned Vehicles Positioning at Container Terminals and Warehouses, pp. 391–399. Springer, Transport Systems Telematics (2010)
12. Magott, J., Nowakowski, T., Skrobanek, P., Werbińska, S.: Analysis of possibilities of timing dependencies modeling-example of logistic support system. In: Martorell, S, Soares, CG, Barnett, J. (eds). Safety, Reliability and Risk Analysis: Theory, Methods and Applications, vols. 1–4, Proceedings of European Safety and Reliability Conference (ESREL)/17th Annual Meeting of the Society-for-Risk-Analysis-Europe (SRA-Europe) Valencia, Spain, Sep 22–25, 2008, s. 1055–1063 (2009)
13. Nyquist, D.C., McFadden, K.L.: A study of the airline boarding problem. J. Air Transp. Manag. **14**(4), 197–204 (2008)
14. Restel, F.J.: Measures of reliability and safety of rail transportation system. In: Advances in Safety, Reliability and Risk Management—Proceedings of the European Safety and Reliability Conference, ESREL, pp. 2714–2719 (2012)
15. Restel, F.J.: Impact of infrastructure type on reliability of railway transportation system. J. Konbin **25**(1), 21–36 (2013)
16. Restel, F.J.: Reliability and safety models of transportation systems—A literature review. In: PSAM 2014—Probabilistic Safety Assessment and Management (2014)
17. Siergiejczyk, M., Krzykowska, K., Rosiński, A.: Parameters analysis of satellite support system in air navigation. Adv. Intell. Syst. Comput. **1089**, 673–678 (2015)
18. Stańczyk, P., Stelmach A.: Modeling of aircraft during take-off and landing operations using Artificial Neural Networks. In: Safety and Reliability of Complex Engineered Systems—Proceedings of the 25th European Safety and Reliability Conference, ESREL (2015)
19. Steffen, J.H., Hotchkiss, J.: Experimental test of airplane boarding methods. J. Air Transp. Manag. **18**, 64–67 (2012)
20. Tubis, A., Werbińska-Wojciechowska, S.: Inventory management of operational materials in road passenger transportation company—case study. In: CLC 2013. Carpathian Logistics Congress—Congress Proceedings, pp. 65–70 (2013)

21. Tubis, A., Werbińska-Wojciechowska, S.: Safety measure issues in passenger transportation system performance: case study. In: Steenbergen, R.D.J.M. (eds.) Safety, reliability and risk analysis: beyond the horizon: proceedings of the European Safety and Reliability Conference, ESREL 2013, Amsterdam, The Netherlands, 29 Sep–2 Oct 2013. CRC Press/Balkema, Leiden, pp. 1309–1316 (2014)
22. Valis, D., Zak, L., Pokora O.: Engine residual technical life estimation based on tribo data. Eksploatacja I Niezawodnosc- Maintenance Reliab. **16**(2), 203–210 (2014)
23. van den Briel, M.H.L., Villalobos, J.R., Hogg, G.L.: The aircraft boarding problem. In: Proceedings of the 12th annual industrial engineering research conference (IERC'03), Portland, May 19–21, Portland (2003)
24. van Landeghem, H., Beuselinck, A.: Reducing passenger boarding time in airplanes: A simulation based approach. Eur. J. Oper. Res. **142**(2), 294–308 (2002)
25. Werbiska-Wojciechowska S., Zając P.: Use of delay-time concept in modelling process of technical and logistics systems maintenance performance. Case study. Eksploatacja i Niezawodność—Maintenance Reliab. **17**(2), 174–185 (2015)
26. Zając, M.: Principles of work load in intermodal transshipment point, CLC. Carpathian Logistics Congress—Congress Proc. **2013**, 685–690 (2013)
27. Zajac, M., Swieboda, J.: The method of error elimination in the process of container handling. Int. Conf. Mil. Technol. (ICMT), pp. 1–6 (2015)
28. Zajac, M., Swieboda, J., An Unloading Work Model at an Intermodal Terminal. Theory and Engineering of Complex Systems and Dependability, pp. 573–582. Springer (2015)
29. Zając, P.: Evaluation method of energy consumption in logistic warehouse systems. Springer, ISBN 978-3-319-22043-7 (2015)

WLAN System with Iterative Decoding of OFDM Multi-symbols

Robert Kotrys, Maciej Krasicki, Piotr Remlein, Andrzej Stelter and Paweł Szulakiewicz

Abstract In this paper, a transmission scheme with iterative processing of received signals is proposed for Wireless Local Area Networks (WLANs) employing spatial multiplexing. The scheme consists in applying channel coding, interleaving and iterative detection and decoding (IDD) separately for a tuple of OFDM symbols (hereinafter called an OFDM multi-symbol), transmitted simultaneously in the same signaling interval via different space streams. In a conventional approach to iterative decoding, high receiver latency occurs because of the need to wait until the whole (possibly very large) data frame has been acquired by the receiver prior to the start of any decoding. In the proposed solution, the latency is limited to single OFDM symbol interval. This is a great advantage when iterative decoding is used.

Keywords Wireless LAN · OFDM modulation · Iterative decoding

1 Introduction

Multiple-input–multiple-output (MIMO) systems increase the spectral efficiency for wireless systems [1]. Among space-time techniques vertical Bell Labs layered space–time (V-BLAST) offer the best tradeoff between performance and

R. Kotrys (✉) · M. Krasicki · P. Remlein · A. Stelter · P. Szulakiewicz
Faculty of Electronics and Telecommunications,
Poznan University of Technology, Poznan, Poland
e-mail: Robert.Kotrys@put.poznan.pl

M. Krasicki
e-mail: Maciej.Krasicki@put.poznan.pl

P. Remlein
e-mail: Piotr.Remlein@put.poznan.pl

A. Stelter
e-mail: Andrzej.Stelter@put.poznan.pl

P. Szulakiewicz
e-mail: Pawel.Szulakiewicz@put.poznan.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_26

complexity [2]. However, the traditional methods of detection adopted in the V-BLAST do not work well when the channel coding and interleaving is used.

Bit-Interleaved Coded Modulation with Iterative Decoding (BICM-ID) [3] is well known for its excellent performance. In the literature, BICM-ID has also been considered for multi-stream OFDM systems [4, 5]. Although up-to-date WLAN specifications (802.11n/ac) incorporated OFDM-BICM, it seems as if they underrated the advantages of iterative decoding, as they keep—with no choice—Gray labelling and a high-constraint-length convolutional code, both improper for iterative decoding purposes. Apart from the compatibility with previously existing WLANs, there are two reasons for such position of a standardization body, i.e., high decoding latency and high computational payload of iterative turbo-decoding algorithms.

Iterative detection and decoding consists in exchanging so-called extrinsic information between the demapper and the decoder. These devices are separated by (de-interleaver blocks. The iterative procedure can only begin after the whole frame has been acquired and the complete codeword has been demapped and deinterleaved. Additionally, each decoding iteration consumes a considerable amount of time, since the demapper must wait until the decoder has output its up-to-date extrinsic information, which involves analyzing the complete codeword.

Authors of the paper propose to apply the well-known techniques (IDD, MMSE receiver) [6] in the future WLAN high throughput multi-stream transmission. The results of the simulation show several advantages of the proposed system in comparison to high throughput multi-stream transmission technique specified in the IEEE 802.11ac. The proposed system requires to change only that part of the 802.11ac PHY specification which concerns high throughput multi-stream transmission method.

2 OFDM Multi-symbol

The authors claim that the latency issue can be overcome in the case of WLANs exploiting spatial multiplexing. To achieve this goal, it is proposed to apply encoding, interleaving and iterative detection and decoding separately for each OFDM multi-symbol, i.e., a tuple of OFDM symbols, transmitted in the same signaling interval, as shown in Fig. 1.

A data frame is sliced into blocks that contain bits to be transmitted within one OFDM multi-symbol. Each block includes extra tail bits to ensure that every maximum likelihood path terminates in a state known to the receiver. The authors propose to use a short-constraint-length encoder, like $[5,7]_8$ instead of the conventional one $[171,133]_8$. It is simple and robust for iterative decoding. Additionally, it strongly reduces requirement for decoding computation power and limits the amount of data payload wasted for the transmission of extra tail bits

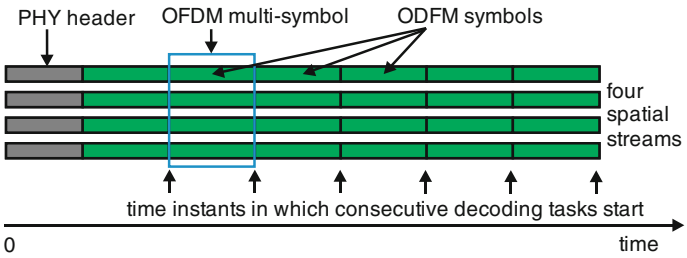


Fig. 1 Proposed structure of WLAN frame

ending every OFDM multi-symbol data block. But honestly, the number of the transmitted tail bits is only a small fraction of all data bits falling into one OFDM multi-symbol, i.e., 4 bits (0.48 %) for the $[5,7]_8$ encoder in the case of transmitting 4 streams of 16-QAM signals over a 20 MHz-wide channel. The applied interleaver is of a random type. Thanks to a high number of transmit streams and a huge number of subcarriers, even for the case of a 20-MHz-wide channel, the interleaver holds (to a certain extent) the ideal interleaver features, such as a weak correlation of fading coefficients affecting the subsequent encoded bits. It is essential to reach maximum outcomes from iterative decoding.

The last demand to be met by the transmitter is a different labelling used in the procedure of mapping the encoded bits onto complex symbols. The Gray labelling is the worst choice if one desires to benefit from iterative decoding. As shown in [7, 8], the labellings maximizing asymptotic coding gain should be used instead.

The proposed OFDM multi-symbol solution mitigates the latency required to start iterative decoding and, additionally, speeds up the execution of every iteration. Optionally, a couple of OFDM multi-symbols can be decoded in parallel by multi-core processors or independent hardware decoders.

Obviously, the iterative decoder, operating according to the Maximum-A Posteriori probability (MAP) criterion [9], is characterized by a much higher computational complexity than a Viterbi algorithm. What is more, apart from the decisions made on data bits, the iterative decoder must generate estimates of encoded bits. But the higher computational payload would be partially compensated by using the low-constraint-length encoder with its smaller number of states, i.e., the $[5,7]_8$ encoder has only 4 states in comparison with 64 states of the conventional $[171,133]_8$. Moreover, the simple 4 states code with IDD has better decoding efficiency, indicated by a lower BER (bit error rate) at a given Eb/N0 (energy per bit/noise power spectral density ratio), than 64 states code with Viterbi decoding. Another action that may be taken in order to limit the computational payload of the iterative receiver is to use sub-optimal decoder design.

3 Simulation Setup

To examine the performance of the proposed transmission scheme, a simulation experiment is conducted in MATLAB. The system operates in the 5 GHz band and uses only one 20 MHz-wide channel. It is the worst case from the iterative decoding point of view, as the interleaver depth grows with the channel bandwidth and the number of spatial streams. Both the transmitter and the receiver are arbitrarily equipped with 4 antennas, so that 4 space streams are exploited. The transmitter design, shown in Fig. 2, comprises channel encoding ($[7,5]_8$ code) and pseudorandom interleaving, both done separately for bits belonging to every single OFDM multi-symbol; the number of encoded bits per OFDM multi-symbol is 832 (4 bits per subcarrier for 16-QAM modulation × 52 data subcarriers × 4 space streams).

The complex symbols occupying the subcarriers are converted into time-domain samples of their respective OFDM symbols by means of IFFT, which—prepended by the conventional 0.8 μs cyclic prefix—are transmitted through a baseband channel. Several channel models are considered: the uncorrelated Rayleigh fading channel (with no correlation between fading coefficients affecting subsequent subcarriers), and the standard 802.11n channel models: B, D and E, specified in [10, Appendix 3.1]. For each channel model, no time correlation of fading coefficients for subsequent OFDM symbols in one space stream, as well as no cross-correlation between fading coefficients for different space streams is assumed.

At the receiver, shown in Fig. 3, the cyclic prefix is removed and then the time domain samples, received by consecutive antennas, are transformed by the FFT to the frequency domain.

The frequency domain samples are the sums of the symbols transmitted on given subcarriers through different space streams, affected by noise and fading. The MMSE multi-stream decorrelator [6] is applied to recover individual symbols (ideal channel side information is assumed). Then the iterative decoding starts. As mentioned above, it can run separately for bits belonging to every OFDM multi-symbol. The demapper and decoder exchange their extrinsic information in the form of log-likelihood ratios, LLRm and LLRc, respectively, related to the encoded bits. Having done a sufficient number of iterations, the decoder outputs

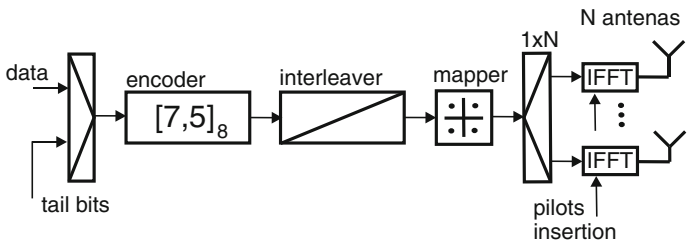


Fig. 2 Block diagram of the multi-stream transmitter

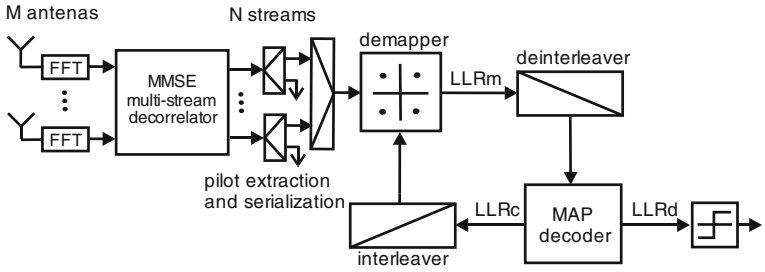


Fig. 3 Block diagram of the multi-stream iterative receiver

LLRd—the sequence of LLRs related to the data bits connected with a given OFDM multi-symbol. Passed through the decision unit, the LLRs become the estimates of the data bits.

As a reference, an OFDM-based system with a conventional $[171,133]_8$ encoder, a multi-symbol-length block interleaver provided by the 802.11ac specification, Gray-labelled 16 QAM symbols, and a non-iterative receiver, comprising Viterbi decoder and soft-decision demapper is considered. It is assumed for the reference system that the length of the data frame is the same as for the proposed scheme.

4 Simulation Results

The performance of the compared systems is expressed by FER (Frame Error Rate) versus Eb/N0 characteristics, shown in Fig. 4 for 16-QAM modulation and Fig. 5 for 64-QAM modulation. First, let us analyze the transmission over standardized WLAN channel models E ("large office") affecting the symbols transmitted on different subcarriers. In that case, the first-iteration performance of the proposed scheme (rightmost solid line with circles) is worse than the FER exhibited by the reference system (the dashed line with diamond). But, in the second iteration (which is the first complete iteration), the proposed scheme outperforms the reference system for 0.6 dB for 16-QAM and 1.4 dB for 64 QAM modulation at 1 % FER level. The execution of subsequent iterations results in a further drop in BER, e.g., in the 8th iteration (the leftmost solid line with circles), there is as much as 5.3 dB for 16-QAM and 5.8 dB for 64-QAM of gain over the reference system at 1 % FER level.

Figures 6 and 7 present the performance of the proposed scheme with 16-QAM and 64-QAM modulation, over standardized WLAN channel models B, D, and E. For all of them, iterative decoding converges, i.e., the FER diminishes from one iteration to another. However, the performance improvement due to iterative processing is not equal for every channel type. The most significant gain is observed for a highly dispersive environment, represented by the E ("large office") WLAN

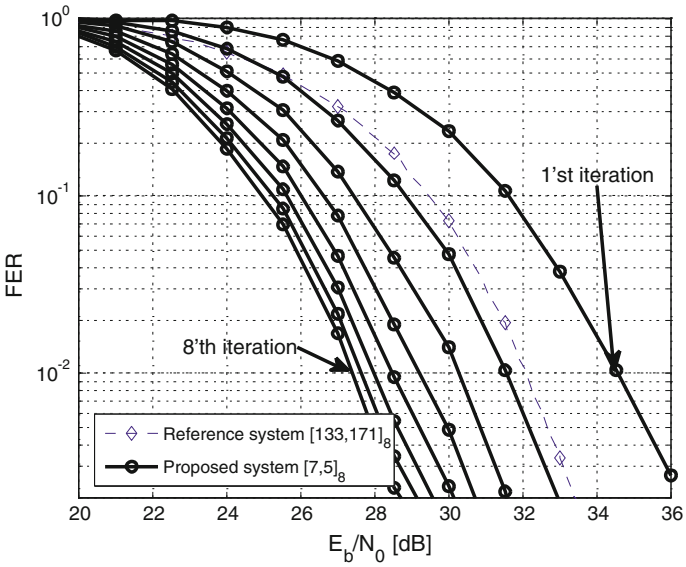


Fig. 4 FER versus Eb/N0 for the proposed scheme and for the reference system transmitting over standardized 20 MHz WLAN channel models E with 4 space streams and 16-QAM modulation

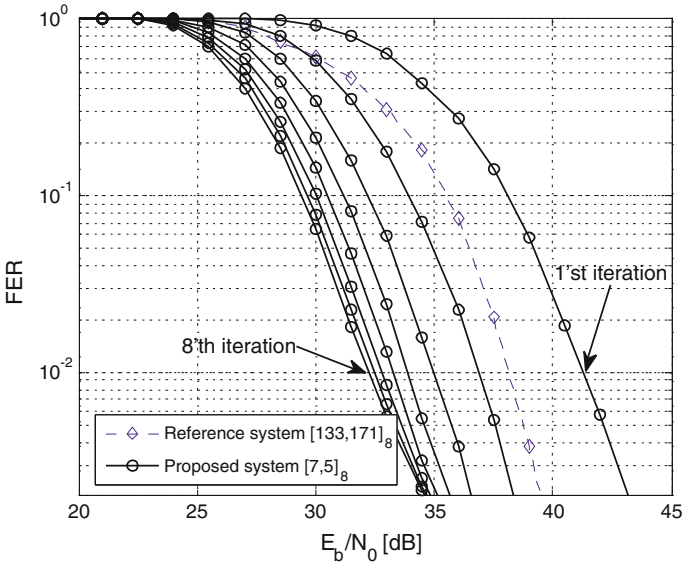


Fig. 5 FER versus Eb/N0 for the proposed scheme and for the reference system transmitting over standardized 20 MHz WLAN channel models E with 4 space streams and 64-QAM modulation

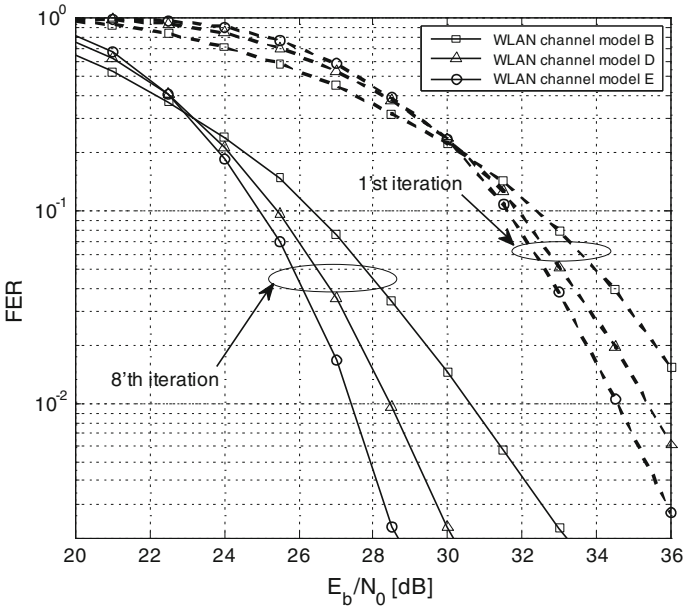


Fig. 6 FER versus Eb/N0 after 8th iteration for the proposed scheme transmitting over standardized 20 MHz WLAN channel models B (square), D (triangle) and E (circle) with 4 space streams and 16-QAM modulation

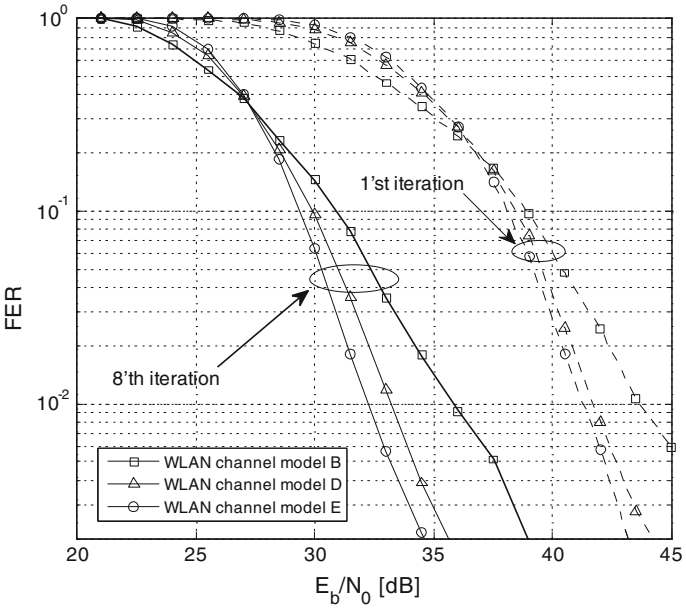


Fig. 7 FER versus Eb/N0 after 8th iteration for the proposed scheme transmitting over standardized 20 MHz WLAN channel models B (*square*), D (*triangle*) and E (*circle*) with 4 space streams and 64-QAM modulation

channel model, i.e., about 5.3 dB at 1 % FER level in the 8th iteration. For low-dispersive environments, such as B ("a typical flat") channel model, there is a relatively small iterative decoding gain.

5 Conclusion

In the case of spatial multiplexing, the transmission scheme proposed in the paper seems to be an attractive solution for future multi-antenna WLAN systems to save power and bandwidth. It facilitates dealing with the high latency of iterative decoding by considering every OFDM multi-symbol as if it were a separate data frame. Thereby, there is no need to wait for the reception of the original (possibly large) data frame before the iterative decoding starts. What is more, it is possible to process a couple of multi-symbols in parallel.

The mapping rule, convolutional code, and interleaver design are re-defined to meet the requirements of effective iterative decoding. The reason for which multi-stream transmission is assumed is the desire to transmit consecutive bits of the OFDM multi-symbol over fairly uncorrelated channels and to keep a sufficient interleaver depth, which equals the number of the encoded bits belonging to the OFDM multi-symbol. There is a strong correlation between the fading coefficients affecting adjacent subcarriers, and hence, the proposed technique would be inefficient in the case of a single spatial stream.

When applied to WLANs transmitting over 40–80- or 160-MHZ wide channels, the proposed scheme is expected to derive even more benefits from iterative decoding than for the analyzed case of 20-MHz-wide channels due to a higher interleaver depth.

Acknowledgments This work was supported by the Polish Ministry of Science and Higher Education under research grant DS-81-146-DSPB/2016.

References

1. Foschini, G.J.: Layered space-time architecture for wireless communication in a fading environment when using multi-element antennas. Bell Labs. Tech. J. **1**, 41–59 (1996)
2. Wolniansky, P. W., Foschini, G. J., Golden, G. D., Valenzuela, R. A.: V-BLAST: An architecture for realizing very high data rates over the rich-scattering wireless channel. In: Proceedings of URSI International Symposium Signals, System, Electron, pp. 295–300 (1998)
3. Chindapol, A., Ritcey, J.A.: Design, analysis, and performance evaluation for BICM-ID with square QAM constellations in Rayleigh fading channels. IEEE J. Sel. Areas Commun. **19**(5), 944–957 (2001)
4. Lee, H., Lee, B., Lee, I.: Iterative detection and decoding with an improved V-BLAST for MIMO-OFDM systems. IEEE Sel. Areas Commun. **3**(24), 504–513 (2006)

5. Boronka, A. and Speidel, J.: A low complexity MIMO system based on BLAST and iterative anti-Gray-demapping. In: Proceedings IEEE Personal, Indoor and Mobile Radio Communications Conf., Beijing, China, pp. 1400–1404 (2003)
6. Liu, D., Fitz, M.: Low complexity affine MMSE detector for iterative detection-decoding MIMO OFDM systems. IEEE Trans. Commun. **56**(1), 150–158 (2008)
7. Huang, Y., Ritcey, J.: Optimal constellation labeling for iteratively decoded bit-interleaved space-time coded modulation. IEEE Trans. Inf. Theory **51**(5), 1865–1871 (2005)
8. Krasicki, M.: Comments on 'Optimal constellation labeling for iteratively decoded bit-interleaved space-time coded modulation. IEEE Trans. Inf. Theory **58**(7), 4967–4968 (2012)
9. Benedetto, S., Divsalar, D., Montorsi, G., Pollara, F.: A soft-input soft-output APP module for iterative decoding of concatenated codes. IEEE Commun. Lett. **1**(1), 22–24 (1997)
10. Perahia, E., Stacey, R.: Next generation wireless LANs 802.11n and 802.11ac. Cambridge University Press (2013)

Context Information in a Collaborative Recommender System Deployed in Real Environment

Urszula Kużelewska

Abstract Modern e-commerce solutions (WWW services, e-stores, news portals) develop continuously, gathering and offering more and more new, interesting and various items. Unfortunately, common users are not able to deal with this information overload and reach most of them. They limit to the most popular, however often not the most interesting to them resources. A solution for this problem are personalized recommender systems. There are some popular and effective methods to build a good recommendation system: collaborative filtering, content-based, knowledge-based and hybrid. Another approach, which made a significant progress over the last several years, are context-aware recommenders. There are many additional information related to the context or application area of recommender systems, which can be useful to generate accurate propositions, e.g. user localisation, items categories or attributes, a day of a week or time of a day, weather. Important issue is evaluation of recommender systems effectiveness. Usually, they are only assessed with respect to their prediction accuracy (RMSE, MAE). However, in real environment recommendation lists are finally evaluated by users who take into consideration many various factors, like novelty or diversity of items. In this article a multi-module collaborative filtering recommender system with consideration of context information is presented. The context is included both in post-filtering module as well as in similarity measures simply extended with category relationship. Evaluation was made off-line with respect to prediction accuracy and on-line, on real shopping platform.

Keywords Recommender system · Collaborative filtering · Contextual recommendations

U. Kużelewska (✉)
Bialystok University of Technology, Wiejska 45a, 15-351 Bialystok, Poland
e-mail: u.kuzelewska@pb.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_27

1 Introduction

Nowadays, recommender systems are widely used in many areas as a solution to deal with information overload. The amount of information appeared on the Internet increases rapidly. There are many new services, web pages of companies, blogs, shops, music, video, etc. The unconstrained next step to cope with searching them is to use new technologies that can assist us to find resources of interest among the overwhelming available items. One of such tools are recommender systems (RS), which are electronic applications with the aim to generate for a user a limited list of items from a large items set. In case of personalised RS the list is constructed basing on the active user's and other users' past behaviour. People interact with recommender systems by visiting web sites, listening to the music, rating the items, doing shopping, reading items' description, selecting links from search results. This behaviour is registered as access log files from web servers, or values in databases: direct ratings for items, the numbers of song plays, content of shopping basket, etc. After each action users can see different, adapted to them, recommendation lists depending on their tastes [4, 11, 14].

Context information is additional valuable data, which is worthy to include in recommendation process. To propose a restaurant for a customer the nearest places should be recommended. In shops, before Christmas, the best suggestion are the items, which could be gifts. Razors are inappropriate recommendation for women.

In [2] the importance of the contextual information in recommender systems has been mentioned. The authors define it as information, which is known a priori and characterised by additional related to the domain factors having a known hierarchical structure that does not change significantly over time. Due to great attention on this issue and many articles, that have appeared recently [1, 5, 6, 10, 12], finally, the Context-Aware Recommender System (CARS) field has been formed.

The methods, which belong to CARS can be divided into pre-filtering, post-filtering and contextual modelling methods [13]. Pre-filtering algorithms apply contextual knowledge e.g. to remove irrelevant data before recommendations calculation, which is then performed with standard methods. Post-filtering approach uses common algorithms to generate recommendations, as well, then the contextual information is used to adjust recommendation lists. The last type, contextual modelling, use this background data in the process of recommendation generation.

One of the first pre-filtering solutions is *exact pre*−filtering [8], in which the ratings not related to the specific context of interest are removed before recommendation calculation. Another example is *item splitting* and *microprofiling* proposed by Baltrunas and Ricci [3]. They split user profiles into set of overlapping subprofiles representing the given user in a particular context. An example of contextual modelling is RPMF proposed by [16]. The backgound information is encoded in or reflected by the user-specific and item-specific latent factors. Based on this, tree based random partition is applied to split the user-item-rating matrix by grouping users and items with similar contexts, and then apply matrix factorization

to the generated sub-matrices. Finally, a framework for building context-aware recommender system was proposed by Hussein et al. in [10].

Effectiveness of recommender systems is usually measured as their predictive ability. For this purpose error measures (RMSE, MAE) are used to estimate removed ratings from test sets and compare them to real values. Such approach, although can be calculated off-line, is often insufficient to evaluate their real effectiveness. This is the reason of wide research in the field of recommender systems evaluation. One of propositions is sales diversity, claimed as relevant quality of recommendation from the business point of view [7]. It can be measured, for instance, by the total number or the ratio of items that are recommended in the top-N to at least one user. Vital index is recommender system coverage, which can be defined as user coverage (fraction of users that receive at least one recommendation) or as items coverage (fraction of items that can be recommended) [15]. In prediction ability calculation, if a recommendation list is shorter than required or even empty—it does not affect the error negatively. The authors of [9] suggest to report both accuracy and coverage as performance measures of recommender systems.

This paper contains results of experiments on collaborative filtering recommender system *what2buy* with context information included. There were tested both post-filtering and contextual modelling methods. The post-filtering approach is based on collaborative filtering item-based and user-based techniques with standard similarity metrics, whereas the contextual modelling solution involves relationship among item's categories in similarity measure which is used in item-based collaborative filtering module, as well. Quality of prediction (RMSE) and real effectiveness (items from recommendations, which were bought by customers in real environment as well as recommendations' coverage) was examined in the experiments.

2 Architecture and Description of What2buy Recommender System

The architecture of the system and its individual parts were strictly designed for this particular selling platform. It was created and implemented in re.com.sys Ltd. company. There are over 50,000 products and nearly 1000 categories. There were the following aims to achieve: increase conversion rate, increase average number of items in baskets and propose for loyal customers the items, which are particularly interesting for them.

The conversion rate is an index of users, who bought something during the same visit to the number of users, who were only visiting items. To increase it, new visitors should see interesting products without time-consuming searching. The only information gathered from them were: the visited product pages and the

categories the products belong to. It was observed, that the time spent in the store was very short: the visited only several pages from one or two categories.

Relatively high number of items in basket is efficient for both: seller and customer. It reduces the influence of shipping cost. Customers eager to increase the content of their carts, but only if they see an interesting offer.

Last aim is related to increase loyalty of regular customers. The have a great history of transactions, which is a good source of profiles of their preferences. They have favourite products and categories, they know the rules of navigation in the service. For them the related other products from different categories should be proposed.

Taking into considerations the above challenges and objectives, there was proposed a hybrid collaborative filtering system with 2 source of data and 2 types of recommenders. The data were gathered from transactions (*transSData*) and from user pre-transactional behaviour—visits and operations on basket (*visSData*). Transactions data was used as input to user-based collaborative filtering module (*UBR*) which is designed to generate recommendations for regular customers. The data *visSData* is a source to item-based collaborative filtering module (*IBR*) to generate propositions for new customers as well as users, who only wander among the pages.

It should be noticed importance of contextual information (category of items) in every of the mentioned objectives: in *IBR* the proposed items ought to belong to the same category as the items registered in users' path, whereas in *UBR* it is desirable to recommend items new and surprising from different from well known by user categories. To achieve it there were proposed two approaches: context pre-filtering module or a new similarity measure which process the context information during similarity calculation.

The new similarity measure calculation bases on common widely used similarity measures (e.g. Pearson correlation, Eculidean distance) and contextual information. It was proposed on Bialystok University of Technology and tested in what2buy system. The proposed index takes into consideration relationship among the items as well as among the categories. In both parts the input data is behavioural information from users. Equation 1 describes the measure.

$$sim_{BaseCtx}(x_i, x_j) = \alpha \cdot sim_{Base}(x_i, x_j) + \beta \cdot sim_{Base}(c_a(x_i), c_b(x_j)) \tag{1}$$

sim_{Base} denote standard similarity measure (e.g. Pearson correlation, Eculidean distance). The final similarity value depends on similarity between items: x_i and x_j and between the categories they belong to: c_a, c_b. Coefficients α and β ($\alpha \in [0, 1]$, $\beta \in [0, 1]$) adjust importance of these components. High values of this index require strong matching in part of items as well as categories. However, in case of cold start, when new items are introduced to the store's offer ($sim_{Base}(x_i, x_j) = 0$), it is possible to determine similarity basing on the relationship among the categories.

3 Experiments

The set *transSData* contains data from 14 months (see Table 1), whereas the set *visSData* consists data from 9 months (see Table 2). There were saved all transactions data, whereas the in the second set, only successful data (items from paths from the same category and from users who finally added at least one item to the basket).

First of all, it was examined effectiveness of the recommender modules on the source data with respect to RMSE error. There were examined for both modules the following similarity indices: Pearson Correlation (*Pearson*), LogLikehood Similarity (*LogLikehood*), Cosine-based Coefficient (*Cosine*), Euclidean Distance Similarity (*Euclidean*), CityBlock Measure (*CityBlock*) and respective context-based ones: Pearson Context-based Correlation (*PearsonContext*), LogLikehood Context-based Similarity (*LogLikehoodContext*), Context Cosine-based Coefficient (*CosineContext*), Euclidean Distance Context-based Similarity (*EuclideanContext*), CityBlock Context-based Measure (*CityBlock Context*). In all cases $\alpha = \beta = 0.5$). The results are presented in Table 3 for both *transSData* and *visSData* datasets (Table 4).

The lowest value of RMSE error was the case with Euclidean Distance Similarity in both modules, however the measure based on context data and Euclidean distance was the following one, slightly only worse: the difference was

Table 1 Description of transactions data *transSData*

Total number of users	Number of users versus transaction length (L)	
156844	L = {1,2}	99202
	L = [3, 10]	40438
	L $\geq$ 11	17202
Total number of items	Number of items versus transaction length (L)	
35105	L = {1,2}	12139
	L = [3, 10]	10918
	L $\geq$ 11	12047

Table 2 Description of transactions data *visSData*

Total number of users	Number of users versus path length (L)	
46691	L = {1,2}	24412
	L = [3, 10]	17868
	L $\geq$ 11	4397
Total number of items	Number of items versus path length (L)	
21489	L = {1,2}	7641
	L = [3, 10]	10327
	L $\geq$ 11	6240

Table 3 RMSE values for both modules of recommender system: *UBR* and *IBR* with common similarity measures

UBR module on *transSData* data				
Pearson	*LogLikehood*	*Cosine*	*Euclidean*	*CityBlock*
0.60	0.87	0.52	0.0	0.93
IBR module on *visSData* data				
Pearson	*LogLikehood*	*Cosine*	*Euclidean*	*CityBlock*
6.87	10.44	2.49	5.12	8.12

Table 4 RMSE values for both modules of recommender system: *UBR* and *IBR* with context similarity measures

UBR module on *transSData* data				
Pearson Context	*LogLikehood Context*	*Cosine Context*	*Euclidean Context*	*CityBlock Context*
0.44	1.04	0.44	0.59	0.98
IBR module on *visSData* data				
Pearson Context	*LogLikehood Context*	*Cosine Context*	*Euclidean Context*	*CityBlock Context*
0.60	11.33	5.58	8.88	8.63

2–3 %. It must be mentioned, that for *EuclideanContext* measure on *transSData*, but with the context information taken from *visSData*, the value of RMSE was equal 0.38. It was the lowest error value in all cases.

Very important measure in recommender systems evaluation is coverage. This index measures filling a recommendation list with required length. This is particularly important in real environment. The recommender modules with all mentioned above similarity measures were examined with respect to coverage recommendation lists of different length. For this experiment only input data of users who had 2 items in their history was taken, because they are the most common in the both datasets. The results are presented in Tables 5, 6, 7 and 8 for both *transSData* and *visSData* datasets.

In case of *UBR* module including context information slightly decreased coverage (1–2 %), however in case of *IBR* module the coverage increased significantly (10–15 %). When an item is novel or it has not many relation with the other items, it has not enough similar items on the recommendation list generated for it. Including well-established similarity among the categories increased coverage and filling the lists of propositions.

As it was mentioned before, the system is working in real environment on selling platform. It was possible to compare its factual effectiveness evaluated by real customers. It was measured a ratio of the bought items, which were recommended and presented to the customer before, to the total number of items, which were bought that day. The results are presented in Tables 9 and 10.

Table 5 Coverage values (%) for *UBR* module of the examined recommender system with common similarity measures

Required recommendation list length = 10				
Pearson	*LogLikehood*	*Cosine*	*Euclidean*	*CityBlock*
82 %	98 %	98 %	98 %	100 %
Required recommendation list length = 20				
Pearson	*LogLikehood*	*Cosine*	*Euclidean*	*CityBlock*
80 %	98 %	98 %	98 %	100 %
Required recommendation list length = 30				
Pearson	*LogLikehood*	*Cosine*	*Euclidean*	*CityBlock*
78 %	98 %	98 %	98 %	100 %

Table 6 Coverage values (%) for *UBR* module of the examined recommender system with context similarity measures

Required recommendation list length = 10				
Pearson Context	*LogLikehood Context*	*Cosine Context*	*Euclidean Context*	*CityBlock Context*
94 %	97 %	97 %	97 %	100 %
Required recommendation list length = 20				
Pearson Context	*LogLikehood Context*	*Cosine Context*	*Euclidean Context*	*CityBlock Context*
94 %	97 %	97 %	97 %	100 %
Required recommendation list length = 30				
Pearson Context	*LogLikehood Context*	*Cosine Context*	*Euclidean Context*	*CityBlock Context*
94 %	96 %	96 %	96 %	100 %

Table 7 Coverage values (%) for *IBR* module of the examined recommender system with common similarity measures

Required recommendation list length = 10				
Pearson	*LogLikehood*	*Cosine*	*Euclidean*	*CityBlock*
38 %	89 %	89 %	89 %	99 %
Required recommendation list length = 20				
Pearson	*LogLikehood*	*Cosine*	*Euclidean*	*CityBlock*
31 %	85 %	85 %	85 %	99 %
Required recommendation list length = 30				
Pearson	*LogLikehood*	*Cosine*	*Euclidean*	*CityBlock*
27 %	83 %	83 %	83 %	98 %

It can be seen, that customers the most often select items, which were recommended basing on their path in the service. Surprisingly, the recommendations, which were the most interesting for the customers, were generated by the module

Table 8 Coverage values (%) for *IBR* module of the examined recommender system with context similarity measures

Required recommendation list length = 10				
Pearson Context	*LogLikehood Context*	*Cosine Context*	*Euclidean Context*	*CityBlock Context*
97 %	99 %	99 %	99 %	99 %
Required recommendation list length = 20				
Pearson Context	*LogLikehood Context*	*Cosine Context*	*Euclidean Context*	*CityBlock Context*
97 %	99 %	99 %	99 %	99 %
Required recommendation list length = 30				
Pearson Context	*LogLikehood Context*	*Cosine Context*	*Euclidean Context*	*CityBlock Context*
95 %	98 %	98 %	98 %	98 %

Table 9 A level of customers interest (%) of presented recommendations with respect to common similarity measures

UBR module on *transSData* data				
Pearson	*LogLikehood*	*Cosine*	*Euclidean*	*CityBlock*
0.61	1.5	0.88	0.62	0.61
IBR module on *visSData* data				
Pearson	*LogLikehood*	*Cosine*	*Euclidean*	*CityBlock*
7.15	17.31	4.63	5.45	5.31

Table 10 A level of customers interest (%) of presented recommendations with respect to context similarity measures

UBR module on *transSData* data				
Pearson Context	*LogLikehood Context*	*Cosine Context*	*Euclidean Context*	*CityBlock Context*
0.61x	1.5	0.88	0.62	0.61
IBR module on *visSData* data				
Pearson Context	*LogLikehood Context*	*Cosine Context*	*Euclidean Context*	*CityBlock Context*
7.15x	17.31	4.63	5.45	5.31

IBR with LogLikehood similarity measure, which was not the best one in off-line evaluation. The following best results were generated by the system with the context similarity based on Euclidean distance.

4 Conclusions

Rapid increase in the volume of information on the internet reduces its accessibility. On the other hand, the information delivered to the user should be highly matched with their personal preferences. One of the personalisation tools are recommended systems, which identify customer tastes analysing their behaviour on the selling platform.

There are many approaches to personalization in recommender systems: content-based recommenders, collaborative filtering systems, knowledge-based technique. Real environment recommendation services are the most often complex hybrid systems composed of many different method modules. Designing an architecture of such network requires deep analysis of the domain specificity: customers behaviour, frequency of shopping, attachment to favourites products, etc. Often, a very important factor is context information, e.g. categories and dependence among them. Finally, to select and adjust the optimal approach, it also involves performing and repeating a series of experiments to evaluate overall effectiveness.

In this article a recommender system in real environment was presented. There was examined influence of context information: relationship among item's categories introduced in post-filtering module as well as in similarity measure calculation. The context similarity measures base on common indices, e.g. Euclidean distance or LogLikehood function, however take into account users interaction on products as well as on categories the products belong to. Finally, the overall effectiveness of the recommender system, which generates proposition using this measure, was better: customers more often selected the propositions, although in the experiments its RMSE value was not the lowest.

Acknowledgments This work was partially supported by Rectors of Bialystok University of Technology Grant No. S/WI/5/13.

References

1. Abbas, A., Zhang, L., and Khan, S. U.: A survey on context-aware recommender systems based on computational intelligence techniques. Computing 1–24. Springer (2015)
2. Adomavicius, G., Tuzhilin, A.: Context-aware recommender systems. In: Handbook on Recommender Systems, pp. 217–253. Springer (2011)
3. Baltrunas, L., Ludwig, B., Ricci, F.: Matrix factorization techniques for context aware recommendation. In: Fifth ACM conference on recommender systems, pp. 301–304. ACM (2011)
4. Bobadilla, J., Ortega, F., Hernando, A., Gutirrez, A.: Recommender systems survey. Knowl.-Based Syst. **46**, 109–132 (2013)
5. Braunhofer, M., Ricci, F., Lamche, B., and Wrndl, W.: A Context-aware model for proactive recommender systems in the tourism domain. In: Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct, pp. 1070–1075. ACM (2015)

6. Datta, S., Gupta, P., and Majumder, S.: SCARS: A scalable context-aware recommendation system. In: Computer, communication, control and information technology (C3IT), 2015 third international conference on, pp. 1–6. IEEE (2015)
7. Fleder, D., Hosanagar, K.: Blockbuster cultures next rise or fall: the impact of recommender systems on sales diversity. Manag. Sci. **55**(5), 697–712 (2009)
8. Gorgoglione, M., Panniello, U.: Including context in a transactional recommender system using a prefiltering approach: two real E-commerce applications. In: The 23rd IEEE international conference on advanced information networking and applications (AINA-09), pp. 667–672. IEEE (2009)
9. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. **22**(1), 5–53 (2004)
10. Hussein, T., Linder, T., Gaulke, W., Ziegler, J.: Hybreed: a software framework for developing context-aware hybrid recommender systems. User Model. User-Adap. Inter. **24**(1–2), 121–174 (2014)
11. Jannach, D. et al: Recommender Systems: An Introduction. Cambridge University Press (2010)
12. Liu, H., Zhang, H., Hui, K., He, H.: Overview of context-aware recommender system research. In: 3rd International Conference on Mechatronics, Robotics and Automation (ICMRA 2015), pp. 1218–1221. Atlantis Press (2015)
13. Panniello, U., Tuzhilin, A., Gorgoglione, M.: Comparing context-aware recommender systems in terms of accuracy and diversity. User Model. User-Adap. Inter. **24**(1–2), 35–65 (2014)
14. Ricci, F. et al: Recommender Systems Handbook. Springer (2010)
15. Shani, G., Gunawardana, A.: Evaluating recommendation systems. In: Ricci et al. Recommender Systems Handbook, pp. 257–297. Springer (2011)
16. Zhong, E., Fan, W., Yang, Q.: Contextual collaborative filtering via hierarchical matrix factorization. In: Proceedings of the SIAM International Conference on Data Mining, pp. 744–755 (2012)

The Concept of the Effective Multi-channel CSMA/CA Detector

Dariusz Laskowski, Marcin Pólkowski and Piotr Lubkowski

Abstract The increasing demand for Internet Protocol (IP) wireless networks with increased capacity forced work on multi-channel radio devices. Along with an increasing number of wireless devices working simultaneously on similar frequencies, there is a problem with detecting the signals generated by wireless devices belonging to one system. To this end, dealt with the analysis of the available methods for the correct sensing of signals, which belong to one mobile radio system, using one of the currently most popular multi-channel modulation—OFDM (Orthogonal Frequency-Division Multiplexing). Paper describes preliminary analysis and implementation of effective multi-channel detector (EMCD), which purpose was to detect the signals coming from the nodes be-longing to one radio network. Future model will be built in Matlab environment, allowing for very accurate simulation of even very intricate problems.

Keywords OFDM · CSMA/CA · Multi-channel

1 Introduction

A MANET (Mobile Ad hoc NETwork) devices operating in continuous mode (as opposed to Wireless Sensor Networks—WSN) usually apply the CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance) access method. In the

D. Laskowski (✉)
Military University of Technology, Gen. S. Kaliskiego 2 Street,
00-908 Warsaw, Poland
e-mail: dariusz.laskowski@wat.edu.pl

M. Pólkowski
Transbit Sp. z o. o, Łukasza Drewny 80 Street, 02-968 Warsaw, Poland
e-mail: marcin.polkowski@transbit.com.pl

P. Lubkowski
Military University of Technology, Gen. S. Kaliskiego 2 Street, 02-908 Warsaw, Poland
e-mail: piotr.lubkowski@wat.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_28

classical method of CSMA/CA, broadcasting station checks the state of the radio link as soon as created is radio frame. If radio channel is free, the station begins broadcasting frames, and, in the case of busy channel, a transmission is paused until the medium is released. In order to detect a collision or other transmission errors, the receiving station must send a confirmation of reception of broadcasting frames. The CSMA/CA protocol allows detection of a collision, where the two stations will transmit at the same time or almost simultaneously after a period, a medium was busy. Another possible cause of the collision is promotional assessment error that causes a delay of the link status (station begins broadcasting, because it is not receiving signals from other nodes). Improvement of this method is to add the RTS/CTS mechanism (request to send/clear to send), which removes the hidden node problem.

The mentioned method is ideal for devices operating in the 802.11 standards family (Wi-Fi). The network made up of these devices does not typically have a large number of nodes, so they required low capacity medium. In the case of higher capacity, a large number of collision forces the use of greater number of channels at the same time. Solutions of this type, utilizing a significant number of nodes used are for example in the armed forces [1–3].

Modern multi-channel nodes of MANET network facing the problem of correct detection of the signals belonging to its own system, while maintaining minimum power consumption of end users. This article describes a proposal for efficient multi-channel detector for transmissions originating from the nodes belonging to one radio network.

The remaining part of the paper organized is as follows: Sect. 2 describes the assumption made for implementation of proposed EMCD solution. In Sect. 3 theoretical background of the considered problem is given. Scheme of the simulator is presented in Sect. 4. Section 5 presents some conclusions concerning the proposed EMCD.

2 The Initial Assumptions

Although in the literature many solutions have been proposed to solve the problem of correct carrier and sub-carrier allocation in own system [4–6] they are not well suited to the large variation of MANET solutions that use multi-channel nodes. For a future implementation of the detector model in simulation environment, the following assumptions have been taken:

- The perspective radio will work in n parallel receiving channels, covering selected subranges of frequency band. Receivers will run under control system based on allocated radio resources, received from a frequency broker or a service provider.

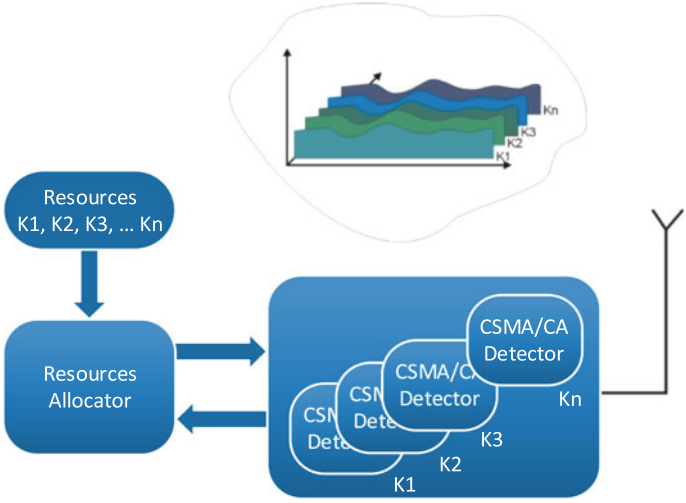


Fig. 1 Multi-channel radio system model based on multiple parallel detectors

- The bandwidth each channel can be dedicated to receive one or more of the allocated frequencies (Fig. 1). Channels can cover the entire bandwidth or unused leave a few sub-ranges (depending on frequency allocation).
- For each channel there is a possibility to start the machine of fast browsing and the detecting the allocated frequencies (if there is more than one allocated frequency with respect to the receiving path).

The above assumptions simplify model EMCD CSMA/CA to:

- N parallel CSMA/CA detectors working independently;
- One CSMA/CA detector fitted into the system's horizon energy detector (t_{se})

The proposed simulator is provided for the solutions with fast searching radio channels. Due to specificity of the radio system, it is proposed to use a hybrid detector, consisting of two main elements:

- The energy detector;
- The correlative detector (Fig. 2)

Individual detectors perform different roles in the model. The energy detector can detect signals without the need to have knowledge of their nature, frequency and bandwidth. The energy detector confirms the hypothesis of the presence of signals from other users. Its purpose consists merely of checking whether the power level of the received radio channel is greater than a threshold value. The correct operation of the radio network based on CSMA/CA algorithm, requires detection of own transmissions, which means work in the same radio network. For this purpose an additional detector based on signals characteristics contained in the preamble

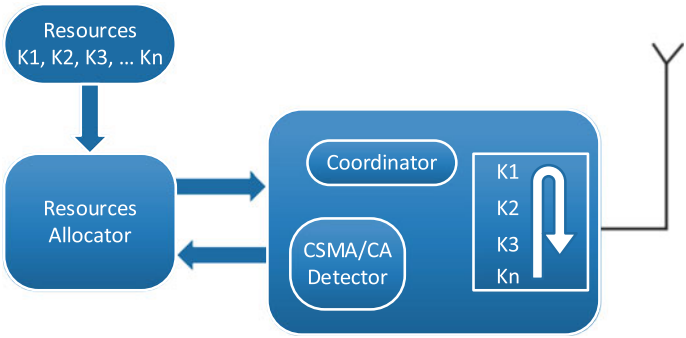


Fig. 2 The working mechanism of single receiver with fast channels scanning

was used. The combination of these detectors enables the classification of received signals.

The simulator will use the OFDM modulation, which main features such as mentioned below:

- efficient use of bandwidth;
- real time control of the radio channel parameters and tuning of the receiver;
- high resistance to multipath fading and interference between symbols;
- low sensitivity to the inaccuracy of the symbol's synchronization; predispose it to mobile applications.

3 Theoretical Background

Orthogonal frequency-division multiplexing (OFDM) is a method of encoding digital data on multiple carrier frequencies [7]. OFDM is a frequency-division multiplexing (FDM) scheme used as a digital multi-carrier modulation method. A large number of closely spaced orthogonal sub-carrier signals are used to carry data on several parallel data streams or channels. Each sub-carrier is modulated with a conventional modulation scheme (such as quadrature amplitude modulation or phase-shift keying) at a low symbol rate, maintaining total data rates similar to conventional single-carrier modulation schemes in the same bandwidth. OFDM symbol on carrier wave is expressed by:

$$g(t) = \mathcal{R}\{x(t)e^{j2\pi f_c t}\} \tag{1}$$

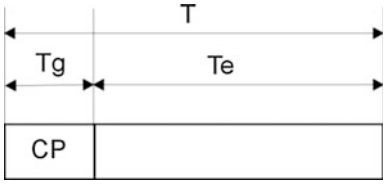


Fig. 3 The OFDM symbol

where OFDM symbol x(t) is expressed by:

$$x(t) = \sum_{s=-\infty}^{\infty} x_s(t - sT) \tag{2}$$

A single OFDM symbol in the baseband is expressed by:

$$x_s(t) = \sum_{n=-N/2}^{N/2} x_{n,s} e^{j2\pi n \Delta f (t - T_g)} \tag{3}$$

where:

$X_{n,s}$ Complex modulation symbol (ex. QPSK) placed on given subcarrier n at symbol s;
N Number of subcarriers;
$\Delta f = 1/T_e$ Subcarrier separation;
T_e Effective OFDM symbol duration;
T_g CP duration (Cyclic Prefix);

A single OFDM symbol is composed of the field CP, and the user data. Its structure is demonstrated below at Fig. 3 while table (Table 1) presents the basic values of OFDM symbol parameters.

In real radio devices it is necessary to apply a filter to reduce signal leaks (side-lobe) in adjacent channels. Therefore, it is important to exclude extreme subcarriers, because they are affected by the characteristics of the filter. The OFDM signals in the frequency domain with an emphasis on the role of the subcarrier present in Fig. 4.

- Useful subcarriers—subcarriers which carry user data;
- Pilots—subcarriers used to channel estimation and synchronization;
- Zero subcarriers—subcarriers that are not use: null subcarrier (DC) and protective subcarriers (Fig. 5).

Table 1 Basic values of OFDM symbol parameters

Parameter	Value
FFT size, N	256
Tg	16 μs
Te	64 μs

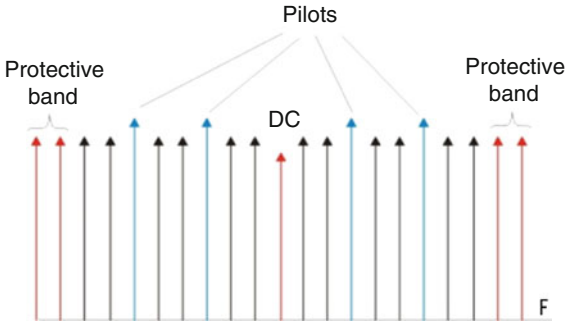


Fig. 4 An OFDM signal representation in frequency domain

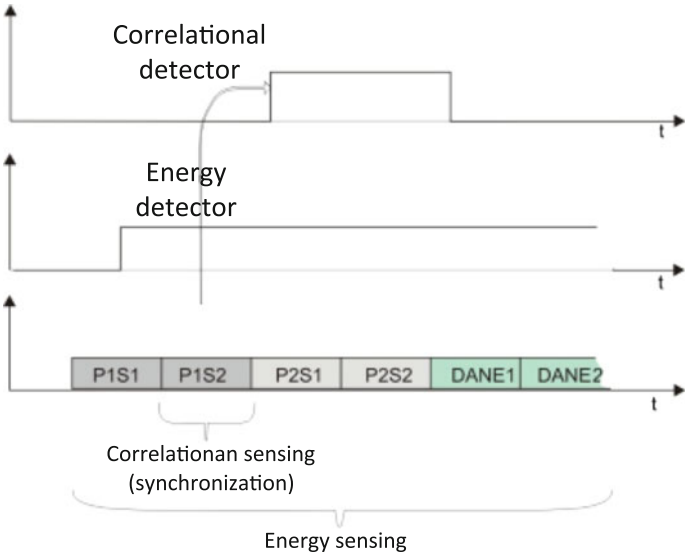


Fig. 5 The energy and correlation detector—sequence of operation

The following table shows the values of the parameters of the OFDM signal in the frequency domain, which have been implemented in the model. Frame structure is shown in the Fig. 6. Every frame is made up of four symbols of OFDM, forming a block of the preamble and the symbols that make up the data block. The symbols P1S1, P2S1, P1S2 and P2S2 form a preamble Symbols P1S1 and P1S2 have the same sign, which is used to track the signal stability. P1S1 forces the activation of the automatic gain control (AGC) to avoid interference and optimize the signal level.

The P1S2 symbol is used by the correlation detector for detecting the signal from its own radio network and synchronization the received frame (Fig. 5) (Table 2)

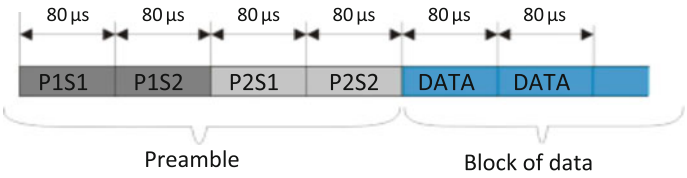


Fig. 6 The frame structure

Table 2 The value of OFDM parameters

Parameter	Value
FFT size, N	256
Number of useful subcarriers	200
Number of subcarriers in protective band	56
Number of pilots	8
Number of useful subcarriers without pilots	192

4 The Simulator Proposition

The sensing system consists of two elements, energy detector and correlative detector. Due to the foundation of the application of OFDM modulation, the primary task of correlative detector is detection the OFDM signal as a signal to other radio stations of its own network (Fig. 7).

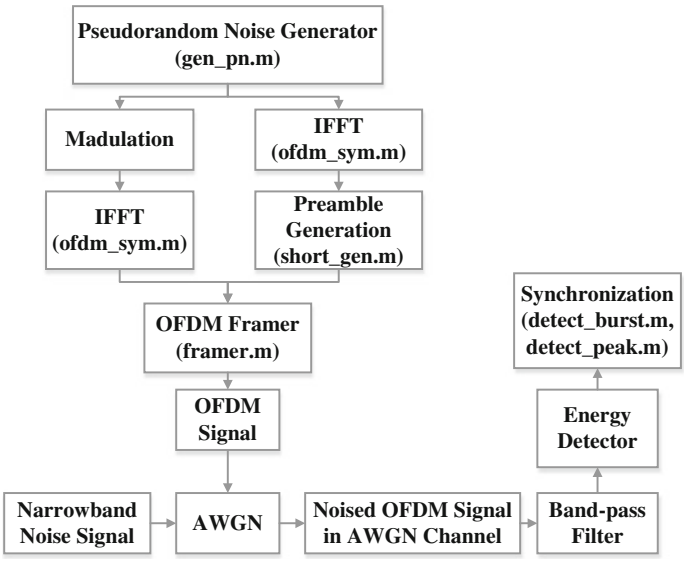


Fig. 7 Simulator block diagram

Simulation of detection of OFDM signal has been implemented in the Matlab R2014b. It is consists of three major modules: OFDM signal generation logic, and then his abnormal and distortion of the radio channel, the channel busy detection using energy detector and detection of signals from own network (receiver input signal synchronization). Detailed diagram of the simulator is set out below.

These modules are described in the following script (M-files, *.m):

• detect_burst.m	–	preamble detection,
• detect_peak.m	–	correlative impulse detection for synchronisation,
• framer.m	–	forming OFDM frame,
• gen_pn.m	–	generation pseudorandom numbers,
• main.m	–	main script,
• mapping.m	–	mapping pseudorandom sequence on subcarriers,
• ofdm_sym.m	–	Inverse Fast Fourier Transformation,
• shortp_gen.m	–	preamble generation,
• shortp_gen_idx.m	–	preamble indexing,
• ofdm_syst.m	–	OFDM symbol structure indexing.

5 Conclusions

Own signals detection in modern mobile radio systems is becoming an increasingly important issue. The article proposes a way of detecting signals from other stations of the radio network using the layout of the detectors, consisting of energy detector and the correlative detector. Energy detector detects the initial signal. On the other hand correlative detector fined the signal belonging to the network and synchronizes receiver with this signal. Hybrid construction of the detector allows limiting the use of hardware resources of the receiving equipment over signals detecting from the devices belonging to a single radio system and exclude unnecessary analysis of the parameters of the signals received by a multichannel receiver.

Currently undergo work is aimed at implementation of the proposed detector in the test environment [8] using Matlab simulation tool to test the effectiveness of this type of detection.

References

1. Fossa, C.E., Macdonald, T.G.: Internetworking Tactical MANETs, Military Communications Conference, MILCOM 2010, November 2009
2. Lubkowski, P. et. al.: On improving connectivity and network efficiency in a heterogeneous military environment. In: Proceedings of the International Conference on Military Communications and Information Systems (ICMCIS 2015), May (2015)

3. Hasan, Z., Bansal, G., Hossain, E., Bhargava, V.: Energy-efficient power allocation in OFDM-based cognitive radio systems: a risk-return model. IEEE Trans. Wireless Commun. **8**, 6078–6088 (2009)
4. Duval, O., Hasan, Z., Hossain, E., Gagnon, F., Bhargava, V.K.: Subcarrier selection and power allocation for amplify-and-forward relaying over OFDM links. IEEE Trans. Wireless Commun. **9**(4), 1293–1297 (2010)
5. Zhidkov, S.V.: Impulsive noise suppression in OFDM based communication systems. IEEE Trans. Consum. Electron. **49**(November (4)), 944–948 (2003)
6. Bansal, A., Kohli, A.K.: Suppression of impulsive noise in OFDM system using imperfect channel state information. Optik—Int. J. Light Electron Opt. **127**(4), 2111–2115 (2016)
7. Ghosh, M.: Analysis of the effect of impulse noise on multicarrier and single carrier QAM systems. IEEE Trans. Commun. **44**(February (2)), 145–147 (1996)
8. Lubkowki, P., Laskowski, D.: Test of the multimedia services implementation in information and communication networks. In: Advances in Intelligent Systems and Computing, vol. 286, pp. 325–332, Springer International Publishing AG, Switzerland (2014)

Clustering Context Items into User Trust Levels

Paweł Lubomski and Henryk Krawczyk

Abstract An innovative trust-based security model for Internet systems is proposed. The TCoRBAC model operates on user profiles built on the history of user with system interaction in conjunction with multi-dimensional context information. There is proposed a method of transforming the high number of possible context value variants into several user trust levels. The transformation implements Hierarchical Agglomerative Clustering strategy. Based on the user's current trust level there are extra security mechanisms fired, or not. This approach allows you to reduce the negative effects on the system performance introduced by the security layer without any noticeable decrease in the system security level. There are also some results of such an analysis made on the Gdańsk University of Technology central system discussed.

Keywords Context · System security · Clustering · TCoRBAC model · Trust levels

1 Introduction

Internet systems are the most commonly used types of systems nowadays. Their popularity results from the ease of their use. They are accessible from nearly all types of computers—from traditional PCs to the more and more popular mobile devices, such as tablets and smartphones. It does not matter what operating system is used, because they are used with web browsers which implement the same standards of HTML, JavaScript, AJAX, etc. They are also accessible from any

P. Lubomski (✉)
IT Services Centre, Gdańsk University of Technology, Gdańsk, Poland
e-mail: lubomski@pg.gda.pl

H. Krawczyk
Faculty of Electronics, Telecommunications and Informatics,
Gdańsk University of Technology, Gdańsk, Poland
e-mail: hkrawk@eti.pg.gda.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_29

place where the Internet is available. So nearly everyone can try to access them. On the other hand, such openness poses a serious threat to their security. Also, their distributed and service-oriented architecture has a significant impact on their security and reliability.

The technology evolves very quickly. Every 5 years there is a major step made—a small revolution. The same applies to the area of security. There were solutions based on traditional MAC (Mandatory Access Control), DAC (Discretionary Access Control) and RBAC (Role Based Access Control) models [1, 2]. Then, they were expanded by using elements of dynamicity—they adaptively expand and shrink users' permission sets [3]. Finally, the analysis of widely understood context was involved [4–7]. The expansion of the idea of "big data" created a new place for security mechanisms. They started to profile users' behavior, and decide upon access on the basis of these profiles [8–10]. In such a way we evaluate from strictly defined access control rules to user trust-based mechanisms—the decision of granting or denying access is based on the system-to-user trust [4, 11–13].

There is also another important matter connected with the security of Internet systems. There should be a balance between the security level and the usability and efficiency of the system [14]. From the system point of view, the security mechanisms should provide an acceptable level of risk, but from the user's point of view the security layer should be nearly invisible as long as the user behaves in a predictable way [15].

There are two main problems in building such efficient security mechanisms. The first one is the complexity of context and its analysis. The context nature is multidimensional. That results in high complexity. Previously we proposed the universal CoRBAC (Context-oriented Role Based Access Control) model and the security audit-based procedure of determining the system security level [6, 16, 17]. The other problem is that strong security mechanisms have a negative impact on the system performance. The remedy for this problem is building some correct and efficient user trust estimation procedures and forcing strong security mechanisms only on low user trust levels. This is the subject of this paper.

2 User Trust Model for Internet Systems

Users behave in a predictable way. The same applies to their interaction with systems. For example, they usually use the same devices, the interaction takes place in the same hours and from the same place. This is very unique to each user and can be treated as the user's fingerprint. As mentioned earlier, the user interaction with the system is accompanied by many context parameters which describe the current overall context of each activity. A written log of such activities, with accompanying contexts, creates an individual user profile. On the basis of such a profile, the system can determine if the user behaves typically, or not. That determines the system-to-user trust levels. We proposed the model with different security

mechanisms which depend on the user trust level [13]. Now we want to deal with the problem of determining the correct amount of user trust levels.

Let us define context (C) as a set of context parameters (CP_i):

$$c_x \in C = CP_1 \times CP_2 \times \ldots \times CP_z \quad (1)$$

Each context parameter (CP_i) is a finite, discrete set of possible values of the parameter:

$$CP_i = \{cp_{i1}, cp_{i2}, \ldots, cp_{in}\} \quad (2)$$

Some context parameters have a continuous character (e.g. time). During the context analysis they are clustered into certain groups, e.g. days of the week. The same situation applies to context parameters with discrete but numerous values, e.g. set of IP addresses. They are clustered into subsets of a specific netmask.

The current context (c_x) accompanying each user with the system interaction is described by the current values of each context parameter (CP_i). Thus, the cardinality of the set of context items is the multiplication of cardinality of each context parameter values set.

Let us analyze an example of a context consisting of three context parameters:

- CP_1—logical localization of user—a set of the following elements: cp_{11} = internal network, cp_{12} = campus network, cp_{13} = external network (Internet),
- CP_2—time of user's activity in the system—a set of elements: cp_{21} = weekday, cp_{22} = weekend,
- CP_3—type of device being used—a set of elements: cp_{31} = traditional PC, cp_{32} = mobile device.

Then the $|C| = 12$. Each context is a three-value tuple, e.g. $c_1 = (cp_{11}, cp_{21}, cp_{31})$. When any of the context parameter changes the value, the current context also changes. For example when the user changes the type of the device used, the accompanying context also changes.

We can create a bi-directional full graph of items transitions—see Fig. 1. The vertexes of this graph are context items (c_i). The edges of the graph describe the changes of the context accompanying user requests. A history of such a user with system interaction with accompanying context value creates a user's profile. Each of the vertexes are also marked by a number determining how often the user interacted in this context (see Fig. 2a). We can cluster vertexes of this graph into groups on the basis of this metric (*f*). The resulting groups form user trust levels, on the basis of which the appropriate security mechanisms are fired.

It is worth mentioning some facts. Firstly, the multiplicity of context parameters results in time-consuming procedures of determining their values. Secondly, the number of items in the graph (context items) is very numerous and grows exponentially. Thus, it is hard to analyze such a big amount of data, which again has a negative impact on the system performance.

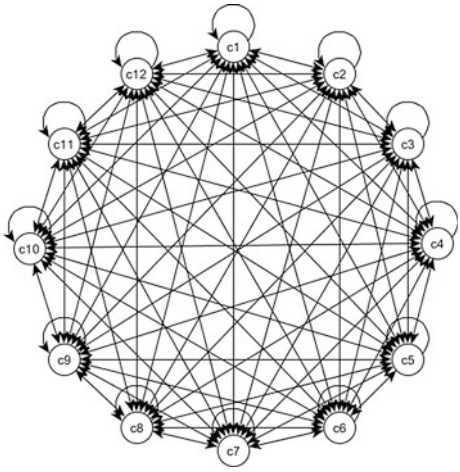


Fig. 1 Example bi-directional graph of possible user context items transition

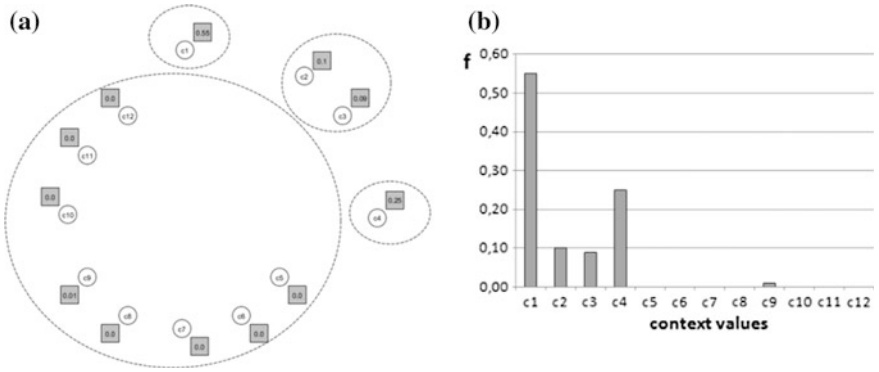


Fig. 2 **a** Vertexes are marked by a number determining how often the user interacted in this context and clustered into 4 groups. **b** Histogram of frequency metric *f* which determines how often the user interacted in the specific context values

3 Classification Algorithm for Determining Categories in User Trust Model

Each user profile can be represented as a histogram of the previously mentioned frequency metric for each context value. Let us analyze the example graph of context items presented in Fig. 2a. Each vertex has a frequency metric *f* assigned which determines how often the user interacted in this context. An example histogram has been depictured in Fig. 2b. Of course, in the real world the context is much more complicated, multi-dimensional (consists of many more context

parameters), and the context parameters have more different values. So the effective context items set is very numerous.

There is a need to cluster the frequencies from the histogram into groups. For these groups some appropriate trust levels are assigned. Each trust level is associated with a corresponding security mechanism. The security mechanisms may include the following: retyping captcha-code, entering the user password again, inputting a special code sent by SMS or e-mail, etc.

The best way to determine the number of the clusters, and to do the clustering, is by using Hierarchical Agglomerative Clustering (HAC) [18–20]. As an input for this algorithm there is a vector of frequency metrics from the user's profile used. Figure 3a presents an example dendrogram as a result of such clustering (all HAC analysis were made with the use of P. Wessa software [21]). There is the Euclidean distance metric used, and the Centroid Criterion strategy of computing the distance between the clusters. Figure 3b presents example distribution of the number of clusters and the distance between the closest clusters (also known as the height of dendrogram).

On the basis of this distribution it can be determined when the distance stops to decrease significantly—this is the most optimal number of clusters. After this point the differences between clusters are so tiny that it is pointless to extract any additional trust levels, and corresponding security mechanisms. Of course, this number is also dependant on the characteristics of the system, and the number of reasonable security mechanisms that may be introduced.

The final step is to determine the ranges of frequency metrics corresponding with each cluster, sort them ascending, and assign them trust levels with security mechanisms. This way we reduce the number of items (context values) from very numerous to four, in our case. This number results from the analysis of our university central system logs.

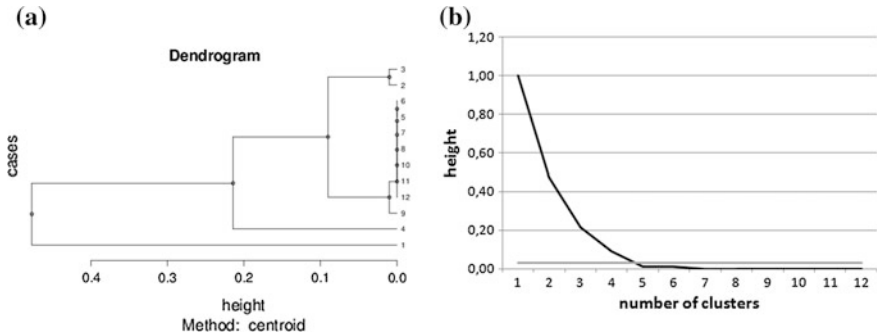


Fig. 3 **a** Example dendrogram as a result of HAC. **b** Example distribution of the number of clusters and the distance between the closest clusters (height)

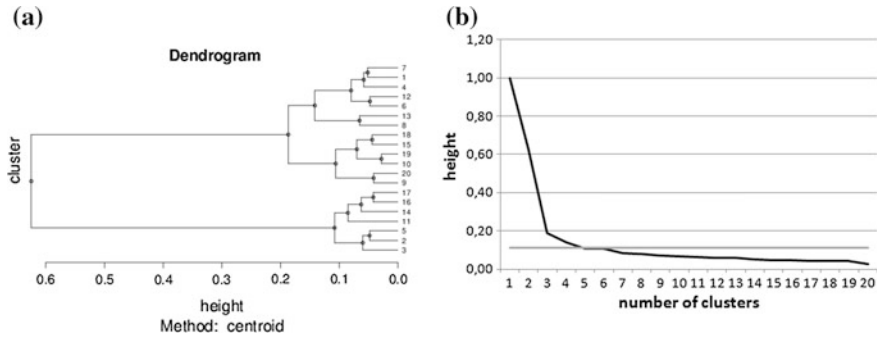


Fig. 4 **a** Dendrogram and **b** distribution of the number of clusters and the distance between the closest clusters (height) in our university case

Figure 4 presents the dendrogram and distribution of the number of clusters and the distance between the closest clusters in the case of our university system. The analysis is taken on a population of 3,000 users. The results are presented for the top 20 clusters cut—the subsequent clusters have small distances (near zero) and can be omitted, they would result only in the illegibility of the charts. The system is used by over 40,000 users. The analysis takes into account a nearly 4-year period of the system's operation.

There is another approach—with the use of unsupervised K-means strategy [22]. The initial number of clusters may be determined using a Mean shift algorithm [23]. Next, there should be the results of determining the average distance from the centroid for each cluster (treated as imperfection of clustering) compared. The analysis should start with the number of clusters determined by the Mean shift algorithm. Then, there should be the nearby numbers of clusters checked too. Finally, the best number of clusters is when the imperfection metric stops to decrease significantly (similarly to HAC). In our case this strategy gives some similar results to HAC strategy, but it is less deterministic. Thus, in our opinion, HAC strategy seems to be a better approach to items reduction and transformation to trust levels.

4 Application of Trust Model in TCoRBAC Approach

The above trust level determination mechanism was applied to a TCoRBAC (Trust- and Context-oriented Role Based Access Control) model [13, 17]. The main goal of this approach is to build separate profiles of each user's interaction with the system. Using the strategy described in this paper of the items' reduction, and assigning appropriate user trust levels, it is possible to fire some extra security mechanisms. The algorithm of authorization and trust verification in the TCoRBAC model is presented as the pseudo-code below.

```
foreach (userReq) do:
  c_x = determine current context;
  authResult = do the basic permission check;
  if(authResult == true)
  then
    profile = compute current profile of user(userReq);
    clusters = do the HAC clustering (profile);
    trustLvl = determine cluster containing current con-
               text c_x of userReq;
    result = enforce extra security mechanism (trustLvl);
    if(result satisfies security policy)
    then grant access;
    else deny access;
  else deny access;
```

The result of clustering the example items transition graph (mentioned at the beginning of the paper) is depicted in Fig. 5. There are four clustered items with the appropriate trust levels assigned. A user interacts in some context which changes while changing one of its context parameter values. This may cause a change of the trust level and adequate security mechanism.

The security mechanisms connected with trust levels may be as follows:

- TL4—fully trusted user—there is no extra security action (cost: 0 s)
- TL3—confirmation of user identity by pointing to 1 of 10 images (cost: 1 s)
- TL2—re-entering the password or other personal data (cost: 3 s)
- TL1—inputting the one-time-password sent by SMS (cost: 5 s)

Table 1 presents an example simple scenario of user interaction with the university system. There is the time spent on each action (from the user's point of view) indicated, as well as the type of action: business or security.

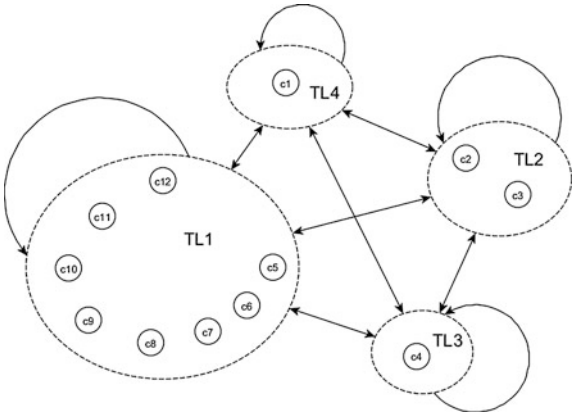


Fig. 5 Items transition graph as a result of clustering procedure and trust level assignment

Table 1 Example simple scenario of user interaction with the university system

No	Activity	Time spent by user	Type of activity
1	Login	3 s	τ_S
2	Computation of current context	≈0 s	τ_S
3	Permission check	≈0 s	τ_S
4	Determination of trust level	≈0 s	τ_S
5	Execution of security mechanism	0/1/3/5 s	τ_S
6	"Student" application choice	1 s	τ_B
7	Selection of subject	1 s	τ_B
8	Grade reading	1 s	τ_B
9	Logout	1 s	τ_S

On the basis of this example scenario there is calculated the user-side efficiency metric (ε) defined as the ratio of the time spent on business activities (τ_B) to the overall time spent on business and security activities ($\tau_B + \tau_S$):

$$\varepsilon = \tau_B/(\tau_B + \tau_S) \tag{3}$$

The resulting efficiency metric varies from 0.25 to 0.4286, depending on which security mechanism is fired. It has a nearly double difference, which is noticeable for the user. Of course, in more complex scenarios this metric will decrease. It is worth mentioning that users mostly interact in TL4, so then the efficiency metric is from the top of the range. The metric decreases in lower trust levels, but it is the result of a higher security level.

The TCoRBAC model was developed for the central system of Gdańsk University of Technology. More information about this model can be found at [6, 13, 16, 17].

5 Conclusions

It is worth pointing out that the described approach is very effective in complicated multi-dimensional cases. The process of clustering is not very efficient, but it may be done asynchronously in the background. There may be one more optimization done: the clustering and profile computation may be actualized periodically—this also reduces the load of the system.

The reduction of many context items to a few trust levels does not decrease the system security level significantly and does not have an impact on the user-side efficiency metric (ε). With reference to Furnell's insights [14], the security level is nearly invisible for the users who do not change their behavior—they are classified at the highest trust level, so there is no need to do any extra security checks.

On the other hand, when the user changes the context very often, s/he gets a regular distribution of frequency metric *f*, so that it is hard to cluster their profile. In such a case it is worth considering doing global (per whole system) clustering and determining global ranges of this metric which cannot be exceeded during per-user clustering and trust level assigning.

The clustering process may also take into account a weight metric pointing to the distance between the context values (computed on the basis of how many context parameter values change from one context value to another). It is a quite intuitive approach but also causes some blur of clusters and decrease of the system security level. In such a case an intruder may behave quite similarly to the real user (use similar context values) and will not be detected. If we use strict context clustering (with equal weight = 1) the intruder has to behave exactly the same as the user (the same context values) in order not to be detected. In the real world, where many context parameters are taken into account, it is very hard to copy the user's behavior precisely.

References

1. Benantar, M.: Access control systems. In: Security, Identity Management and Trust Models. Springer (2006)
2. Bertino, E.: RBAC models—concepts and trends. Comput. Secur. **22**(6), 511–514 (2003)
3. Ricci, A., Viroli, M., Omicini, A.: An RBAC approach for securing access control in a mas coordination infrastructure. In: 1st International Workshop Safety and Security in MultiAgent Systems (SASEMAS 2004), pp. 110–124 (2004)
4. Bhatti, R., Bertino, E., Ghafoor, A.: A trust-based context-aware access control model for web-services. Distrib. Parallel Databases **18**(1), 83–105 (2005)
5. Khan, M.F.F., Sakamura, K.: Context-aware access control for clinical information systems. In: 2012 International Conference on Innovations in Information Technology (IIT), pp. 123–128 (2012)
6. Krawczyk, H., Lubomski, P.: CoRBAC—context-oriented security model (in Polish). Studia Informatica **34**(3), 185–194 (2013)
7. Huang, X., Wang, H., Chen, Z., Lin, J.: A context, rule and role-based access control model in enterprise pervasive computing environment. In: 2006 First International Symposium on Pervasive Computing and Applications, pp. 497–502 (2006)
8. Miettinen, M., Asokan, N.: Towards security policy decisions based on context profiling. In: Proceedings of the 3rd ACM Workshop on Artificial Intelligence and Security—AISec'10, p. 19 (2010)
9. Gupta, A., Miettinen, M., Asokan, N., Nagy, M.: Intuitive security policy configuration in mobile devices using context profiling. In: 2012 International Conference on Privacy, Security, Risk and Trust and 2012 International Conference on Social Computing, pp. 471–480 (2012)
10. Manikopoulos, C., Papavassiliou, S.: Network intrusion and fault detection: a statistical anomaly approach. IEEE Commun. Mag. **40**(October), 76–82 (2002)
11. De Capitani Di, S., Vimercati, S., Foresti, S., Jajodia, S., Paraboschi, G.Psaila, Samarati, P.: Integrating trust management and access control in data-intensive Web applications. ACM Trans. Web **6**(2), 1–43 (2012)
12. Woo, J.W., Hwang, M.J., Lee, C.G., Youn, H.Y.: Dynamic role-based access control with trust-satisfaction and reputation for multi-agent system. In: 2010 IEEE 24th International

Conference on Advanced Information Networking and Applications Workshops, pp. 1121–1126 (2010)
13. Krawczyk, H., Lubomski, P.: User trust levels and their impact on system security and usability. In: Communications in Computer and Information Science. Springer International Publishing, pp. 82–91 (2015)
14. Furnell, S.: Usability versus complexity—striking the balance in end-user security. Netw. Secur. **2010**(12), 13–17 (2010)
15. Pahnila, S.P.S., Siponen, M.S.M., Mahmood, A.M.A.: Employees' behavior towards IS security policy compliance. In: 2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07) (2007)
16. Lubomski, P.: Context in security of distributed e-service environments. In: Proceedings of the Chip to Cloud Security Forum 2014, p. 18 (2014)
17. Lubomski, P., Krawczyk, H.: Practical evaluation of security mechanisms of Internet systems (in review). IEEE Secur. Privacy Mag.
18. Adams, R.P.: Hierarchical Agglomerative Clustering (2016)
19. Borgatti, S.P.: How to explain hierarchical clustering. Connections **17**(2), 78–80 (1994)
20. Bouguettaya, A., Yu, Q., Liu, X., Zhou, X., Song, A.: Efficient agglomerative hierarchical clustering. Expert Syst. Appl. **42**(5), 2785–2797 (2015)
21. Wessa, P.: Free statistics software, office for research development and education version 1.1.23-r7, 2016. http://www.wessa.net/
22. Hartigan, J.A., Wong, M.A.: A K-Means clustering algorithm. J. Roy. Stat. Soc. **28**(1), 100–108 (1979)
23. Comaniciu, D., Meer, P.: Mean shift analysis and applications. In: Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 2, pp. 1197–1203 (1999)

Dependability Metrics for Network Systems—Analytical and Experimental Analysis

Jacek Mazurkiewicz

Abstract Paper presents several dependability metrics for network systems. Methods of metric calculation are also taken into consideration. Two general groups of metrics are analyzed: analytical and experimental. Following analytical metrics are analyzed: network cohesion in the system, number of services involved in the compound service, absolute importance, dependence and criticality of the service and overall reliability of the compound service. Experimental metrics—task response time and service component availability—are also calculated. Numerical experiments were performed on a test case scenario. The dependability means the combination of the reliability and functional parameters of the network. Analysis is done with a usage of open-source simulation environment that can be easily modified and extended for further work. Based on the simulation results, some alternatives can be chosen in case of system or service failure. This way it is possible to operate with large and complex networks described by various—not only classic—distributions and set of parameters. The presented problem is practically essential for organization of network systems.

Keywords Network systems · Reliability · Dependability modeling

1 Introduction

In today's business and service-based information systems knowledge about measurements and metrics of system parts is not only connected with better understanding these systems, but also as a set of means about business values. No, wander then, that this approach is still widely used in almost all present-day domains and primarily has been adapted to industry as well as hi-tech field.

J. Mazurkiewicz (✉)
Department of Computer Engineering, Faculty of Electronics,
Wroclaw University of Science and Technology, Wybrzeze Wyspianskiego 27,
50-370 Wroclaw, Poland
e-mail: Jacek.Mazurkiewicz@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_30

Furthermore, measurement started to benefit from as basis for control systems, which at present, often are compound computer aided platforms for managing other systems.

In computer science, metrics are common and widely used, still in case of service-based information systems this area is still under development [9, 11, 12, 16]. Researchers are trying to invent or improve metrics in case of better system evaluation. Such situation can be especially seen in the dependability area. In this paper, we present some of the metrics focusing on two (analytical and experimental) methods of their calculation. As a result, we try to look to a comparison of their results and answer the question about their utilization in a service-based information system area.

Since Information Systems must be proved experimentally on either real systems or simulations based on traces from real systems in Sect. 2 we present short description of simulation as a methods of system analysis. Next, we propose some defined dependability metrics (Sect. 3) that will be used in examined exemplary case scenario (Sect. 4). Finally, there are conclusions and plans for further work.

2 Simulation Tools

Simulation is an attempt to model a real-life or hypothetical situation on a computer so that it can be studied to see how the system works [9, 10]. Going further, we can say that simulator is a tool that tries to mimic the behaviour of a system. In literature, two main types of simulators can be found: a continuous time and discrete event based [10]. In the area of simulation issues, there main tasks/processes can be named:

- designing of simulation model,
- calculation of the model (using simulation algorithms),
- results analysis.

Simulation of a physical reality requires creating mathematical model that represents its nature. It can be represented in various ways starting from declarative form, functional, spatial and multi-model. After proposing suitable form of the representation simulation can be done using one or more units (depends of a dispersion of large-scale models). It should be noted, that simulation can be done on various levels of abstraction considering user needs, and data needed for results. The more precise models, the more precise results that is way a suitable model should be chosen from a set of models in a set of: stochastic (using random numbers generators); deterministic (continuous time or discrete event based); distributed (realized as a network) and agent based (agents are represented as a part of the system working as a executable threads).

2.1 MS Classification

For modeling and simulation technique (*MS*) [10] large number of tools are available. Moreover this number is still growing, since the requirements become larger and larger in all four groups of its main classification, that is:

1. Analytical tools—help to design a network model and calculate different factors (e.g. reliability) of the model.
2. Simulation tools—simulate dynamic behaviors (e.g. link failure) of a network model besides modeling.
3. Topology discovery tools—extract the actual network information from an existing system and map them graphically and/or in text format
4. Topology generation tools—help to generate small as well as large topologies based on different algorithms.

Figure 1 [10] shows further classification of the tools with respect to their commercial and educational classification.

2.2 Tools for Network Systems Simulation

There are various methods of system analysis. Some researches try to do it using graphs [5], others choose some simulation techniques. In this paper we consider system behavior that will be as close to reality as is can be. Usage of simulator allow us to mimic the behavior of a system. In literature, two main types of simulators can be found: a continuous time and discrete event based simulation [6, 14]. Continuous simulation requires a representation of the system using differential equations [5]. This type of simulator is predominately related with electric power studies. For this reason it will be excluded from further research. The other group of simulators are discrete events that describe the system behavior as a series of events. Classically discrete event simulators are basis for telecommunication and IT analysis tools.

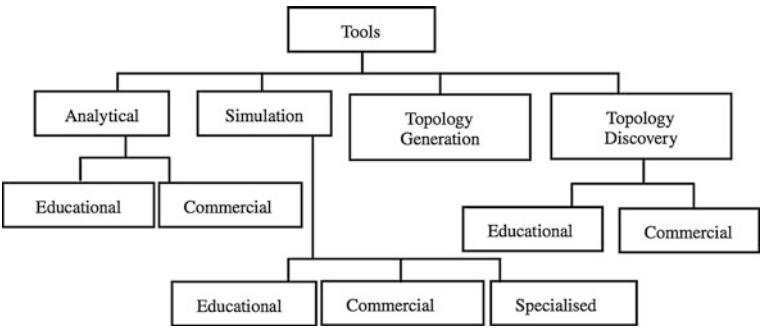


Fig. 1 Classification of the network design and simulation tools

The set of the most popular simulators of this kind is as follows: *OPNET* [2, 8] and *NS-2* [7], both well known by stakeholders, as well as *QualNet* [1], *OMNeT++* [3], *SSFNet/PRIME SSF* [15], and *SGOOSE* [5]. Experiments reported in this paper were performed using the *SSFNet* simulation environment developed by the Renesys Corporation with support from *DARPA*. *SSFNet* has large number of protocols models and network elements; moreover open-source code allows modification. In this paper Java based version of *SSFNet* was used since Java language allowed much faster development then a usage of C++. *SSFNet* simulator consists of three major parts: *SSF* engine, *DML* language [4] and *SSFNet* models.

The *SSF* (*Scalable Simulation Framework*) is public-domain standard for discrete-event simulation implemented in C++ with Java and C++ interface. Scalable Simulation Framework is a base for higher level—the *SSFNet*. *SSFNet* module is a collection of Java packages for modeling and simulation of networks and Internet protocols. Moreover *SSFNet* uses public-domain standard called *DML* (Domain Modelling Language) to configure simulation scenarios. For the purpose of this work some extensions were developed, mainly connected with support for traffic generation (models of user behavior), simulation of business level services, implementation of resource consumption for requests processing. Since fault and failures models are integral part of dependability analysis the *SSFNet* was extended to in-corporate errors. Errors were introduced in different levels beginning from link failures, network adapter failures to software component failures [15].

Additional modules of the tool required the extension of its input language (*DML*) used in standard *SSFNet* version, but the most important extension was implementing Monte-Carlo approach [15] based on running simulation several times and calculating results based on averages values. In this way—during each simulation—the parameters described in by stochastic processes—were the traffic generation which modeled user behavior in a random way. They have different values (according to set-up distributions) including an influence on the system behavior. The capability of multiple runs of simulation was added to standard *SSFNet* package by changes in several *SSFNet* classes (setting up random seed and clearing all static collections). Results of simulation are recorded in specified output file that allows further post- processing in case of dependability metrics.

3 Dependability Metrics

In case of further metrics definitions, some algebraic operation must be noted [11, 12]:

- Selection $\sigma_{condition}(R)$—selection of row in relation R that correspond to a given condition set;
- Projection $\pi_{attribute\_id}(R)$—selection of columns in relation R that eliminates duplicates.

Using this operation, we can define and calculate metrics that will be presented below.

3.1 Analytical Metrics

One of the metrics is called **Network Cohesion in the System *NCY***, understood as a number if direct (unidirectional) links between system nodes. Metric is calculated as:

$$NCY = \left|\pi_{\text{InvokerNode,ServiceNode}}\left(\sigma_{\text{InvokerNode} \neq \text{ServiceNode}}\ (A)\right)\right| \tag{1}$$

Unidirectional link between node *N1* and *N2* means that induction of the service provided by node *N2* from the client is located on the node *N1*.

Number of Services Involved in the Compound Service *NSIC[c]*—can be understood as a number of services related to complex services both directly and indirectly:

$$NSIC[c] = \left|S^R[c]\right| \tag{2}$$

This metrics is the simplest indicator of service complexity.

Absolute Importance of the Service *AIS[s]*—is a number of clients, that are part of the set *s* using their methods. It should be noted that number of clients in a node *n* is not taken into consideration (not counted), because communication between software on the same node is not causing additional load of network. Moreover, possibility of other service availability at the same node is almost equal to zero. For these reasons, Absolute Importance of the Service *s* can be calculated as:

$$AIS[s] = \left|\pi_{\text{Invoker}}\left(\sigma_{\text{InvokerNode=n,Service=s}}(A)\right)\right| \tag{3}$$

Absolute Dependence of the Service *ADS[s]*—is defined as a number of services that service *s* is depended on:

$$ADS[s] = \left|\pi_{\text{Service}}\left(\sigma_{\text{Invoker=s,ServiceNode=n}}(A)\right)\right| \tag{4}$$

Value of this metric is inversely proportional to self-sufficiency of the service, in consequence to a level of its potential autonomy and usability in other environment.

Absolute Criticality of the Service *ACS[s]*—is a the product of its absolute importance and absolute dependence:

$$ACS[s] = AIS[s] \times ADS[s] \tag{5}$$

It represents a point of attention that the creator of a service-oriented system must devote to service *s*. Neither important services that do not depend on a number

of other services, nor insignificant services that depend on many other services are not as critical for the system, as services that are both very important and have many dependencies.

Overall Reliability of the Compound Service *RC[c]*—is the reliability of a complex service c that is inversely proportional to the weakest cell in its chain (reliability of the most important services in complex service c):

$$RC[c] = \frac{1}{\max(\{ACS[s] | s \in S[c]\})} \tag{6}$$

This metrics can be seen as a main parameter/metrics for dependability of complex services.

3.2 Experimental Metrics

In [1] we can find basic set of six dependability attributes, i.e. availability, reliability, safety, confidentiality, integrity, and maintainability. These should be read as follows: availability as readiness for delivering a correct service; reliability means continuation of delivering a correct service; safety, i.e. possibility of appearance of a catastrophic consequences for user and environment; confidentiality, that is risk of compromising unauthorized information; integrity—chance that system will step into erroneous state; maintainability which is capability of a system to recover (submit for repairs and modifications).

Availability function *A(t)* for the system is a probability that system is working properly in time t:

$$A(t) = P\{system\ is\ working\ in\ time\ t\} \tag{7}$$

In the Network Systems with more than one level of abstraction, we can suppose, that the system is available as a probability that in time t all requests come from users to the system and services are supported correctly. On this basis we can estimate that, the **Business Service Availability (*BSA*)** can be computed on the basis of observed system uptime in the analyzed period T over N simulation as:

$$BSA = \frac{1}{NT} \sum_{i=1}^{N} t_{up}^{i} \tag{8}$$

whereas t_{up}^{i} is a time of service being working in i-th simulation.

Above formula (8) requires defining what does it mean that service is working. Since we are looking on the system from the client perspective, we assume that service is working if and only if it responds to the client with a proper response. The downtime starts when for some request there is no proper response (the time of starting of response is used). It finished when for any request there is a proper

answer (also a request send time is used in this case) [13]. In a very similar way we can calculate **Server Availability (*SA*)** as:

$$SA = \frac{1}{NT} \sum_{i=1}^{N} t_{up}^{i} \tag{9}$$

In this case we can calculate another level of abstraction in Network System, that is hardware one. In this case, t_{up}^{i} is a time of server being working in i-th simulation.

Service Component Availability (*SCA*)—defined as a ratio of the expected value of the uptime of a system to the observation time. It means that information system is working in a business sense (considering all sub-services that is service components). We proposed [4] to estimate the availability as a ratio of properly answered requests ($N_{correct}$) over all requests ($N_{requests}$):

$$SCA = \frac{N_{correct}}{N_{requests}} 100\,\% \tag{10}$$

Business Service Response Time (*BSRT*) and is intended to be a numerical representation of client's perception of particular service components quality. It is calculated for each tasks separately as an average delay between the starting time of user response ($t_{i\_request}$) and getting answer ($t_{i\_response}$) from the service (i.e. only requests that were properly answered are taken into account).

$$BSRT = \frac{1}{N_{request}} \sum_{i=1}^{N_{request}} (t_{i\_request} - t_{i\_response}) \tag{11}$$

4 Case Study

For the case study analysis we propose an exemplar service illustrated in Fig. 2. The system is composed of two networks: one is a client network (*client1* and *client2*; replicated 100 times), other service provider infrastructure. For service realization system contains few servers: *WebServer*, *DNSServer*, *DataBaseServer*, etc. Assuming that it is a travel agency system the possibly client behaviour, is called *BuyTrip* shown in Fig. 3.

For the purpose of further analysis, each service is located on only one machine and its failure is no longer than a half of the simulation time. Metrics proposed in Sect. 3 where calculated at the specified test case scenario (Fig. 3).

Number of Services Involved in the Compound Service (*NSIC*):
client2 = 5 [DNS, INTRANET, AUTH, INFOAPP, DB_STORAGE]
Network Cohesion in the System (*NCS*) is: 11

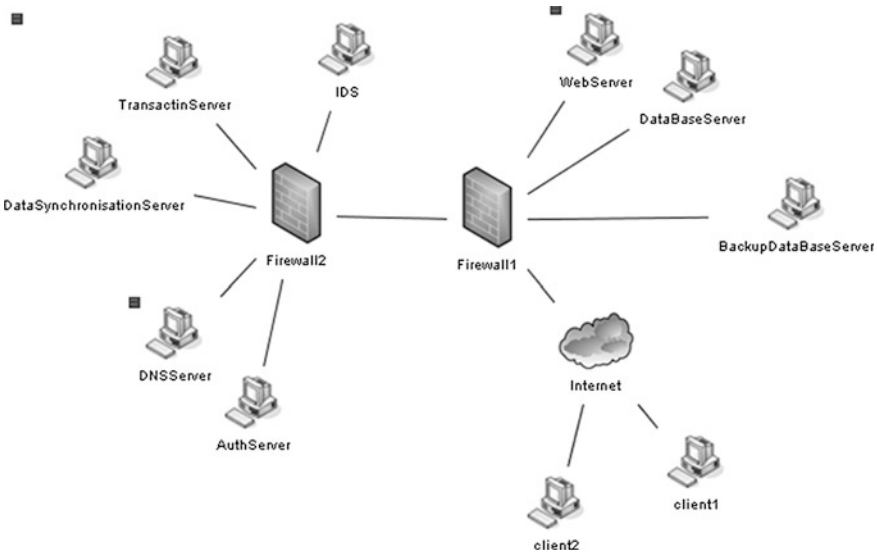


Fig. 2 Service technical infrastructure

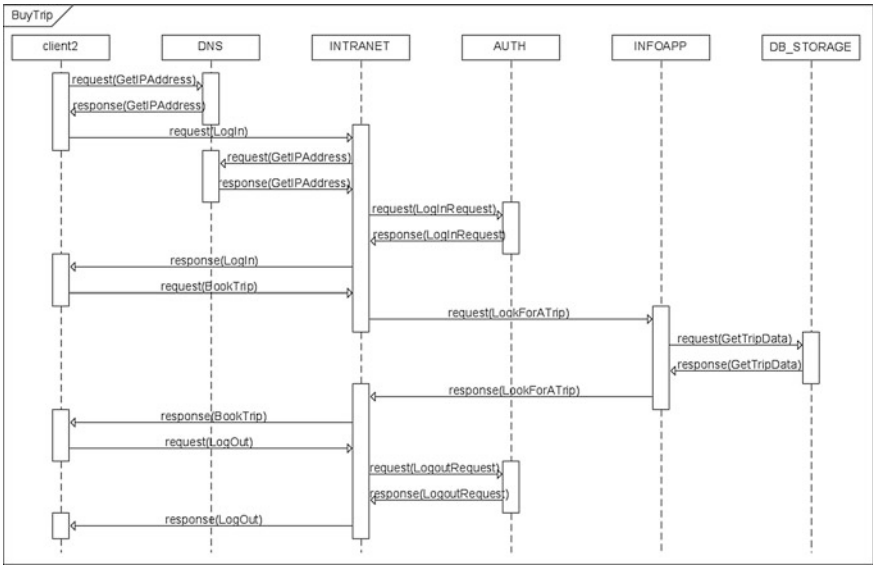


Fig. 3 Test case scenario

Absolute Dependence of the Service (*ADS*):

INTRANET = 3 [DNS, AUTH, INFOAPP] : DB_STORAGE = 0 [] : AUTH = 0 [] : DNS = 0 [] : INFOAPP = 1 [DB_STORAGE] : CLIENT = 0 []

Absolute Importance of the Service (*AIS*):

INTRANET = 1 [client2] : DB_STORAGE = 1 [INFOAPP] : AUTH = 1 [INTRANET] : DNS = 2 [client2, INTRANET] : INFOAPP = 1 [INTRANET] : CLIENT = 0 []

Absolute Criticality of the Service (*ACS*):

INTRANET = 3 : DB_STORAGE = 0 : AUTH = 0 : DNS = 0 : INFOAPP = 1 : CLIENT = 0

Overall Reliability of the Compound Service: 0.3333333333333333

Since simulation allows to observe different parameters of observed model we focused mainly on a metrics proposed in Sect. 3. In order to perform some interesting experiments we consider two hypothetic situations that may accrue in our testbed system: that all elements are working properly, some failures were introduced (the failure of *DataBaseServer* staring at 1000 s and finished at 5000 s). As shown in state charts in Fig. 3, failure of the *DataBaseServer* has strong influence in case of *BuyTrip* scenario.

The results of a comparison of the two configurations ('without failures' and 'failed') are given in Figs. 4, 5 and 6. The simulations was performed for simulation time of 10000 s and repeated 100 times.

As it could be expected, it shows that in case of failures occurrence availability drops down in both cases—correlated service and server—for scenario *BuyTrip* and for server that is related to that action. Since only requests that were properly answered are taken into account calculating *Business Service Response Time* keeps the same value in both cases.

From the results given above, we can see that the *client2* uses 5 services which have various criticality for the whole service. The most dependent service is

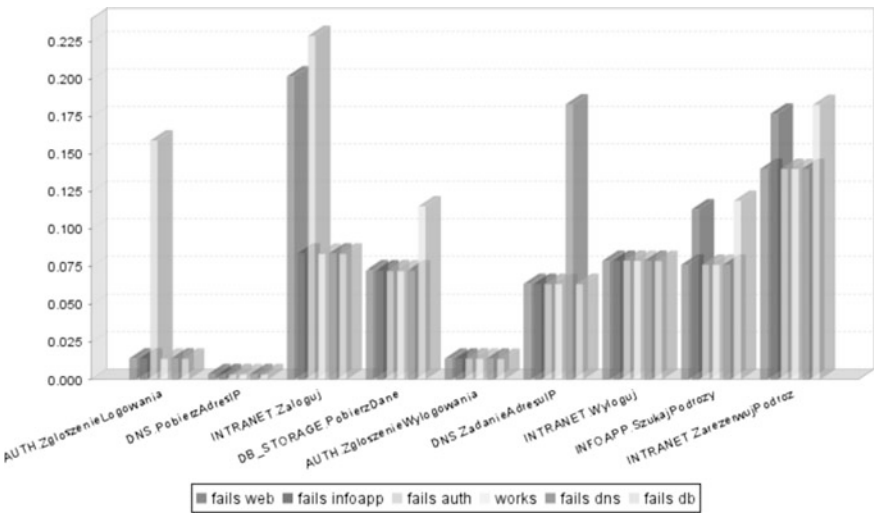


Fig. 4 Business Service Response Time—metric results

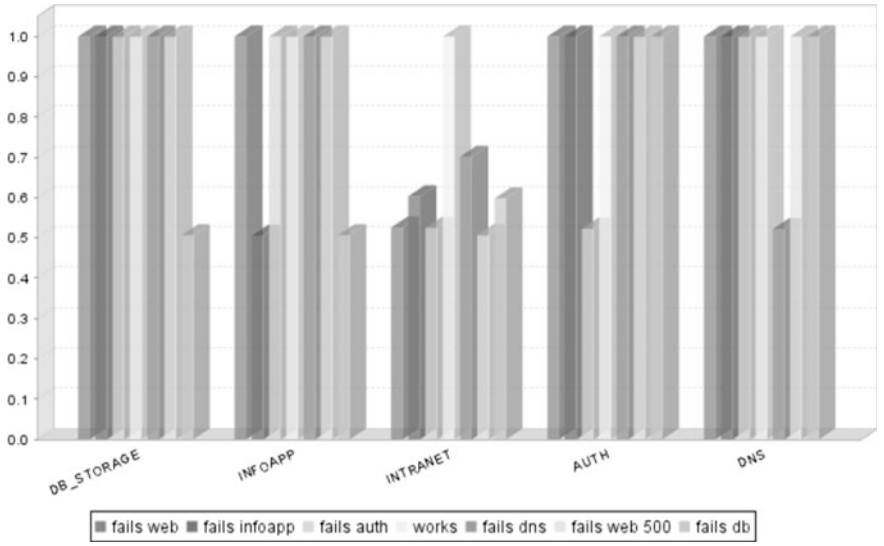


Fig. 5 Service Component Availability—metric results

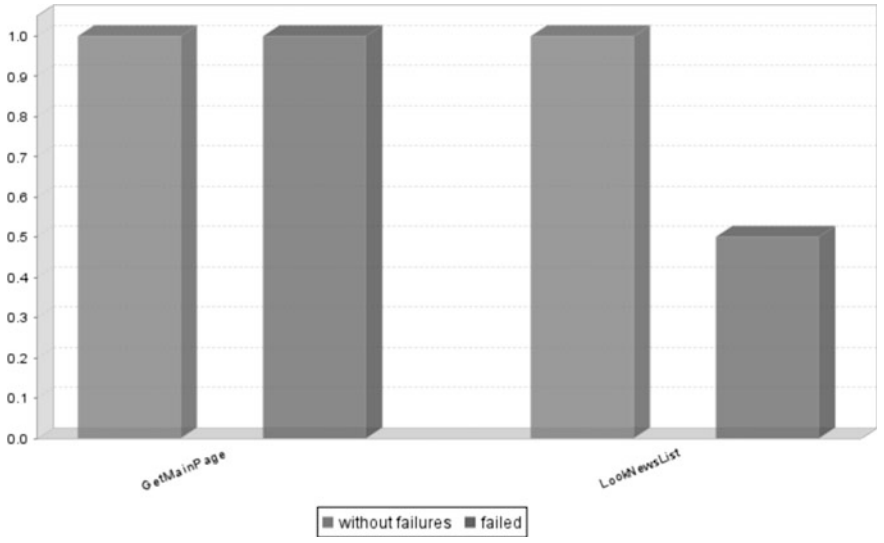


Fig. 6 Business Service Component Availability—metric results

INTERNET, next (also worth to analyze) is an *INFOAPP*. Because of the low values of *ACS* and *AIS*, Overall reliability of *Compound Service* is not reaching value 1.

It means that the system can be seen as dependable. Comparing results of *SCA* metric with *ADS* metric, we can see that *INTERNET* service is crucial for whole

system, since failure of this element influence other parts. Moreover, since (as shown in Fig. 3.) *INTERNET* is using *INFOAPP* and depends on *DB_STORAGE* than based on mathematical transitivity it will influence *INTERNET*.

On the other hand, looking at the results of *BSRT* in all faulty cases we can observe that if database server (and a service located on the node—*DB_STORAGE*) is unavailable than user will wait for a longer time for a proper data and may cause to a significant *INTERNET* delays. These results are not convergent with *ADS*, *AIS* and *ACS* metrics.

5 Conclusions

We have presented a set of analytical and experimental dependability metrics of the network systems. The analytical set of metrics included: network cohesion in the system, number of services involved in the compound service, absolute importance, dependence and criticality of the service and overall reliability of the compound service.

Moreover, two experimental set of metrics, i.e. task response time and service component availability are presented. Based on the results it is possible to create different metrics to analyse the system in case of reliability, functional and economic case. The metric could be analysed as a function of different essential functional and reliability parameters of network services system.

The presented approach—based on two streams of data: dependability factors and the features defined by the type of business service realized—makes a starting point for practical tool for defining an organization of network systems maintenance. It is possible to operate with large and complex networks described by various—not only classic—distributions and set of parameters.

The model can be used as a source to create different measures—also for the economic quality of the network systems. The presented problem is practically essential for defining and organization of network services exploitation.

References

1. Al-Kuwaiti, M., Kyriakopoulos, N., Hussein, S.: A comparative analysis of network dependability fault-tolerance, reliability, security, and survivability. IEEE Commun. Surv. Tutorials **11**(2), 106–124 (2009)
2. Bonabeau, E.: Agent-Based Modelling: Methods and Techniques for Simulating Human Systems. Proc. Natl. Acad. Sci. (2002)
3. Jennings, N.R.: On Agent-Based Software Engineering, Artificial Intelligence, vol. 117, pp. 277–296. Elsevier Press (2000)
4. Kołowrocki, K.: Reliability of Large Systems. Elsevier, Amsterdam (2004)
5. Kyriakopoulos, N., Wilikens, M.: Dependability and Complexity: Exploring Ideas for Studying Open Systems, EN. EC Joint Research Centre, Brussels, Belgium (2001)

6. Mascal, C.M., North, M.J.: Tutorial on Agent-based modelling and simulation. In: Winter Simulation Conference (2005)
7. Melhart, B., White, S.: Issues in defining, analyzing, refining, and specifying system dependability requirements. In: Proceedings of the 7th IEEE International Conference and Workshop on the Engineering of Computer Based Systems (ECBS 2000), April 3–7, 2000, pp. 334–340. IEEE Computer Society, Edinburgh, Scotland, UK (2000)
8. Michalska, K., Walkowiak, T.: Modeling and simulation for dependability analysis of information systems. In: Świątek, J. et al. (eds.) Information Systems Architecture and Technology. Model Based Decisions, pp. 115–125. University of Technology, Wroclaw (2008)
9. Nicol, D., Liu, J., Liljenstam, M., Guanhua Y.: Simulation of large scale networks using SSF. In: Proceedings of the 2003 Winter Simulation Conference, vol. 1, 7–10 December 2003, pp. 650–657 (2003)
10. Nikolaidou, M., Anagnostopoulos, D.: A Distribution System Simulation Modeling Approach, Simulation Modeling Practice and Theory, vol. 11, pp. 251–267 (2003)
11. Rud, D., Schmietendorf, A., Dumke, R.: Product metrics for service-oriented infrastructures. applied software measurement. In: Proceedings of the International Workshop on Software Metrics and DASMA Software Metrik Kongress (IWSM/MetriKon). Magdeburger Schriften zum Empirischen Software Engineering, November 2006, pp. 161–174, Potsdam, Germany, Hasso-Plattner-Institut, Shaker Verlag (2006)
12. Rud, D., Schmietendorf, A., Dumke, R.: Resource metrics for service-oriented infrastructures. In: Proceedings of the SEMSOA 2007, pp. 90–98, May 10–11, Hannover, Germany (2007)
13. Volfson, I.E.: Reliability criteria and the synthesis of communication networks with its accounting. J. Comput. Syst. Sci. Int. **39**(6), 951–967 (2000)
14. Walkowiak, T., Mazurkiewicz, J.: Algorithmic approach to vehicle dispatching in discrete transportation systems. In: Sugier, J., et al. (eds.) Technical Approach to Dependability, pp. 173–188. Wroclaw University of Technology, Wroclaw (2010)
15. Walkowiak, T., Mazurkiewicz, J.: Functional availability analysis of discrete transportation system simulated by SSF tool. Int. J. Crit. Comput.-Based Syst. **1**(1–3), 255–266 (2010)
16. Zhang, J., Zhang, L.-J.: Criteria analysis and validation of the reliability of web services-oriented systems, web services. In: IEEE International Conference on Web Services (ICWS'05), pp. 621–628 (2005)

Assessing the Costs of Losses Incurred as a Result of Failure

Katarzyna Pietrucha-Urbanik

Abstract Recently, availability of the water supply system safety is the important issue in the water management. Water network is very complex and therefore requires proper analysis of its functioning. Water companies are responsible for providing potable water to the recipients, who anticipate that drinking water utilities guarantee reliable and safe supply. The aim of this paper is to present the method of failure cost assessment with application of multidimensional comparative analysis. In the paper the set of cost analysis was presented, that can be implemented to the assessment of planning investments and modernization. The approach will enable the comparison of costs level intended for failure removal in different water supply system.

Keywords Water network functioning · Failure losses · Water supply

1 Introduction

The task of water supply system is to provide water to consumers with adequate quality, quantity, at any time, with acceptable price. Unfortunately, sometimes it is difficult to fulfil all those parameters during failure which random character disturbs in its total elimination. The important factor associated with undesirable event is the cost of losses bore during and after failure occurrence. Legally for the delivery of water with quality threatening the life or health is responsible a water supplier who ensures the safety and comfort of the use of water services. Not only a recipient, but also the water company bears the consequences of unsatisfactory quality parameters, resulting in financial losses for not supplied water and additional costs associated with the failure of water distribution system.

K. Pietrucha-Urbanik (✉)
Faculty of Civil and Environmental Engineering, Rzeszow University of Technology,
Al. Powstańców Warszawy 6, 35-959 Rzeszow, Poland
e-mail: kpiet@prz.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_31

Rules for sharing the cost of losses are formulated by the producer of drinking water. Rules for sharing the costs depend on many factors, among others, a side which makes demands. It is important to take into account the interests of the consumer. Requirements relating to the supply of drinking water in the required quantity and quality should be included in the contract. An important issue when concluding the contract is taking responsibility for a failure to comply with the requirements of the contract [10]. One way to encourage the producers to fulfill the commitments of the agreement is the introduction of the rules for charging the manufacturer's total or proportional cost of losses borne by consumers in the case of limiting service or its omission [7, 9, 14].

In extreme cases or extraordinary situations caused by lack of water, depending on the randomly occurring forces of nature, such as, for example, drought, flood, earthquake or if the producer does not feel responsible, also when situation is not caused by his fault, such cases should be subjected to appropriate insurance [8, 13].

Waterworks company losses resulting from unpredictable events can be connected to the damage or partial loss of infrastructure, payment of compensation to customers for water interruption in water supply related to the loss of confidence of water consumers. In order to estimate the losses of water recipient in case of water supply system failure it should be remembered that the profit of any company is the difference between income and expenditure. The enterprise starts to bear losses if costs start to exceed income [11]. Losses of water company as a result of undesirable events can be calculated as the sum of costs incurred for penalties (or court costs), to remove the consequences of such events (e.g. for repairing water mains) [5]. Losses should be increased by income from the water which has not been sold due to failure.

Costs of failure removal consist of the cost of materials needed to restore damaged or destroyed infrastructure; salary costs, when the water company has its own repair brigade these costs may increase slightly or a significant increase in such costs may occur in case of the payment of overtime or necessity to rent a repair brigade and other costs, e.g. transport of materials. For the water company especially important is the amount of unsold water can only be estimated using the average amount of water consumed by customers before failure [4, 12].

An increasing awareness of water consumers about their rights causes that the water supply market is significantly more focused to the recipient who bears the costs in the form of charges and has the right to demand an appropriate level of service. Analysis of the costs bore to maintain the reliability at a particular level is very important when planning renewal or repair of water network [18, 21]. When designing new or modernizing old water networks such systems should be developed which allow to maintain the reliability at the determined level. It is important to reduce the costs or keep the same costs to maintain the high level of reliability. In the papers [16, 17] the importance of taking issues related to the protection of the recipient was highlighted, proposing the implementation of the indicators describing the level and costs of services provided by water companies which will also help in improving their functioning. Specially developed tools are very helpful

in planning expenditures and future modernizations in budget of waterworks company [22]. Therefore, the management of the water supply should perform the detailed analysis of undesirable events and procedures when an event occurs, which also help to maintain proper level of reliability in critical infrastructure [6, 19]. Only such action will ensure the reduction of losses being a result of failure to a minimum. The objective of this work is to characterize the change of costs of water network renewal.

2 Materials and Methods

The analyses were made using the data from the daily operation of the examined water network, also from the failure protocols of Municipal Water Company. The analyses were made by the Statsoft computer program Statistica. Data characterizing failures include information about failure, the pipe age, temperature, time of failure removal, type of ground, data on the consequences of failure, costs of the failure removal, material cost and salary cost, costs of losses including unsold water during failure.

The analysis of costs spent on failure removal by water company was made through cluster analysis which involves aggregating the similar observations. Such method focuses on grouping parameters, which vary considerably among themselves [1, 3]. Created in this way cluster is then organized by analysing similarities among all distinguished elements. Taken into account indicators may form such group through similarity or distance. In this way the method is obtained, which shows both the same and different properties.

In order to estimate the distance between the clusters the following method can be included [15]:

- single linkage (nearest neighbor), the distance between two clusters is predefined by the distance of the nearest neighbors belong to different clusters,
- complete linkage (the furthest neighbour), the distance between clusters is the longest distance between any two objects in the different clusters,
- unweighted pair-group average, the distance between two clusters is defined as the average distance between all pairs of objects belong to two different clusters, here also method of weighted pair-group average can be distinguished where the size of the proper clusters is used as a weight,
- unweighted pair-group centroid, the centre is a cluster of the average point in the multidimensional space defined by the dimensions, similar is weighted pair-group centroid (median) method, where the calculation is weighed as to take into account the differences between the clusters sizes,
- the Ward's method, the most effective method, distances between clusters are estimated on the basis of variance analysis, where the sum of squared deviations of any two clusters that can be formed at any stage is minimized.

Among others the most reliable methods is single/complete linkage, weighted pair-group average and the most effective is the Ward's method, as the linkage type the Euclidean distance was implemented. In order to verify if the individual failure belongs to cluster the method proposed by Z. Hellwig was implemented. The Hellwig method is used to estimate the distance between points belonging to two different subsets [2]. The Hellwig distance is known as the critical value. The first step is the calculation of the minimum value for each row in the distance matrix, then determining a new variable of arithmetic mean and standard deviation. In the end the calculation of the critical value as the sum of arithmetic mean and standard deviation is determined [20]. In order to present the characteristics of the analyzed failure the average of each cluster is determined by k-means method. K-means compared with hierarchical clustering methods guarantees maximum different from each other, especially after grouping a large number of elements considerably increases transparency of the obtained results.

3 Results and Discussion

Hierarchical clustering methods were applied, as to create a tree diagram in which the hierarchy of individual clusters was visualized (Fig. 1).

The method of single linkage form indistinguishable concentration of data in comparison to other methods of linkage. At the beginning of observation the single linkage method allowed to find outliers that not belong to any group. In comparison to this method complete linkage formed a compact cluster of small diameter. In the case of the Ward's method, formed clusters have small size and clusters are more complete. The Ward's method gave very good results because it created a very homogeneous clusters.

As to characterize the particular cluster the method of k-means was applied, which belongs to iterative methods. This method guarantees receiving clusters, which maximally differs from each other, what after grouping a large number of elements considerably increases transparency of the obtained results. In the Fig. 2 are shown results of the clustering k-means (Table 1).

The aggregated cost of each clusters can be characterized as follow: 1st cluster—high cost of material and average total costs, 2nd cluster—high cost of material and high total costs, 3rd—very high cost of material and very high total costs and 4th cluster very low cost of material and low total costs.

As to find out which parameter influences clusters the detailed analysis of the distinguished clusters was shown on the Fig. 2.

The highest costs of failure removal occurred in the third cluster (1.74 % of all cases) and the lowest in the fourth cluster (57.39 % of all cases). The other two clusters are both characterized by low material cost and high total cost of failure removal in case of the first cluster and the average total cost in the case of the second cluster, what was demonstrated by a standard deviation.

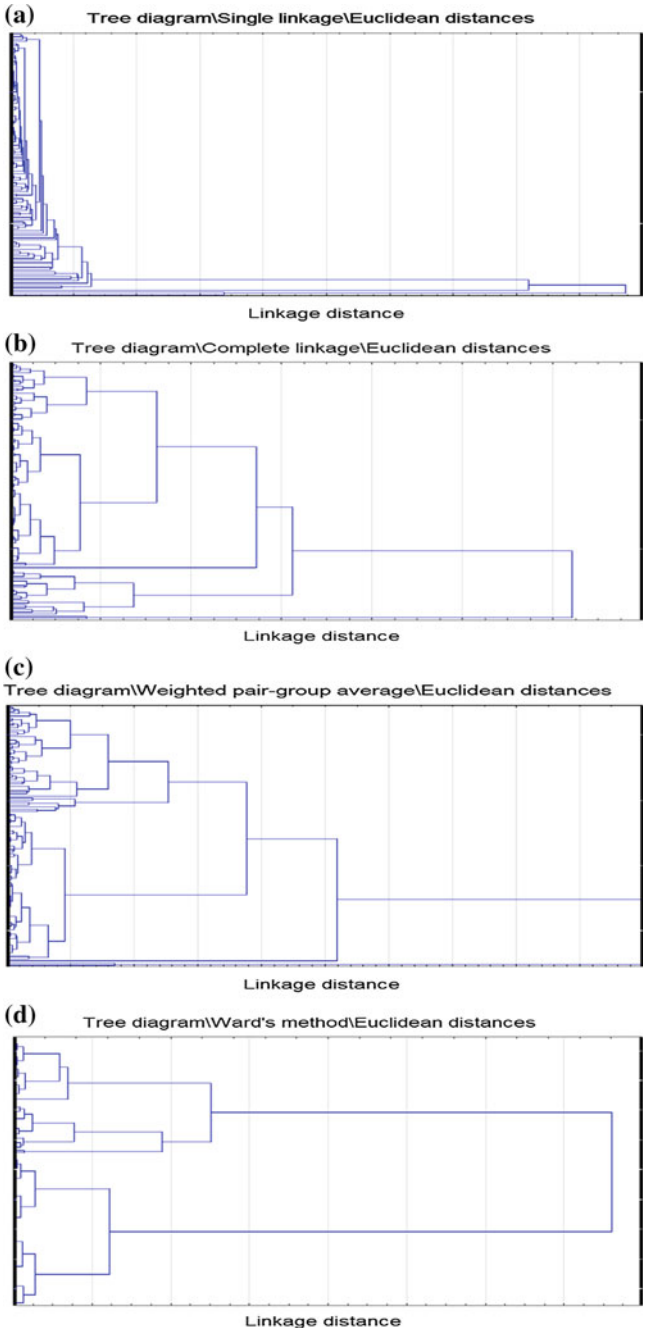


Fig. 1 Tree diagram of differentiation of particular failures for the single (**a**) and complete linkage (**b**), weighted pair-group average (**c**) and Ward's method (**d**)

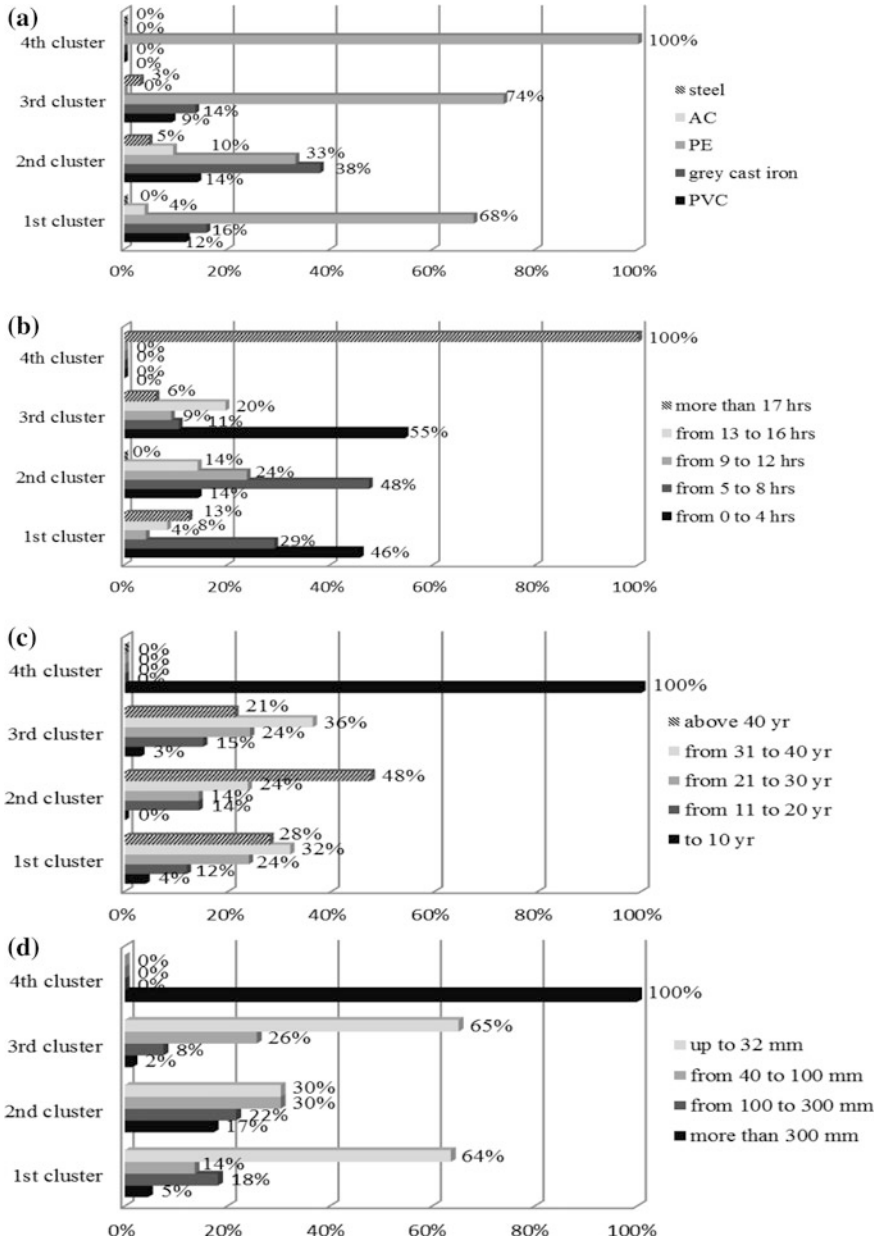


Fig. 2 Characteristics of particular clusters by k-means method, considering: **a** material, **b** time of failure removal, **c** pipe age, **d** pipe diameter

Table 1 Division of costs associated with failure removal by k-means method

Specification n%	1st cluster 22.61 %	2nd cluster 18.26 %	3rd cluster 1.74 %	4th cluster 57.39 %
Arithmetic average				
Cost of material (€)	245.32	244.70	340.17	41.27
Total costs (salary of worker, equipment etc.) (€)	1302.99	2271.45	5970.73	459.87
Standard deviation				
Cost of material (€)	168.87	470.89	252.73	67.81
Total costs (salary of worker, equipment etc.) (€)	236.55	422.42	375.46	214.05

Conducted analysis highlighted the failure parameters influencing the cost associated with the failure removal. The third cluster was dominated by failures, which occur on pipes of 31–40 years and are characterized by long time of failure removal more than 13 h, what resulted in higher cost of failure removal. The lower cost of failure removal was associated with removal failure of pipe from PE on water supply connections (1st cluster). Detailed analysis showed that the increased costs of failure removal are on pipes made of grey cast iron and asbestos cement, while smaller on plastic.

4 Conclusions and Perspectives

The presented method may be useful in providing information about borne costs of failure removal. Data clustering helps in aggregating cost data from failure into homogeneous cluster during enhancing heterogeneity over groups. The methodology can be used to describe the cost of water network failure removal by use of the multidimensional comparative analysis method. It can provide a basis for comparative assessment between different supply systems in terms of the cost of failure removal on water pipes.

For the future research, the analysis expanded by results of tests performed in water supply systems in which modernization took place, will constitute the assessment of functioning effectiveness of the system. The continuity of water supply significantly depends on failure occurrence in water network. Water pipes constitute a significant part of the water companies assets. Therefore for water industry it is important to minimize the cost of water supply functioning and at the same time provide water with appropriate parameters.

References

1. Abonyi, J., Feil, B.: Cluster Analysis for Data Mining and System Identification. Birkhauser Verlag AG, Germany (2007)
2. Anderberg, M.R.: Cluster Analysis for Applications. Academic Press, London (1973)
3. Glowacz, A., Glowacz, A., Glowacz, Z.: Recognition of thermal images of direct current motor with application of area perimeter vector and Bayes classifier. Measur. Sci. Rev. **15**, 119–126 (2015)
4. Hamilton, S., McKenzie, R.: Water Management and Water Loss. IWA Publishing, London (2014)
5. Kwietniewski, M., Roman, M., Trębaczkiewicz-Kłoss, H.: Water and Sewage Systems Reliability. Arkady Publisher, Warszawa (1993)
6. Markovic, G., Ondrejka Harbulakova, V.: The current approaches in urban drainage. In: Colloquium on Landscape Management, pp. 63–70. Mendel University, Brno (2015)
7. Mazurkiewicz, J.: Agent approach to network systems dependability analysis in case of critical situations. In: Dependability Problems of Complex Information Systems. Adv. Intel. Syst. Comput. **307**, 73–89 (2015)
8. Ondrejka Harbulakova, V., Markovic, G.: Infiltration systems used in urban area—an overview, In: Zborník recenzovaných vedeckých prác ÚEI, pp. 71–76. TU, Košice (2015)
9. Ondrejka Harbulakova, V., Purcz, P., Estokova, A., Luptakova, A, Repka, M.: Using a statistical method for the concrete deterioration assessment in sulphate environment. Chem. Eng. Trans. **43**, 2221–2226 (2015)
10. Pietrucha-Urbanik, K.: Assessment model application of water supply system management in crisis situations. Global. Nest. J. **16**, 893–900 (2014)
11. Pietrucha-Urbanik, K.: Failure analysis and assessment on the exemplary water supply network. Eng. Fail. Anal. **57**, 137–142 (2015). doi:10.1016/j.engfailanal.2015.07.036
12. Pietrucha-Urbanik, K.: Failure prediction in water supply system—current issues, theory and engineering of complex systems and dependability. Adv. Intel. Syst Comput. **365**, 351–358 (2015). doi:10.1007/978-3-319-19216-1_33
13. Rak, J.R.: Some aspects of risk management in waterworks. Ochr. Srod. **29**, 61–64 (2007)
14. Rak, J.: Selected problems of water supply safety. Environ. Prot. Eng. **35**, 23–28 (2009)
15. Sokołowski, A.: Statistics in medicine—advanced methods. Course Materials (2011)
16. Tabesh, M., Soltani, J., Farmani, R., Savic, D.: Assessing pipe failure rate and mechanical reliability of water distribution networks using data-driven modeling. J. Hydroinform. **11**, 1–17 (2009)
17. Tchorzewska-Cieslak, B., Szpak, D.: Proposal of a method for water supply safety analysis and assessment. Ochr. Sr. **37**, 43–47 (2015)
18. Tchórzewska-Cieślak, B., Pietrucha-Urbanik, K., Urbanik M.: Analysis of the gas network failure and failure prediction using the Monte Carlo simulation method. Eksploat. Niezawodn. **18**, 254–259 (2015). http://dx.doi.org/10.17531/ein.2016.2.13
19. Valis, D., Zak, L., Pokora, O.: Contribution to system failure occurrence prediction and to system remaining useful life estimation based on oil field data. Proc. Inst. Mech. Eng. Part O-J. Risk Reliab. **229**, 36–45 (2015)
20. Ward, J.H.: Hierarchical grouping to optimize an objective function. J. Am. Stat. Assoc. **58**, 236–244 (1963)
21. Wieczysty, A.: Methods of Assessing and Improving the Reliability of Municipal Water Supply Systems. Committee of Environmental Engineering Sciences, Cracow (2001)
22. Zimoch, I., Lobos, E.: Application of the theil statistics to the calibration of a dynamic water supply model. Environ. Prot. Eng. **36**, 105–115 (2010)

Impulse Transmission Model of Macroeconomic Cycle Within the Framework of the Theory of Shocks: Aspect of Economic Security

Z.A. Pilipenko, E.V. Savenkova, A.I. Pilipenko, E.A. Morosova and O.I. Pilipenko

Abstract Hereinafter presented approach to the categories of the theory of shocks outlines a certain progress on the way to understand shocks in the context of cyclical development of national economies and on the global level [1]. The research evolves the Slutsky' hypothesis of the cyclical model of the economic system's reaction on the random effects of impulses (shocks) combined with the impulse transmission approach to the model of the macroeconomic business cycle by Ragnar Frish, tested by American scientists Irma And Frank Adelman [2] with the help of Klein-Goldberger model.

Keywords Market prices · Volatility · Assets real value · Bifurcation · Shock · Marx cycle of reproduction · Kondratiev "big cycles" · The Slutsky' hypothesis of the cyclical model · Economic security

Z.A. Pilipenko (✉)
Banking Supervision Department (Formerly Banking Regulation & Supervision Department), Bank of Russia, 12 Neglinnaya Street, Moscow 107016, Russia
e-mail: PZA1@cbr.ru

E.V. Savenkova · E.A. Morosova · O.I. Pilipenko
Department of Finance and Banking, Peoples' Friendship University of Russia, Miklukho-Maklaya str. 6, Moscow 117198, Russia
e-mail: Savenkova.rudn@mail.ru

E.A. Morosova
e-mail: Moroz_Elena@rambler.ru

O.I. Pilipenko
e-mail: Zuslik1@rambler.ru

A.I. Pilipenko
Department of Economics & Mathematics Models, Peoples' Friendship University of Russia, Miklukho-Maklaya str. 6, Moscow 117198, Russia
e-mail: students_forum@mail.ru

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*, Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_32

1 Introduction

Theoretical findings of the paper are dedicated to the decision of three groups of interconnected problems: (1) to estimate the quality of external factors of economic systems (characterized with different level of organization structure [3]) capable to cause disruptive/creative effects on economies; (2) to ascertain the functions (properties) of mechanism of shocks' transmission to national economic systems; (3) to detect marginal conditions of structural connections of economic systems-shocks' recipients, when the process ensuring market equilibrium is blocked.

Analytical part of paper focuses on the transition from the investigation of essence of phenomenon of shock and the multiple forms of its destructive actions with the regard to structural connections of dialectic pairs to the concretization of evident cyclical reaction of national economic systems to the external shock effects. In this very context, it becomes clear the access to the practice of macroeconomic regulation that is associated with the solution of the economic security' problems at the national level. They are connected with the situation, when the market price parameters of the financial assets are achieving the marginal price parameters in comparison with their real value. This results in the generation of the economic and financial imbalances that mean the economic crisis' beginning and the necessity of the regulation measures in respect of national economic security.

The use of Marx methodology to structure interconnections between macroeconomic segments by phases of GDP formation revealed that the shocks, which are known in the world economy, do concentrate on the first and third stages of the Marx cycle of reproduction—the circulation of goods and factors of production. The conducted research made possible to structure time parameters of price shocks, denoting their dominating exceptionally on the stages of evolution from one big wave of Kondratiev' "big cycles" to another. On this basis the beginning of macroeconomic cycle is supposed to be connected with the reaction of economic systems to the external shock effects. This conclusion confirms the hypothesis expressed in the previous century assuming the impulse transmission (or "shock"—in our interpretation) character of cyclical development of macroeconomic dynamics. Considering this, the government is to realize the constant monitoring of the formation of marginal price parameters on the markets of the financial assets in order to prevent in time the situation that could threaten national economic security.

Violent development of contemporary financial crisis, sequentially bringing down bond' and equities markets and then the markets of real assets as well as its global disruptive power and transformation into structural crisis challenged many approbated theoretical postulations and empirical instruments having allowed to arrest crises occurrences of the 20th century rather swiftly. In these circumstances the scientific society almost unanimously proclaimed the failure of macroeconomics [4] with its basement on the effective markets' theory and decisive transformation of structural connections in the global economy and, consequently, the laws of its

development [5]. All this combined, in our opinion, necessitates of the scientific rethinking of the role of shocks in the world economy' structurization in the situation when their influence forces across-the-border relationships become the main operating mechanism of financial imbalances' transmission to all national parts of global economic system.

Conceptual method of understanding the shocks was formed in the process of structuring three groups of interconnected problems: (1) to estimate the quality of external factors of economic systems (characterized with different level of organization structure) capable to cause disruptive/creative effects [3] on economies; (2) to ascertain the functions (properties) of mechanism of shocks' transmission to national economic systems; (3) to detect marginal conditions of structural connections of economic systems-shocks' recipients, when the process ensuring market equilibrium is blocked.

1.1 Scientific Research Preceding the Evolvement of Shocks' Theory Regarding the Laws of Cyclical Development of Economic Systems

At the beginning of the previous century the distinguished Russian scientists in the field of economics, statistics and mathematics—Kondratiev, Slutsky [6] and other representatives of the famous Kondratiev' Conjuncture Institute—raised the problem of shocks to the level of the superlative scientific importance. Thus, E.E. Slutsky was among the first to doubt the adequacy of determinist models to explain economic cycles. He validated a hypothesis pointing out that the random effects—"impulses" (or "shocks"—in modern vocabulary)—on economic system launch the cyclical response (reaction) to the external influence [1].

This point of view was shared by the great Norwegian economist Ragnar Frish of the University of Oslo, who, in his famous work "Problems of Impulse Transmission in Economy", called this approach to the model of macroeconomic business cycle as "impulse transmission" [7].

At the end of the 1950s impulse transmission approach was tested in the empirical research of economic fluctuations by Irma and Frank Adelman. Analyzing the influence of random shocks with the help of Klein-Goldberger model, they proved, that the amplitude and duration of cycles generated by modelled external impulses show striking resemblance to the actual observations [2]. The model applied by the scientists in their experiment was exposed to the effects of random shocks: the resulting fluctuations of business activity repeated the US economic dynamics, structured according to the data of NBER.

2 The Essence of Concept of Shocks as the Impulse, Generating the Beginning of the Macroeconomic Cycle

At the basis of the concept there's the detection of the unity in variety of shocks known in the world economy: the analysis of all the available information was made possible due to the wide range of investigations devoted to the history of development of crises in different countries. Close scrutiny of the shocks' diversity allowed to point out the peculiarities (features) helping to unite all the shocks described in the literature:

- first, all of them were the forms of external (with respect to the object) factors' (or group of factors) realization [3];
- second, all the shocks grew in the sphere of real and financial assets' market exchange and then affected wider economy through it;
- third, they led to different (by quality) forms of destruction (creation), reflected by the indicators of stability (soundness) of structural relationships (national and international market exchange of real and financial assets) in the markets, national economies, regions and global economic environment in different (by duration) periods of time.

In order to identify the function of shocks in impulse transmission model of macroeconomic cycle' model we distinguished three groups of interconnected problems (Fig. 1):

2.1 The Determination of Quality of External (with Respect to Economic Systems) Factors Provoking a Shock Effect

The complexity of the first group of problems is connected with the variety of shocks known in the economic history (the most illustrious cases are listed below):

(1) Tulip bulbs market crash in Holland in 1636 after the multifold increase of prices;
(2) The South Sea Bubble in the equities markets in 1720 in France;

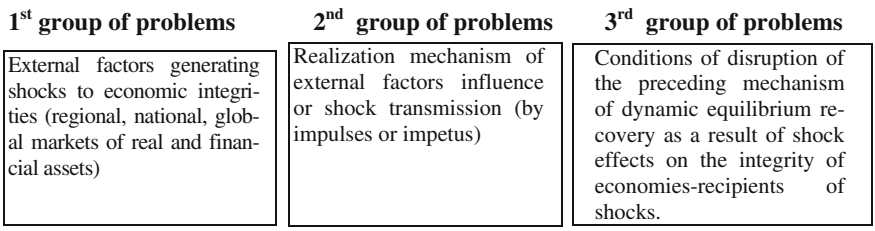


Fig. 1 Build-up logic of the shocks' theory in the context of shock factors, their transmission mechanism and interaction with the structural relationships of economic systems [8]

(3) Sharp decline of the Mississippi Company stock in 1720 in Great Britain;
(4) Regular shocks as a result of industrial revolutions starting from 1825;
(5) Stocks prices shocks around the world in 1927–1929;
(6) Shocks induced by "Great Depression" in 1929–1932;
(7) Banking crashes in Mexico and other developing countries in the 1970s;
(8) Real estate price shocks and stock market crashes in Japan in the 1980s;
(9) Real estate market shocks and stock market crashes in Finland, Norway and Sweden in 1985–1989;
(10) Real estate market shocks and stock market crashes in Thailand, Malaysia, Indonesia and a few other Asian countries in 1992–1997;
(11) Shocks of foreign investments flowing to the economy of Mexico in 1990–1993;
(12) The US housing bubble and mortgage credit crisis in 2007, resulting in "Great Instability" in the world economy at the end of the 2000s.

The analysis of the most famous shocks in the world economy (excluding natural disasters) made it possible definitely to connect their growth with the sphere of market circulation and differentiate, using the methodology of Marx, by phases of transnational reproduction in the following way: either on the third stage of reproduction process, in the sphere of realization of finished goods in the world markets (Goods—'Money'), or on the first stage (Money-Goods), intermediating the transnational movement of material and financial resources.

Definition of circulation as the exclusive sphere of shocks' generation is the key element of understanding of their importance for the organizational process in the world economy. The exchange relationships exactly mediate the connections of dialectically interconnected parties of buying/selling transactions (households-buyers and companies-sellers, investors and financial intermediaries, borrowers and creditors and so on in any economic system: regional, national and global). Hence, the shock impact on structural relationships in the sphere of circulation is capable to destroy the integrity and sustainability of systemic organization.

So, the essence of shocks lies in their capacity to execute the influence of external factor on the structure of economic systems.

2.2 Allocation of the Functions of the Mechanism of Shocks Effects on the National Economic Systems

The manifestation of the essence of the shocks influence in the dialectically connected participants' behavior on the markets, acting as buyers and sellers, allows to determine the specifics of the traded material and financial assets. The contradictions and the harmony in their relations relating to the exchange of an assets take the form of market price fluctuations around its base value. The external factor effect on the structural links between the participants of the dialectical pairs will not be a shock until the price fluctuations do not overcome the marginal level for each

particular market values. In this context, it can be argued that the shock impact on structural links of any economic integrity (which displays it by the marginal bounds) is implemented through the feverish behavior of one group of participants of the dialectical relations that become dominant market relations and demonstrate irrational behavior on the demand side, as well as on the supply side.

To explain the boom behavior of one group of participants on the markets (either on the demand-side or on the supply-side) J. Akerlof and R. Shiller introduced "confidence multiplier". Like the consumption' multiplier [9], it appears because there are several rounds in the spending circle: change in the confidence will lead to the changes in incomes and confidence in the next round and it will be the same in all the subsequent rounds. This confidence multiplier leads either to increase or to decrease of demand (in direct proportion to the trust) for a particular asset, through, for example, the initial increase in demand, pushing the market to a certain margin, finishes, ultimately, with the massive dumping of the asset in exchange for liquidity. It is in this process the bifurcation takes place as a detachment of the market price of an asset from its real value at first up and then below its value [10]. This effect is realized as the contradictory dialectical unity of the asset price (market-defined) and the based costs (measured by the manufacturer), of the buyer and the seller: as the structural links between them under the influence of external factors going beyond their limits, economic system loses its structural stability.

On this basis, all historically known shocks can be integrated and called as the price shocks, because the shape of their manifestations connects with the bifurcation of price parameters of the asset and its base value, and they can be differentiated according to the exchange objects and specific features of the massive speculations with them.

2.3 Determination of the Marginal State of the Structural Relations of Economic Shocks Recipient Systems Under Which the Ensuring Market Equilibrium Mechanism Blocks

It is a subjective behavioral mechanism that pushes exchanges market imbalance, where certain assets are traded, to shock state under the influence of the external factor, manifesting itself as increasing amplitude of market price fluctuations around its base asset value. As soon as one party to the transaction begins constantly breaking the structural balance of dialectical relations on the market there takes place the irrationally increasing demand or supply, the market price of an asset jumps smoothly up or falls down compared to its base value. It is in this sense that any shock is connected with the price shock, causing the effect of bifurcation of the market price around the asset actual cost (base). Initiation of external shocks and assets price margin overcoming on the particular market cause destruction of the structural relations between participants of dialectical pairs of buyers and sellers.

3 Mathematical Interpretation of the Bifurcation Effect in the Shocks Theory

The phenomenon of the bifurcation effect in the shocks theory can be given a mathematical interpretation. Let us imagine the appearance of the higher yielding assets on the stock market, which attract attention of the rational economic agent. Rather high income paid for the period τ and the possibility to increase the investment till the level of m for the relatively small forecast horizon attracts new participants whose behavior gradually begins to differ from rational.

Expansion of economic agents allows principals-issuers increase securities issue, more and more coming off their financial security. In other words, the behavior of principals also acquires the character of irrationality. Let the number of economic agents whose behavior is characterized as irrational because they do not analyze the real availability of securities and real terms possible interest payments by issuers without loss of economic self-sufficiency, grow exponentially with denominator q. Hence, cash inflow to issuers of securities data obeys the same geometric progression. This means that during the period n irrational economic agents acquire securities for the sum of $q^{n-1}m$.

If the behavior of principals was more rational, they could estimate the limiting moment (n—period number τ) since which resource inflow would be completely paid as interest income. Really, the condition under which some financial receipts remain to principals after interest income is paid to economic agents, can be written down, for example, as follows

$$q^{n-1}m - rm\sum_{i=1}^{n-2} q^i > 0, \tag{1}$$

where r—the interest rate for the period τ.

Applying the formula for the sum of members of a geometrical progression

$$\sum_{i=1}^{n-2} q^i = \frac{q^{n-1}-1}{q-1},$$

Let's transform an inequality (1) to a kind

$$q^{n-1}m - rm\frac{q^{n-1}-1}{q-1} > 0. \tag{2}$$

As $q^{n-1} > 1$, (2) it is possible to present an inequality as follows

$$\frac{1}{q^{n-1}-1} > \frac{r}{q-1} - 1. \tag{3}$$

The left part of an inequality (3) is always positive. The sign on the right part depends on the parity between sizes q and r. Obviously, two cases are possible. If $r < q - 1$ the inequality (3) is carried out at any n, i.e. the sum of monetary receipts increases. If $r > q - 1$ at enough big n the inequality (3) is not carried out and there can come the moment when it is not enough financial receipts even for interest payment.

The rational economic agent and the principal easily could estimate the moment of crisis of the tendency as to money resources growth to their falling. Really, in conditions when $r > q - 1$, the following restriction can be received from an inequality (3) that makes sense as below

$$q^{n-1} < \frac{r}{r - q + 1}. \tag{4}$$

Taking the logarithm of an inequality (4), we will come to the following estimation of the moment of tendencies change

$$n_{\text{lim}} = 1 + \frac{\ln\left(\frac{r}{r-q+1}\right)}{\ln q}, \tag{5}$$

where n_{lim}—marginal number of the periods of financial operation with participation of irrational economic agents.

Even such a simple scheme throughout the several following periods let us describe the exchange operations of the given principal on the share market in connection with its default.

4 Conclusions

Logic designs of the theory of shocks were based on the integrity of economic systems dealing with dialectic pairs of the phenomena, processes, etc., their structural communications mediate, as a rule, universal relations between participants concerning an exchange (purchase and sale) of various assets. It allows to combine very many historically known shocks and to call them all as price shocks, capable to destroy structural communications of counterparts as dialectic pairs. All the price shocks appear in the form of assets market prices volatility around their real (base) cost (value).

The analytical part of the work is devoted to transition from the intrinsic analysis of the shock phenomenon and numerous forms of shock appearance in dialectically connected partners' destruction to a concrete definition of the shock. In the last case the cyclic reaction of national economic systems on shock influences is interpreted as the impulse transmission approach to the model of the macroeconomic business cycle, including the external factors presented by the global economy and finance in

the newest history. The phenomenology of Kondratyev' "big cycles" in researches of K. Peres [11] allows to prove dialectic character of interaction of real sector of economy and its financial segment, played a key role not only on last wave of macroeconomic dynamics at the end of the 1990s–the beginning of the 2000s, but also within the framework of all previous cycles.

As a result priorities vary: in a limiting condition the real sector does not require financial resources and they are pushed out from manufacture to the financial markets. And eventually—on the other marginal stage of the big cycle—financial resources start dominating in the development of some other branches of real sector of economy, predetermining their progress. This example demonstrates the brake off in the structural connections of dialectic pairs of economic agents under the influence of shocks connected with external factor.

The usage of K. Marx attitude has given the chance to concentrate all known economic shocks at the first and the third stages of the GDP reproduction cycle. Within the limits of the developed approach time parameters of price shocks that has revealed their domination exclusively at stages of transitions from one wave of Kondratyev' "big cycles" to another have been determined. As a result the macroeconomic cycle is defined by reaction of economic systems to shock influences going from external factors, that confirms the hypothesis stated as impulse-transmission (in our interpretation—shock) character of cyclic development of macroeconomic dynamics.

Such approach allows to expand mathematical interpretation of behavior of dialectic pairs of economic events, processes, systems for an estimation of productivity of their reactions with the account volatile structural communications. As a result it will allow to come nearer to answer to the question: what is the macroeconomic effect for participants of dialectic interaction at coincidence (discrepancy, backlog, an advancing, reflection) amplitudes of their changes, development stages, etc.

As a result, there appears a problem for the government that should solve by the way of allocating the parameters that can reach their margins and generate the multiplying process of increasing economic and financial imbalances under the influence of the price shocks. This means the problem of national economic security and calls for the monitoring of the basic economic and financial parameters in the country as well as for the development of the government measures aimed at the prevention of the threats to national economic security.

References

1. Slutsky, E.E.: Economic and statistical works: favorites/Slutsky Evgeny; [translation from German, Italian, French, English], pp. 567–623. Eksmo (2010)
2. Irma, Adelman, F.: The Dynamic properties of the Klein-Goldberger model. Econometrica (1959)
3. Kedrov, B.M.: The Unity of Dialectics, Logic and Theory of Cognition, 3rd edn, p. 176. KomKniga (2010)

4. What went wrong with economics and how the discipline should change to avoid the mistakes of the past. The Economist, July 18th–24th (2009)
5. Reinhart, C.M., Rogoff, K.S.: This Time is Different. Eight Centuries of Financial Folly. Princeton University Press, Princeton (2009)
6. Slutsky, E.E.: The Accumulation of Accidental Causes as the Source of Cyclic Processes. Problems of Economic Conditions. Conjunctural Institute (1927). Sluckiy, E.E.: The Summation of Random Causes as the Source of Cyclic Process. Econometrica (1937)
7. Economic Essays in Honour of Gustav Cassel. Allen and Unwin, London (1933)
8. Pilipenko, Z.A.: The impact of global financial shocks on the global economy. Doctoral thesis in economics, M.V. Lomonosov' Moscow State University (2012)
9. Shiller, R.: Irrational Exuberance. Princeton University Press, Princeton (2000)
10. Shiller, R.: Irrational Exuberance, 2nd edn. Princeton University Press, Princeton, N.J. (2004)
11. Perez, Carlota: The double bubble at the turn of the century: technological roots and structural implications. Camb. J. Econ. **33**(4), 779–805 (2009)

Multi-agent Systems for Intelligent Retrieval and Processing of Information

Aneta Poniszewska-Maranda and Łukasz Gebel

Abstract Today the amount of information available in the world is so large that it exceeds significantly the perceptual capacity of a man. The progress of science and increased access to the media, in particular to the Internet, allows an increasing number of people to create and publish their own content. Refer to the information that can be found in our area of interest is limited to read a very small part of the works on the topic. Selective viewing of the contents and limited time resources prevent the users to meet their information needs. The paper presents the possibilities of multi-agent systems to realize the intelligent retrieval and processing of information.

Keywords Intelligent agent · Multi-agent system · Retrieval and processing of information

1 Introduction

For many years, the continuous, very dynamic development of information technology can be observed. Initially, computers and software were used mainly as separated units, so the complexity of the created systems was relatively small. With the appearance of Web network the technology development has been focused on the examination of the possibility of building the software that not only uses the network resources, but also becomes their part and co-founder of the Internet. Currently, the universality of network technologies has led to the creation of a huge number of applications and materials. So a large amount of information requires a new approach to how it is processing. An example of this new approach is the agent paradigm.

To obtain this purpose it is necessary to investigate the suitability of the theory of multi-agent systems to create the solution that implement intelligent retrieval and

A. Poniszewska-Maranda (✉) · Ł. Gebel
Institute of Information Technology, Lodz University of Technology, Lodz, Poland
e-mail: aneta.poniszewska-maranda@p.lodz.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_33

processing of information. The exemplary multi-agent system was proposed and it can carry out an intelligent analysis of the repositories of text documents and obtain the information that is naturally specified using the human intelligence. Evaluation of the effectiveness of proposed solutions and their architecture design process based on operation of agents will reply on the merits of the connection of theory of multi-agent systems and intelligent retrieval and processing of information.

The paper presents the possibilities of multi-agent systems to realize the intelligent retrieval and processing of information. Creating multi-agent system can perform intelligent analysis of repositories with text documents and obtain information, which naturally are determined using human intelligence. Evaluation of the effectiveness of proposed solutions and design process of their architecture based on agents answered on the merits of connection of multi-agent systems theory and intelligent information retrieval and processing.

The presented paper is structured as follows: Sect. 2 gives the outline of intelligent agents and multi-agent systems, Sect. 3 presents the intelligent information retrieval and processing methods. Section 4 deals with the idea of using the multi-agent systems in the field of intelligent information retrieval and processing and contains a proposal of multi-agent realization of sentiment analysis of text documents.

2 Multi-agent Systems

One of the most cited definition of an agent is the one given by Wooldridge and Jennings. According to this definition *agent* is a computer system located at a certain *environment*, which can take *autonomous decisions* to *achieve their design goals*. The agent is part of the environment which is able to follow. Events occurring in the environment can have an effect on the agent, but only the agent decides to carry out certain actions [1, 2].

An *intelligent agent* is one that is capable of flexible autonomous actions in order to meet its design objectives: reactivity, pro-activeness and social ability. Intelligent agents are capable of interacting with other agents, are able to perceive their environment and respond to changes that occur in it and they are able to exhibit goal-directed behavior by taking the initiative. And, of course, all these functions are made by an agent in order to satisfy their design objectives. Agents operate and exist in some environment. An agent has the ability to communicate in order to achieve better goals for themselves or for the system in which they exist [1–4].

Multi-agent system is composed of multiple interacting software components known as agents, which are typically capable of cooperating to solve the problems that are beyond the abilities of any individual member. A multi-agent system consists of a number of agents that interact with one another. In the most general case, the agents will be acting on behalf of the users with different goals and motivations. To successfully interact, they will require the ability to cooperate, coordinate and negotiate with each other, much as people do [1, 3, 5, 6].

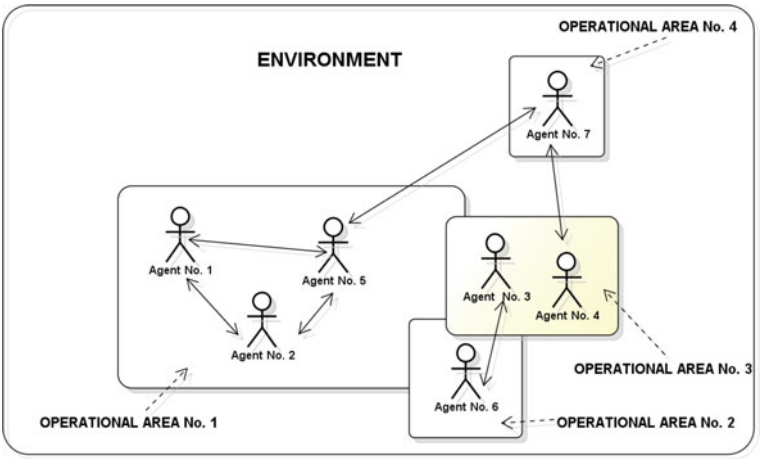


Fig. 1 General schema of multi-agent system

Multi-agent system is an information system which consists of many elements called agents which work in environment or environments. Agent is a computer program that can autonomously perform actions to meet its design objectives. General schema of such systems is presented in Fig. 1 [7–9].

3 Intelligent Retrieval and Processing of Information

Development of science and increased access to media, in particular to the Internet makes it easy to create and publish own content. Everyone can share their creativity in the form of article, book, photos or paintings. Finding and reading all of interesting materials is probably impossible but we can omit important sources of information.

Information retrieval and processing methods can be helpful to cope with such problems. This computer science domain is defined as searching for information in documents which do not have unified internal structure. Such documents are usually text documents which are written in natural language. It makes it harder to extract required information automatically, because usually natural, human intelligence is needed in such tasks. What is more such exploration of text document can be very computationally complex which creates the need for effective solutions [10, 11].

The classification process of set of documents is composed of the following stages:

- collection of documents, through which it will be possible to generate the useful classifier,
- preliminary processing of text documents, containing the stemization and tokenization of words in a sentence,

- initially processed documents are divided into test set and training set—training set is used to learn a classifier and test set to validate its effectiveness,
- classifier training with the use of extracted features or whole representations of objects—the training typically involves the presentation of examples in the form of vectors of features or objects and giving to which class they should belong to,
- testing of classifier's results on the documents coming from the test set—if the classifier has acquired in the course of training the generalizing properties then it should effectively classified the documents, which were not used for a training.

A very important element of the classification is the extraction of features. The extraction of features in text documents is mainly for finding various dependencies between the words of these documents. This is a very intuitive approach, because we can expect that certain sets of words mainly occur in the texts coming from specified domain. The best selection of features should allow as much as possible the degree of separation documents belonging to different classes. If the values of the selected features are not significantly different for documents with different labels, this prevents the correct classification.

The problem of information overload, which meets every day the average person is not associated only with the web network. It refers also the information stored in different information systems of the companies. To solve this problem the multi-agent systems which use intelligent methods of information retrieval and processing can be used.

Not many approaches of using the multi-agent systems in retrieval and processing of information can be found in literature.

The authors of [12] analyze the distributed search techniques for use in a peer-to-peer network-based Information Retrieval (IR) system. Agents have to cooperate to forward the queries among themselves so as to find appropriate agents, and return and merge the results in order to fulfill the information retrieval task in a distributed environment. In presented approach, the agent society is connected through an agent-view structure maintained by each agent and it can be significantly enhanced by dynamically reorganizing the underlying agent-view topology and deploying context sensitive distributed search algorithms.

In [13], the authors present ant information retrieval server based on ontology and multi-agent, which integrates several kind of agents, such as interface agent, pre-processing agent, management agent, information processing agent, and information searching agent with mobile ability. The system also uses ontologies to classify the domains of documents and assist user to normalize their queries. Using this system, dynamic changes of information on the Internet can be reflected timely, and the navigational ability of retrieving information can be improved.

The authors of [14] present consensus-based approach used within a multi-agent system, which assists users in retrieving information from the Internet. The consensus methods are applied for reconciling inconsistencies among independent answers generated by agents (using various search engines) for a given query. AI methods have been developed to allow the agent's self-improvement.

Multi-agent system for information retrieval and processing consist of information agents which are the units that have the ability to extract the information from given repositories, for example repository of text documents. Repository can be understood as any set of information like database, website or even the whole Internet. *Information Agents* should have an access to at least one such repository and be able to solve some kind of problem posed by the user and related to this repository [1, 9].

The user of proposed multi-agent system for information retrieval and processing instructs it to perform some task by using the communication module. This module consists of the interface, allowing the option choice to define the problem and will vary depending on the type of application. Information Agents receive the orders from communication module and try to retrieve information from repositories and process it so it will be readable and understandable to the user. Collected information is transferred to the communication module and presented to the user. Information Agents can have access to different repositories which will affect the results of their work. What is more agents can communicate with each other to improve the results.

4 Multi-agent Intelligent Algorithms of Sentiment Analysis Basing on Content of Text Documents

One of the information retrieval tasks is an analysis of sentiments contained in the content of text documents. Solution to this problem is an automatic method that will determine if a given document is polarized positively or negatively. The Internet users very often search for and read the content that contains opinions about a subject of their interests. What is more, nowadays exchanging points of view with the use of the Internet became a trend and is very common [10, 15].

It is easy to imagine a situation, when a person who is trying to make decision about buying some product, would like to know other people's opinions about it. But the problem is a huge number of available opinions, which can be found on the various websites. Reading all of these opinions is practically impossible. Similar problem concerns every domain in which user can find a big amount of opinions expressed in natural language, for instances reviews of movies, books or restaurants. It creates the need for tool that would be able to analyze the content of such opinions and intelligently determine their positive or negative polarization.

An attempt to solve the problem of automatic detection of polarization of text documents can be the approached based on classifying the feature vectors or lexicons of words. Description of these methods and proposals of their multi-agent realizations are presented in the following subsections.

4.1 Multi-agent Sentiment Analysis Algorithm Based on Classifying Feature Vectors

First solution of determining the polarization of text documents is algorithm based on classifying the feature vectors. It consists of typical stages associated with documents classification process: preprocessing, feature extraction and classification of extracted vectors.

Considering large number of analyzed documents and their various sources it seems reasonable to propose the multi-agent realization of such algorithm. It uses the intelligent agent concept and contains four types of agents:

- Manager Agent,
- Preprocessing Agent,
- Extractor Agent,
- Classifier Agent.

Manager Agent is a coordinator who manages all steps of algorithm process. He can creates another types of agents which can perform the tasks needed in particular steps of the process. Manager Agent also collects the results of Agents' work and presents it to the user.

Next type of agents are *Preprocessing Agents*. Their job is to load the text files from algorithm's input path and depending on given configuration, performing stemization of words in loaded document and removal of stop words. After preprocessing this agent returns text document to Manager Agent.

Extractor Agents have to extract the features of document and build vector of these features' representation. For the process of learning and testing agents the vectors should be labeled with tags "positive" or "negative".

Last type of agents in proposed multi-agent realization of algorithm are *Classifier Agents*. These agents are trained with the use of previously prepared feature vectors. After training they are tested to determine the evaluation of their activity. The ability of classification of these agents gives them purely autonomous character, because their decision cannot be forced by no means, but it is an effect of intelligence module which they have, so the trained classifier.

Multi-agent sentiment analysis algorithm is presented in Fig. 2. This algorithm consists of three main stages. First one is *preprocessing of documents*. Manager Agent creates specified number of Preprocessing Agents. This number can be different depending on hardware resources used for execution of algorithm. Every Preprocessing Agent works parallel with other Agents. Thereby many documents can be preprocessed at the same time. Agent who ended preprocessing of one document loads the next one until all documents will be preprocessed.

Next step is *extraction of documents features*. This time Manager Agent delegates this task to the Extractors Agents which build the feature vector with the use of chosen extraction methods like for instance word frequency or TF-IDF (*term frequency method—Inverse Document Frequency*) measure [10, 11]. Extractor

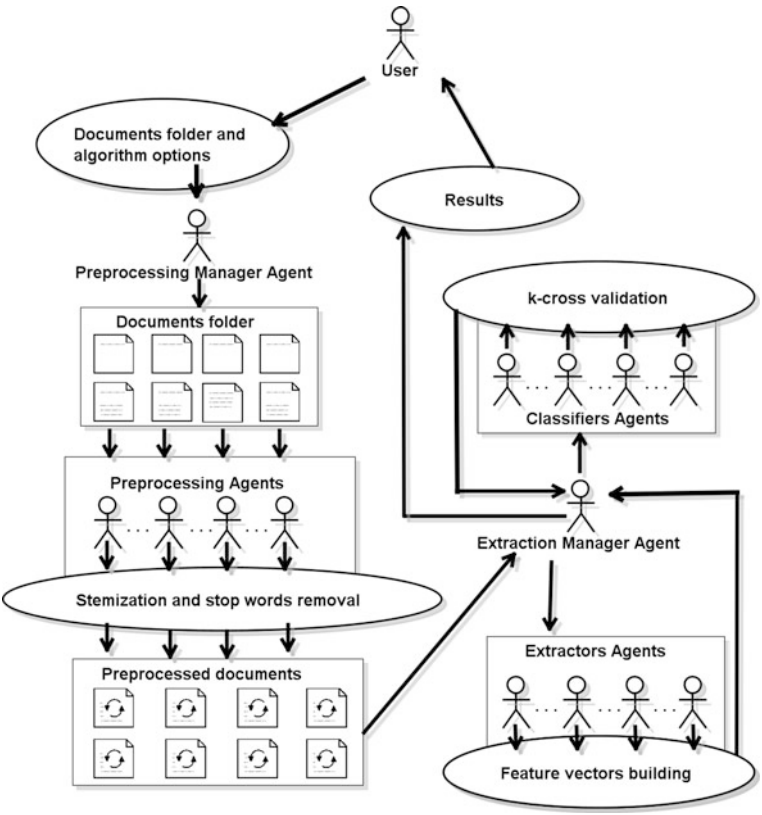


Fig. 2 Illustration of multi-agent sentiment analysis algorithm for text documents, based on classifying feature vectors

Agents work parallel until all of the documents from set of documents have the vector representation.

After getting all of feature vectors the *k-cross validation process* is turn. Its purpose is to train and test the Classifier Agents. This process involves on dividing the set of feature vectors on k equal parts in which the number of positive and negative documents are relatively close. Next, every part of the set is used for training of different classifiers being the decision-making core of Classifier Agent, while the rest part of the set is used to test the effectiveness of such agent.

At the end of the algorithm Manager Agent computes the average precision of trained Classifier Agents and returns it to the user. When the result is satisfying, one Agent can be trained with the use of bigger number of documents. Otherwise algorithm can be performed once again for another input parameters. Choice of algorithm's parameters can be done by user of a system or by Manager Agent.

4.2 *Multi-agent Sentiment Analysis Algorithm Based on Lexicon of Words*

Another way to solve the problem of automatic sentiment analyzing based on content of text documents is the use of lexicons of words. Such lexicons contain the list of words with appropriate polarization value. It means that every word in lexicon is labeled as "positive", "negative" or sometimes "neutral". There are also much more complex lexicons that assign numerical values to every word. For instance, word "fantastic" could be assigned as 1.0 or 9.0 while word "boring" could be equal to −1.0 or −9.0 [15, 16].

Using the same intuition a multi-agent realization of classification algorithm based on lexicon of words can be used. Such algorithm would be able to extended by given lexicon while working on analyzing the sentiments of text documents (Fig. 3).

The preliminary phase of documents preprocessing is identical as in algorithm presented in Sect. 4.1. New type of Agent proposed in algorithm based on lexicon of words is *Lexicon Builder Agent*. Agents of this kind search for words connected

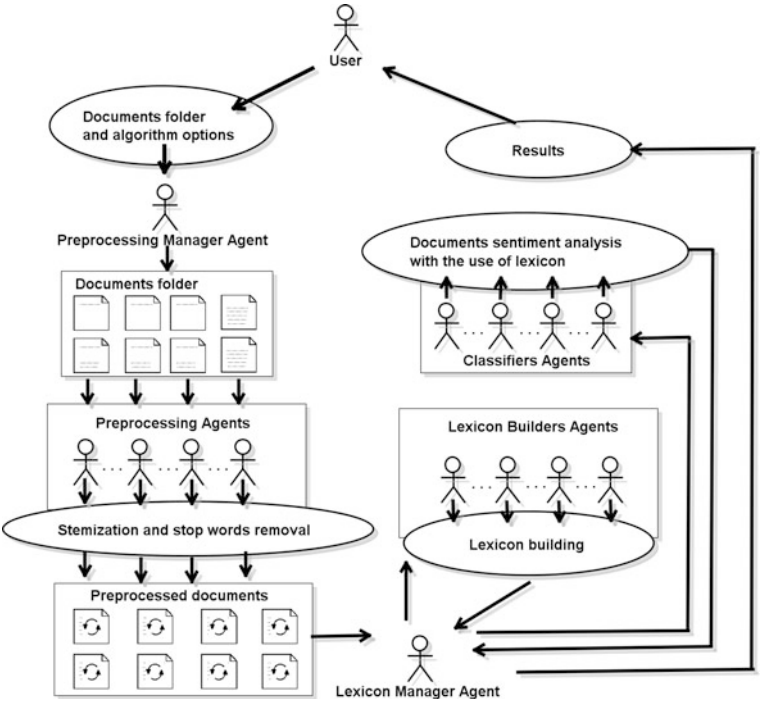


Fig. 3 Illustration of multi-agent sentiment analysis algorithm based on lexicon of words

with appropriate connectives like "and" or "but". These two connectives are just examples and list of useful connectives can be expanded and different for every language. By using particular connectives Lexicon Builder Agents are expanding the lexicon using rules about the same or different polarization.

Classifier Agents work differently then Agents in algorithm based on classifying feature vectors. They use the expanded version of word's lexicon and sum up the values of negatives and positives words in currently processed text document. If the sum of polarization is negative then document is labeled as "negative", otherwise it is labeled as "positive".

It should be noted that Classifier Agents are strictly dependent on lexicon of words. If such lexicon lacks in quality or it is very poor it is expected that the results of agents will not be satisfying. Depending on type of analyzed texts it may be necessary to build the specific lexicons just for them. Words can have different polarization if they occurs in horror book and different in book for children. It can be a key aspect and should be considered while testing the effectiveness of the algorithm.

4.3 Textlligent—*Multi-agent System for Text Document Analysis*

Functional requirements of *Texlligent* system can be divided into four groups. User can use the system for preprocessing of set of text documents and to perform the analysis process of sentiments contained in documents with the use of two methods; basing on the classification or using the lexicons. The fourth system functionality is the possibility of hierarchical clustering of documents. Each function is designed with the use of multi-agent architecture, proposed in the algorithms presented in the previous section and the user is separated from the details of system realization.

First processing module is *Documents Reading Module* which is responsible for loading the documents. The supervisor of this module is *Document Reading Manager Agent* who synchronizes work of *Document Reading Agents*. *Preprocessing Module* is used by other modules to get the documents after stemization and stop words removal. The effect of this processing module is an input for sentiment analysis modules.

Sentiment Analysis Based on Feature Vectors Classification Module is supervised by *Classification Manager Agent* who delegate *Dictionary Builders Agents*. These Agents are additional units which gather all words needed for building the feature vectors. Next, *Extractors Agents* builds the feature vectors and *Classifiers Agents* perform k-cross validation process. The final result of the module is presented to the user. *Sentiment Analysis Based on Lexicon Module* works in the similar way. It has its own *Manager Agent* which delegates work to appropriate Agents. Manager presents results to the end user of *Textlligent* system.

5 Conclusions

The analysis presented in the paper confirms that the combination of multi-agent systems theory and intelligent information retrieval and processing methods can be practically used and has many advantages. The research that was carried out proved that combination of multi-agent systems that solves intelligent information retrieval and processing problems improves a designing process of systems of big complexity as well as effectiveness and efficiency of such systems.

Regarding multi-agent approach for designing of information retrieval and processing system, its main advantage is the simplification of building model of such systems. The units-agents are useful for clear presentation and separation of responsibilities of system's elements. This approach is transparent and understandable for human perception. Building very complex systems can became simpler and will minimize the errors during this process. The exchange of communicates between Agents and general communication model of such systems can be easily designed.

On contrary, the implementation of systems designed in multi-agent manner requires more work. It is caused by big number of separated components which must communicate with each other. Additional work on communication protocol is rewarded by simplification of parallelization of processes. The use of Agents helps to design the system properly and gain parallelization out of the box.

Proposed multi-agent information retrieval approach and their practical realizations can be developed to use distributed environment with many huge text document repositories. The agents can work in cloud environments which can be another direction of further research. Universal character of presented solution makes it possible to adjust it to different requirements.

References

1. Wooldridge, M.: An Introduction to Multi-agent Systems. Wiley, Chichester (2002)
2. Brenner, W., Zarnekow, R., Wittig, H.: Intelligent Software Agents. Foundations and Applications. Springer, Berlin (1998)
3. Weiss, G.: Multi-agent Systems. A Modern Approach to Distributed Artificial Intelligence. The MIT Press, Cambridge (1999)
4. Chomątek, L., Poniszewska-Maranda, A.: Modern Approach for Building of Multi-agent Systems, LNCS (LNAI), vol. 5722, pp. 351–360. Springer, Heidelberg (2009)
5. Padgham, L., Winikoff, M.: An Introduction to Multi-agent Systems. Wiley, Chichester (2007)
6. Poniszewska-Maranda, A.: Multi-agent system to assure the logical security of data in distributed information system. J. Appl. Comput. Sci. **21**(1), 119–133 (2013)
7. Toru, I.: Mutli-agent Platforms. Lecture Notes in Artificial Intelligence, vol. 1599. Springer, Berlin (1998)
8. Young, C.I.: Intelligent multi-agent application system in an AI system. Artif. Life Robot. **12**, 6–13 (2008)
9. Azhana, A., Sharifuddin, A.M., Yusoff, M., Zaliman, M.: An Exploratory Review of Software Agents, Information Technology, 2008. ITSim 2008, vol. 3, pp. 1–8 (2008)

10. Manning, C.D., Raghavan, P., Schuetze, H.: An Introduction to Information Retrieval. Cambridge University Press, Cambridge (2009)
11. Han, J., Kamber, M.: Data Mining. Concepts and Techniques, 2nd edn. Morgan Kaufmann, San Francisco (2006)
12. Zhang, H., Croft, W.B., Levine, B., Lesser, V.: A Multi-agent approach for peer-to-peer based information retrieval system. In: Proceedings of AAMAS, International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 456–463 (2004)
13. Wu, Ch., Jiao, W., Tian, Q.-J., Shi, Z.: An information retrieval server based on ontology and multi-agent. J. Comput. Res. Dev. **6** (2001)
14. Sliwko, L., Nguyen, N.T.: Using multi-agent systems and consensus methods for information retrieval in internet. Int. J. Intel. Inf. Database Syst. **1**(2) (2007)
15. Internet course about natural language processing, https://www.coursera.org/course/nlp
16. Bo, P., Lillian, L., Shivakumar, V.: Thumbs up? Sentiment classification using machine learning techniques. In: Proceedings of EMNLP, pp. 79–86 (2002)

Architecture for Internet of Things Analytical Ecosystem

Andrzej Ratkowski

Abstract Data analysis is a crucial part of Internet of Things concept. Smart objects generate significant amount of data that should be processed in order to utilize full potential of IoT. The paper presents proposed architecture that enables creating data analysis ecosystem for Internet of Things (IoT). The purpose of that architecture is to separate data analysis problem from orchestration of smart objects. Another important benefit of the ecosystem is to share analysis logic for IoT without revealing analytical algorithms.

Keywords Internet of Things · IoT · Data analysis for IoT

1 Introduction

Internet of Things (IoT) as a vision of a world-wide network of interconnected objects [1] is a very promising concept however the concept is still in an early stage of development. IoT was defined by Haller [2] as "*a world where physical objects are seamlessly integrated into the information network, and where the physical objects can become active participants in business processes. Services are available to interact with these 'smart objects' over the Internet, query and change their state and any information associated with them, taking into account security and privacy issues*". It is estimated, that quantity of IoT objects potentially interconnected via information network, will be 10–100 times larger than it is now in contemporary Internet of PCs and mobile devices [3]. Such population of devices will produce wide stream of data that should be analyzed and utilized to orchestrate objects behavior. Hence analysis should be performed in near-real time and data flow should be processed fast and efficiently. This is the motivation to elaborate specific architecture that will support data flow and analysis. Important feature of

A. Ratkowski (✉)
Institute of Control and Computation Engineering,
Warsaw University of Technology, Warsaw, Poland
e-mail: A.Ratkowski@elka.pw.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_34

the proposed architecture is entity that possess knowledge on analytical algorithms should be able to share that knowledge without revealing algorithms itself.

2 State of the Art

Data analysis for IoT domain is a vital research problem. It is believed that the key to real usability of IoT paradigm is ability to provide real time data from various different distributed sources to other machines, smart entities and people for a variety of services [4]. Data analysis area for IoT covers data gathering and exchange [5], analysis algorithms used for data abstraction [6], semantic representation and cognitive analysis [7], data privacy and security [8]. Finally data analysis result could be also useful in orchestration of smart objects [9].

Literature review shows that data analysis for IoT is a complex issue, hence analysis should be separated from other aspects. It would be also economic to reuse algorithms without repeating effort on algorithm elaboration by using algorithm sharing ecosystem similar to mobile application ecosystems like Google Play [10].

Due to the amount and heterogeneity of data generated by smart objects it is recommended to use Big Data technologies [11] that support processing of data that have volume, velocity and variety properties [12].

Reference architecture for IoT was elaborated by IoT-A project [13] however the architecture does not address detailed data analysis aspect.

3 Analytical Ecosystem Vision

3.1 IoT Ecosystem Use Case

Following example illustrates use case of IoT ecosystem. A home is equipped with smart home installation based on IoT principles. The installation consist of sensors (light, temperature, air-quality, human presence etc.) and actuators (light switches, heating, air-conditioning etc.). Installation uses also data gathered from home residents smartphones and social services (facebook etc.). All sensors and actuators are connected to a local hub that coordinates their actions. The local hub is a computer system that stores set of simple rules that control actuators on the basis of facts gathered from sensors. Example of rules: if a person is present in the room and temperature drops below 20 °C then turn on heating; if a house is empty and temperature drops below 15 °C then turn on heating; etc.

The problem is how to generate set of rules with a view to comfort of living, energy saving, security and so on. It can be assumed that there is a data analytic algorithm *shRuleGenerator* possibly based on heuristics that can gather data received by sensors and generate such rules. The algorithm can infer the model of

the environment (house) on the basis of received data. For example, heat capacity of each room—analyzing heating/warming rate; or personal heat perception—analyzing feedback on temperature feeling from residents smartphones.

Such algorithm should be the same for certain type of houses (for flats, detached houses), so every owner of such IoT smart home installation do not have to develop his own algorithm but can use general one available in analytical ecosystem. Local hub collects historical data from sensors and sends them periodically to analytical ecosystem.

Analytical ecosystem enables exchange of algorithms between consumers and suppliers. Analytical ecosystem provides consumer with CPU power and data storage for algorithms execution. Consumer does not know algorithm but is able to execute it on provided data. Algorithm supplier does not know data feed by consumer but is aware of usage of algorithms.

3.2 Requirements

Architecture and data model has to satisfy following functional and non-functional requirements of analytical ecosystem:

RQ1. All services of analytical ecosystem for consumer and supplier are available through APIs.
RQ2. Supplier is able to provide and register algorithms.
RQ3. Algorithm is expressed in a convenient way in language preferred by supplier.
RQ4. Algorithm can execute (call) other algorithms.
RQ5. Consumer is able to find desirable algorithm.
RQ6. Consumer is able to execute desirable algorithm on provided data
RQ7. Consumer knows semantics of algorithm but does not know algorithms itself.
RQ8. Supplier does not know data that are feed by consumer.
RQ9. Both consumer and supplier are able to get information about usage of algorithms (number of executions).
RQ10. Consumer is billed for number of processed records, storage usage and algorithm execution.
RQ11. Supplier is billed for CPU used by his algorithms.
RQ12. Supplier is rewarded for usage of his algorithms.
RQ13. Analytical ecosystem supports semantic transformation and normalization of data feed by consumer.
RQ14. Analytical ecosystem provides anonimisation and privacy security mechanisms.
RQ15. Architecture is scalable in order to serve changing stream of execution of algorithms and to store input data for algorithms.

4 Analytical Ecosystem Architecture

4.1 Functional Architecture

Figure 1 depicts functional architecture of analytical ecosystem. Architecture consist of seven modules. Each module is characterized in next sections. Fulfillment of requirements is marked with (RQxx) mark.

4.1.1 Consumer API

Consumer API is a set of functions that provides consumer with all ecosystem services (RQ1). It consists of following function subsets:

- algorithm register querying and evaluation
- execute algorithm with provided data (RQ6)
- feed data and receive result of execution (RQ6)
- storage management
- billing and accounting information (RQ9) (Table 1).

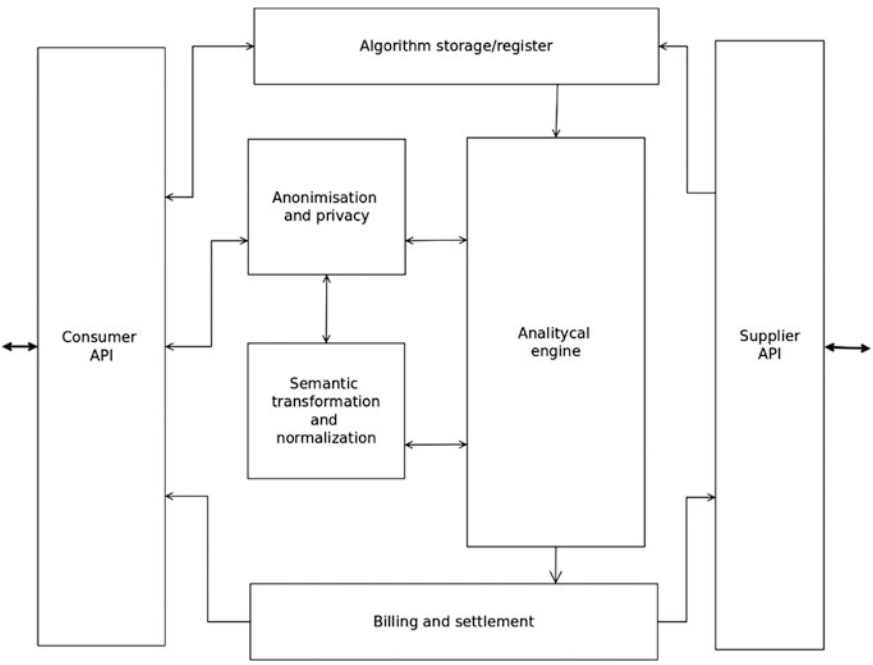


Fig. 1 Analytical ecosystem functional architecture

Table 1 Main consumer API functions

Function	Specification
algorithmSet findAlgorithm (criteriaSet)	Queries register based on criteria, returns set of algorithms that meets supplied criteria
void scoreAlgorithm (algorithmId, score)	Scores algorithm
setId createDataSet()	Creates new data set identified by *setId*
void feedData(setId, entityCollection, size)	Feeds data set *setId* with collection of entities
resId executeAlgorithm(setId, algorithmId)	Executes algorithm *algorithmId* on data set *setId*, returns id of result set
resultSet getResultSet(resId)	Fetches result set *resId*
void releaseDataSet(setId)	Empty and free space occupied by data set *setId*
void releaseResultSet(resId)	Empty and free space occupied by result set *setId*
account getCurrentAccount()	Returns account balance and list of billing items.
billingData getBillingDetails (resId)	Returns detailed information on billing item

4.1.2 Supplier API

Supplier API provides supplier with all ecosystem services (RQ1). It consists of following function subsets:

- algorithm registering (RQ2)
- billing and accounting information (RQ9)
- algorithms usage statistics and CPU usage (Table 2).

Table 2 Main supplier API functions

Function	Specification
algorithmId registerAlgorithm(algorithmCode, algorithmDescription)	Registers algorithm and labels it with *algorithmDescription*
void unregisterAlgorithm(algorithmId)	Deletes algorithm *algoritmId* from register
void updateAlgorithm(algorithmId, algorithmCode, algorithmDescription)	Updates previously registered algorithm
retCode testAlgorithm(algorithmId)	Tests syntax correctness of registered algorithm
account getCurrentAccount()	Returns current account state
billingData getBillingDetails(algorithmId)	Returns billing data for specified algorithm

4.1.3 Algorithm Storage/Register

Algorithm storage/register is responsible for registering by supplier and sharing for consumer of analytical algorithms (RQ2, RQ5). Algorithms are defined by name, semantical description, price per execution and analytical algorithm. Algorithm itself can be expressed in language chosen from set of languages supported by ecosystem (RQ3). Among those languages we can find general purpose languages, (e.g. Python, Scala) and data analysis oriented (e.g. R, Pig Latin). Algorithm storage also provides data about usage popularity of algorithm, average CPU usage and reputation based on consumers evaluation scores.

4.1.4 Billing and Accounting

Billing and accounting module holds and provides data on mutual settlements between supplier, consumer and ecosystem itself. Billing and accounting module is feed with data from analytical engine on algorithms execution and resources used by every execution. On the basis of received data billing and accounting module calculates charges from consumer and rewards for supplier for executions of his algorithms(RQ10, RQ11, RQ12).

4.1.5 Semantic Transformation/Normalization

The module is responsible for mapping data feed by consumer into semantically normalized form. For example smart home installation feed raw readouts from temperature sensors. Such readouts should be transformed into more synthetic information on temperature in each room. Semantic transformation/normalization module can provide such mapping. The mapping should use relatively simple algorithms in order to be fast and consume little CPU resources (RQ13).

4.1.6 Privacy and Security

Privacy and security module scrambles data that can be used to identify analyzed objects or contains sensible information (e.g. localization data). The module hashes such data and stores mapping of hashed data into primary data. Module is also responsible for decoding hashed result of analysis into primary data. Scrambling should be CPU effective (RQ14).

4.1.7 Analytical Engine

Analytical engine is core element of the ecosystem. It is responsible for executing algorithms and allocating CPU resources for execution. The analytical engine is

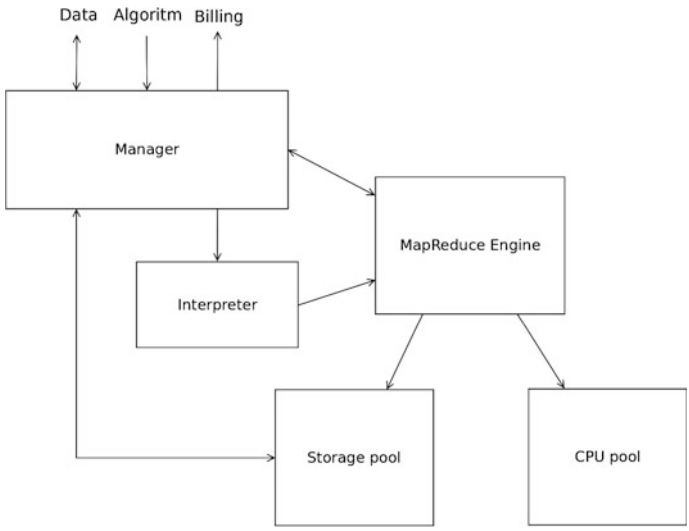


Fig. 2 Analytical engine internal structure

based on MapReduce [14] programming model (RQ15). Analytical engine ensures isolation of data between different consumers through multi-tenancy architectural pattern [15]. Internal structure of analytical engine is depicted on Fig. 2.

Manager element is responsible for coordination of algorithm execution, requests scheduling and calculation of executions and resource utilization. *Interpreter* translates algorithm that was provided by supplier in desired language, to set of *map* and *reduce* functions. *Manager* After translation *MapReduce* component is requested by *Manager* to execute algorithm. *MapReduce* is also responsible for resources allocation.

4.2 Data Model

Figure 3 presents data model of analytical ecosystem. Data model organizes information on algorithms (*Algorithm*). Property of algorithm entity is algorithm code expressed in certain language. Each algorithm is also characterized by input and output data structure (*DataFormat)*. All algorithms are referenced in directory (*AlgorithmDirectory*) that can be queried in order to find fitting algorithm.

Data model stores data on algorithms execution (*AlgorithmExecution*) as well as input and output data sets (*DataSet, ResultSet*) for each execution. Input and output data format has to be consistent with proper data formats associated with algorithm. Data model stores data on resources consumed by algorithm executions (*Resources*)

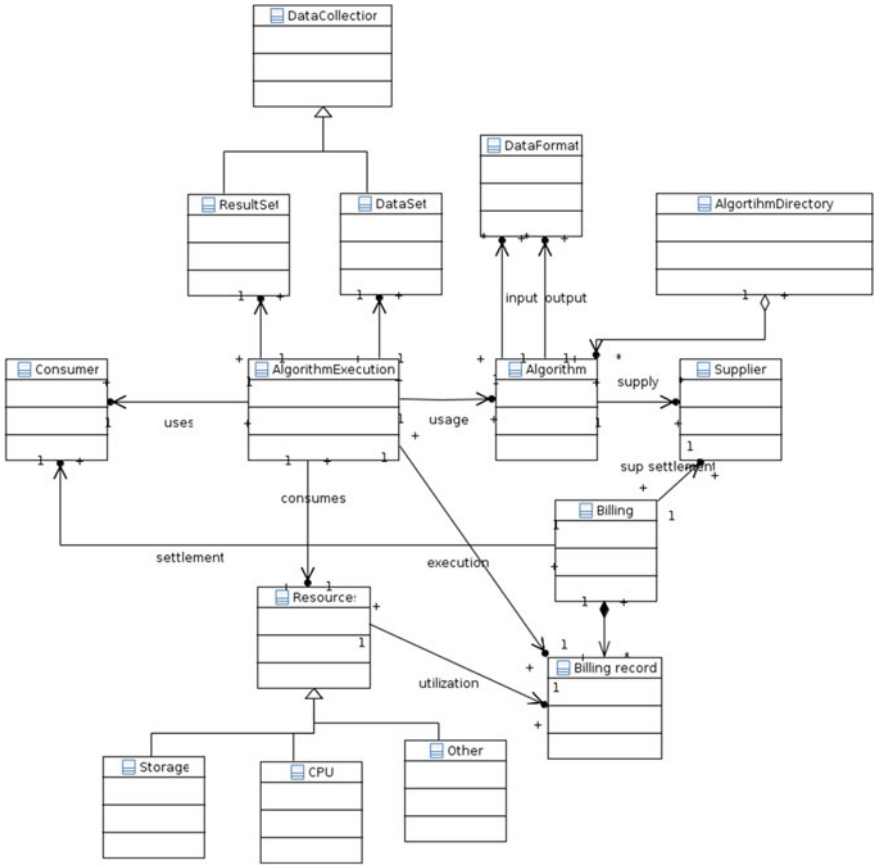


Fig. 3 Data model of analytical ecosystem for IoT

and settlements between algorithm ecosystem, consumer and supplier (*Billing, BillingRecord*). Data model does not cover information on algorithm semantics. This aspect is handed over to supplier.

5 Summary and Further Work

Presented architecture fulfills requirements stated in Sect. 3.2. Analytical ecosystem allows sharing of algorithms between supplier and customer without compromising security of data feed by customer and without revealing algorithm code. Such defined analytical ecosystem plays *trusted third party* role in interactions between algorithm supplier and customer.

Despite of the fact that proposed architecture is on certain maturity leve, following aspects require more detailed consideration:

- physical architecture that covers allocation of modules to physical resources (servers, storages etc.),
- interfaces of architectural modules,
- operational model of *Interpreter* module that will enable using different languages to express algorithms and to execute other algorithms (according to requirement RQ4),
- detailed design of consumer and supplier API including protocols and semantics of functions calling.

The next step of validation of ecosystem concept will be preparing working prototype of full functional ecosystem. The working prototype will be utilized to verify feasibility and usability of ecosystem.

References

1. Santucci, G., Lange, S.: Internet of Things in 2020 a Roadmap for the Future, Joint EU-EPoSS Workshop Report (2008)
2. Haller, S.: The Things in the Internet of Things. In: Proceedings of Internet of Things Conference (2010)
3. Cicconi, J., At&T Cisco IBSG, et al.: The Internet of Things Infographic (2012)
4. Aggarwal, C., Ashish, N., Sheth, A.: The Internet of Things: a survey from the data-centric perspective. In: Managing and Mining Sensor Data. Springer (2013)
5. Abdelwahab, S., Hamdaoui, B., Guizani, M., Rayes, A.: Enabling smart cloud services through remote sensing: an internet of everything enabler. Internet Things J. IEEE **1**(3), 276–288 (2014)
6. Ganz, F., Puschmann, D., Barnaghi, P., Carrez, F.: A practical evaluation of information processing and abstraction techniques for the Internet of Things. Internet Things J. IEEE **2**(4), 340–354 (2015)
7. Wu, Q., Ding, G., Xu, Y., Feng, S., Du, Z., Wang, J., Long, K.: Cognitive Internet of Things: a new paradigm beyond connection. Internet Things J. IEEE **1**(2), 129–143 (2014)
8. Keoh, S.L., Kumar, S.S., Tschofenig, H.: Securing the Internet of Things: a standardization perspective. Internet Things J. IEEE **1**(3), 265–267 (2014)
9. Haller, S., Magerkurth, C.: The real-time enterprise: Iot-enabled business processes. In: IETF IAB Workshop on Interconnecting Smart Objects with the Internet (2011, March)
10. Google Inc. Play application store. https://play.google.com/ (2015)
11. Marz, N., Warren, J.: Big Data: Principles and Best Practices of Scalable Realtime Data Systems, 1st edn. Manning Publications Co., Greenwich, CT, USA (2015)
12. Russom, P.: Big data analytics. TDWI Best Practices Report, Fourth Quarter (2011)
13. Bassi, A., et al.: Final Architectural Reference Model for the IoT v3.0 (IoT-A). www.iot-a.eu (2011)
14. Dean, J., Ghemawat, S.: MapReduce: simplified data processing on large clusters. Commun. ACM **51**(1), 107–113 (2008)
15. Kwok, T., Nguyen, T., Lam, L.: A software as a service with multi-tenancy support for an electronic contract management application. In: IEEE International Conference on Services Computing, SCC'08, vol. 2, pp. 179–186. IEEE (2008)

Optimal Path Evolution in a Dynamic Distributed MEMS-Based Conveyor

Haithem Skima, Eugen Dedu, Julien Bourgeois, Christophe Varnier and Kamal Medjaher

Abstract We consider a surface designed to convey fragile and tiny micro-objects. It is composed of an array of decentralized blocks that contain MEMS valves. We are interested in the dynamics of the optimal path between two blocks in the surface. The criteria used for optimal paths are related to the degradation of the MEMS, namely its remaining useful life and its transfer time. We study and analyze the evolution of the optimal path in dynamic conditions in order to maintain as long as possible a good performance of the conveying surface. Simulations show that during usage the number of optimal paths increases, and that position of sources greatly influences surface lifetime.

1 Introduction

Conveyors have a widespread use across numerous industries where objects should be moved, including the automotive, computer, agricultural, etc. Using conveyors is much safer than using a forklift or other machines to move objects. They enable safe transportation of objects from a start point to a given destination, which if done by

H. Skima (✉) · E. Dedu · J. Bourgeois · C. Varnier · K. Medjaher
Institut FEMTO-ST, UMR CNRS 6174 - UFC/ENSMM,
Université Bourgogne Franche-Comté (UBFC), 15B av. des Montboucons,
Besançon, France
e-mail: Haithem.Skima@femto-st.fr

E. Dedu
e-mail: Eugen.Dedu@femto-st.fr

J. Bourgeois
e-mail: Julien.Bourgeois@femto-st.fr

C. Varnier
e-mail: Christophe.Varnier@femto-st.fr

K. Medjaher
e-mail: Kamal.Medjaher@femto-st.fr

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_35

human labor would be strenuous and expensive. Most of the existing solutions to convey objects in production lines rely on contact-based technologies. However, they are not appropriate for fragile and tiny micro-objects (medicines, micro-electronics parts, etc.), which can be easily damaged, contaminated or even scratched during conveying. Thus, conveyors based on contactless air-jet technology, which avoid contact with conveyed objects, can be a solution in this case [3, 5, 11].

A conveyor generally consists of a single monolithic block dedicated to a specific task in a fixed environment. As a consequence, in case of failure or environment change, the conveyor will be not able to perform the dedicated task and has to be replaced. To address these issues, self-reconfigurable systems, which consist generally of small MEMS-based modules, can be used [12, 16].

A Micro-Electro-Mechanical System (MEMS) is a *microsystem* that integrates *mechanical* components using *electricity* as source of energy in order to perform measurement functions and/or operating in structure having micrometric dimensions. Thanks to their miniaturization, low power consumption and tight integration with control and sense electronics, MEMS devices come in a wide variety of fields such as aerospace, automotive, biomedical and communication technologies. Classical MEMS include accelerometers, gyroscopes, pressure sensors and micromirror arrays.

The work presented in this paper is a part of a project which aims at increasing the efficiency of future production lines. Our ongoing project consists in designing a contactless distributed MEMS-based conveying surface for safe and fast conveying of fragile and tiny micro-objects. The conveying principle consists in sending micro-objects from a start block to a final destination using controlled air flow coming from MEMS valves. To do so, all MEMS valves involved in conveying the micro-object have to be in a good health state and able to accomplish the mission. However, and according to the literature, the reliability is one of the major concerns of MEMS [15]. They suffer from various failure mechanisms [17, 19] which impact their performance, reduce their lifetime, and the availability of systems in which they are utilized. This highlights the need to monitor their behavior, assess their health state, estimate their remaining time before failure (RUL, Remaining Useful Life) and take appropriate decisions accordingly such as control reconfiguration and maintenance [18]. These tasks can be done by using Prognostics and Health Management (PHM) approaches [7, 8, 14]. Therefore, in our project, PHM is applied to monitor the degradation of the MEMS valves and estimate their remaining time before failure with the aim at predicting failures in blocks.

In this paper, and in order to maintain as long as possible a good performance of the conveying surface, we study the evolution of the optimal path in dynamic conditions to find out how to optimize the use of the conveying surface (lifespan and conveying speed). In order to find the optimal path that avoids degraded blocks, we use the well known Dijkstra's algorithm. It optimizes a criterion related to the degradation of the MEMS valves, which is the RUL, and in case of equality the transfer time, which is also related to degradation. The values of these two criteria can be obtained by applying PHM.

To the best of our knowledge, *no article in the literature* deals with increasing conveyor lifespan by avoiding fatigued blocks. However, a similar problem appearing in the literature is the lifetime extension of multi-hop wireless networks from energy point of view [2, 4, 9]. During packet routing, without special precaution, some nodes are used more often than others, and their energy may be exhausted faster the others. To avoid this, packets need to be routed through the optimal path which optimizes the energy of nodes in the network. As in our case, oscillation between optimal paths has been noticed, without further analysis. Moreover, there are differences compared to our problem. In network case, sources and destinations are spread over the whole surface, whereas in conveyors only one or a few nodes (usually on the border of the surface) can act as source or destination, leading to specific usage patterns. Also, in network case oscillations can reduce the lifetime of the network, which is not the case for us. Finally, in network case the lifetime is defined as the duration of time before the first node fails due to the battery exhaustion, whereas in our case the algorithm allows the conveying surface to still operate by avoiding failed blocks.

A second similar problem we have found in the literature is about finding optimal paths in transportation systems. [6] and its related work are about algorithms of finding optimal paths in real transportation systems formed by roads and cars, modeled as stochastic time-dependent (STD) networks. In those systems, many cars exist on a road, so that traffic flow propagates both in time and space (a congestion influences its region for some time, and also the region around it), so congestion is an important parameter, whereas in our case only one object at a time exists in the whole system. Also, external conditions, such as accidents and thunderstorms, affect the model (e.g. travel time), whereas our model is not affected by external factors. Travel time of cars on a road depend on time (i.e. more cars at morning and evening than during the night), whereas in our case it depends on its degradation, which depends on its usage. Cars are driven by humans who add uncertainty, whereas in our case all objects follow specified unstochastic algorithms. To conclude, the model used in our work is simpler and also *different*, making these works useless in our case. Furthermore, our paper is not about finding the optimal paths, but about their evolution during usage.

For clarity of presentation, this paper does not address the PHM part, but only the evolution of the optimal path. After the introduction, Sect. 2 presents the distributed MEMS-based surface. Section 3 introduces the simulator and presents the results. Section 4 concludes the paper.

2 The Distributed MEMS-Based Conveyor

The conveying surface is composed of an array of decentralized blocks, called smart blocks. Figure 1 shows a design scheme of a smart block. Currently, the blocks are in manufacturing phase. Each block contains a micro-controller, a sensor

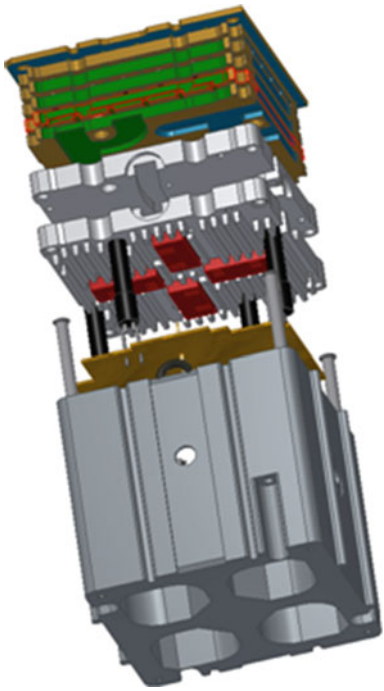


Fig. 1 Design scheme of a smart block

to detect if an object is above it, a power supplier, a MEMS valve and is able to communicate with its four block neighbors thanks to the blinky block [10].

The MEMS used to generate air flow is an electrothermally-actuated valve designed by DunAn Microstaq, In. (DMQ), company. To predict the remaining time before failure of this micro-system, we have first to define its degradation model. This model can be obtained experimentally through accelerated lifetime tests or given by experts of the MEMS. It is influenced by drifts of the physical parameters of the MEMS (friction coefficient, stiffness, etc.) and can be obtained by analyzing the collected data from tests (evolution of the parameters as a function of time) by using appropriate modeling tools such as regression and curve fitting.

Currently, we are performing accelerated lifetime tests to define the degradation model of the targeted MEMS. The simplest and most useful accelerated lifetime tests to define the degradation model of a MEMS is to stress it by applying a square signal (cycling) [13]. To do so, we have already designed an experimental platform where several MEMS are continuously cycling with a square signal of 8 V magnitude and 1 Hz frequency. We are performing measurements every day and at each measurement we estimate the values of the physical parameters. We will keep repeating this process until the occurrence of a failure.

As in the macro-systems, the degradation of MEMS devices can be modeled by linear or nonlinear functions [15]. In this work, we aim at studying the evolution of the optimal path in a dynamic conveying surface. So, while waiting to have

complete experimental data to define the degradation model, we assume that the degradation of the MEMS valve is given by a linear function (see Eq. 1).

Let's suppose that the conveying surface is composed of m blocks, each one containing a MEMS valve denoted as M_k ($k \in [1, m]$) and using the following model:

- *A linear degradation model* $deg(M_k)$:

$$deg(M_k) = 0.01 \times C(M_k) \tag{1}$$

  where $C(M_k)$ is the number of cycles performed by the MEMS M_k.
- *A RUL value* $RUL(M_k) \in [0, 100]$: the remaining time before failure expressed in number of cycles. In practice, this value is calculated using the degradation model and prognostics methods (particle filter [20], Kalman filter [1], etc.). In this work, we assume that the RUL value, which depends on the degradation level, is calculated as follows:

$$RUL(M_k) = (1 - deg(M_k)) \times 100 \tag{2}$$

  We consider that a new MEMS without degradation can perform 100 cycles.
- *A transfer time of the object* $t(M_k) \in [1, 11]$: the time that takes an object to traverse the block and reach the next block. A new MEMS without degradation has a transfer time of 1 s. The transfer time value is calculated as follows:

$$t(M_k) = 1 + deg(M_k) \times 10 \tag{3}$$

To convey an object, a MEMS can communicate only with its four neighbors and can send the object only to one of them. Figure 2 shows the connection between MEMS M_a and its four neighbors M_b, M_c, M_d and M_e (one MEMS per block).

Figure 3 gives an overview of the surface. The block denoted as S represents the source of the object that is going to be moved towards the block denoted as D, which represents the destination of the object. The black blocks represent a path taken by the object.

Let p be an optimal path from S (Source) to D (Destination). The path p is a set of n MEMS that participate at conveying the object, $p = \{S, M_2, \ldots, M_{n-1}, D\}$, where S corresponds to M_1 and D to M_n. The following equations represent respectively the RUL and the transfer time of the path:

$$RUL(p) = \min_{k=1,\ldots,n} RUL(M_k) \tag{4}$$

$$T(p) = \sum_{k=1}^{n} t(M_k) \tag{5}$$

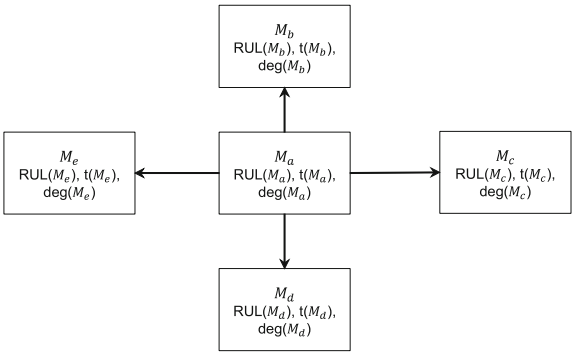


Fig. 2 Representation of the connection between one MEMS and its four neighbors

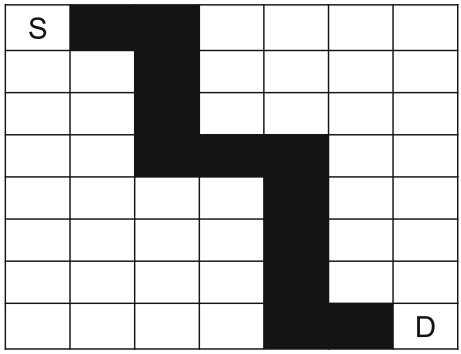


Fig. 3 Representation of a path taken by an object from the source block (S) to the destination block (D)

The main objective is to find a path that maximizes the RUL ($RUL(p)$) and minimizes the transfer time of the objects ($T(p)$) in the conveying surface.

3 Simulation Results

3.1 Simulator Features

We have developed a simulator to analyze the optimal path evolution in dynamic conditions. It is written in Java language and is multi-threaded (each block is a thread). It allows to choose the dimensions of the conveying surface, the number of objects to introduce on the surface, their source(s), and the principal criterion (RUL or time). It creates the surface with random values for both criteria in each block.

Objects are introduced at the given source(s). Each time a MEMS M_k participates at conveying an object, its number of cycles $C(M_k)$ is incremented. As a consequence, its degradation value *deg* increases, its *RUL* decreases and its transfer time *t* increases, cf. (1), (2) and (3) respectively. Hence, RUL and transfer times of blocks change dynamically during the simulation.

Each block has a matrix of the same size as the surface. Each cell of the matrix contains the RUL and transfer time of the corresponding block in the surface. Before starting the simulation, each block communicates with their four neighbors and sends them its matrix; after some time all the blocks have the same matrix which contains the right values of the surface. Once this step is finished, the first object is sent in the surface. Blocks execute asynchronously the algorithm shown in Algo. 1. The big advantage of being asynchronous is that the surface does not need a global clock for all the blocks, which facilitates greatly the surface manufacturing.

Algorithm 1 The algorithm executed asynchronously by each block.

```
1: if the object is above the block then
2:     execute Dijkstra's algorithm with itself as start block, thus finding out the next
       block
3:     send the object to the next block
4:     consequently, its degradation increments
5:     update its matrix by changing the values (RUL and transfer time) of its own cell
6: end if
7: send its matrix to its four neighbors, so that the next block have always the updated
   matrix
```

When the object is in the destination, it gets out of the surface. During this time, the updated matrix is propagating to the other blocks. We assume that information exchange is much faster than object moving, hence the source receives the updated matrix before the object gets out. A new object enters the system as soon as the previous one got out of the surface, so that only one object exists on the surface.

The simulation consists in sending the given number of objects from a given source to the destination. Figure 4 shows the initial surface used in all simulations.

95 – 1.5 (0,0)	98 – 1.2 (0,1)	57 – 5.3 (0,2)	23 – 8.7 (0,3)	64 – 4.6 (0,4)
14 – 9.6 (1,0)	44 – 6.6 (1,1)	16 – 9.4 (1,2)	88 – 2.2 (1,3)	58 – 5.2 (1,4)
79 – 3.1 (2,0)	83 – 2.7 (2,1)	27 – 8.3 (2,2)	83 – 2.7 (2,3)	22 – 8.8 (2,4)
44 – 6.6 (3,0)	98 – 1.2 (3,1)	72 – 3.8 (3,2)	96 – 1.4 (3,3)	99 – 1.1 (3,4)

Fig. 4 Initial surface used in simulations. It contains 20 MEMS and each one is characterized by RUL (*left*), transfer time values (*right*) and position

Blocks are characterized by RUL values of MEMS (left), transfer time values (right) and position.

Note that the multi-threading does not change the results, so the program is still deterministic. The results in our case mean the path along which objects go. We prove in the following that the path is the same, no matter if there is multi-threading or not. So we need to prove that when an object enters a block, the next block will be the same, no matter if multi-threading or not. As written in Algo. 1, when an object arrives in a block, the block executes the lines 2–5, i.e. it executes Dijkstra's algorithm using its matrix. During object movement, the only modification in the surface is that the values (RUL and transfer time) have been modified for the previous block. Its matrix is received from its neighbours (previous block included) much faster than object move, as written above, hence the matrix has up to date values. So, since the matrix is updated, the next block is always the same (i.e. deterministic).

3.2 Choice of Principal Criterion

As mentioned before, two criteria need to be optimized and one of them has to be set as principal one. Since we aim to improve at maximum the lifetime of the conveying surface, we set the RUL as principal criterion, and the transfer time of the object is used only if multiple paths have the same value of RUL.

Figure 5a, b show the evolution of the transfer time and the path RUL value as a function of the number of objects, using each time a different principal criterion and otherwise the same conditions. We notice that we have greater path RUL values and almost the same transfer time if the RUL is used as principal criterion comparing using transfer time as principal criterion.

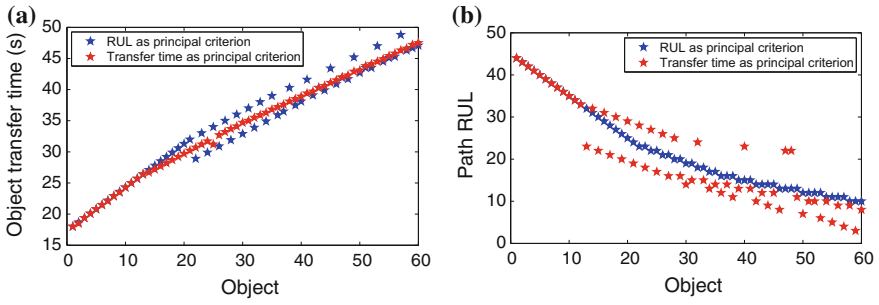


Fig. 5 **a** Object transfer time and **b** path RUL as a function of number of objects when principal criterion changes

3.3 *Optimal Path Evolution*

To study the evolution of the path according to the change of the RUL and transfer time, three types of simulations are done: one source, two sources and several sources. For all simulations only one destination is used, at position (3, 4).

For **one source**, the simulation consists in sending objects from the source (0, 0) to the destination (3, 4). In Fig. 6, arrows indicate some paths taken by objects. The thickness of arrows is proportional to the number of times the path is used. The first object takes the path indicated in Fig. 6a. This path has a RUL value of 44 cycles and a transfer time value of 18 s (the object takes 18 s to reach the destination). The next 20 objects take the same path.

For the 22th object, a new optimal path indicated in Fig. 6b is taken (the dashed arrow indicates the previous path). This is explained by the fact that the block (1, 1) is used 21 times, the RUL value of its MEMS decreases to 23 cycles and the transfer time of the path increases to 32 s. The new optimal path has the same RUL value of 23 cycles but a lower path transfer time of 28 s.

Then, paths oscillate seven times between the new optimal path and the previous one. Figure 6b shows the updated RUL and transfer time values in the surface after sending 21 objects. The 36th object takes a new optimal path indicated in Fig. 6c and which oscillates 2 times with the two previous optimal paths. Figure 6c shows the updated value after sending 35 objects. The same thing for the other objects, once a new optimal path is found, it oscillates with the ancient paths.

For **two sources**, objects enter through two sources (0, 0) and (3, 0) alternatively (Round Robin). Unlike the previous simulation, optimal paths oscillate more or less randomly because the blocks are used by multiple objects. Figure 7 shows all optimal paths taken by the different objects.

For **several sources** (all the blocks on the left side of the surface are sources), optimal paths change randomly due to multiple objects that enter through the different sources which decreases the RUL and so the change of the optimal path (no oscillation between paths). Figure 8 shows the results of this simulation.

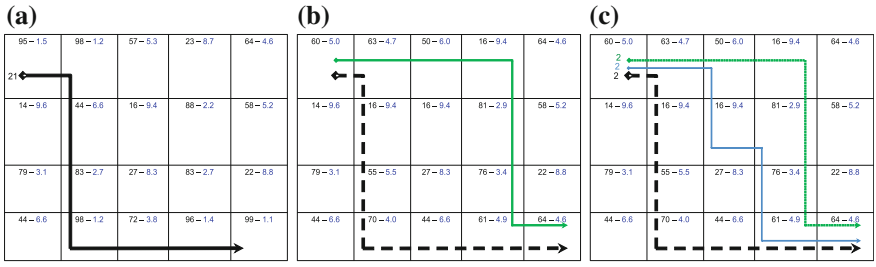


Fig. 6 Optimal path evolution for one source: objects enter through (0, 0) (the number on *left of arrows* indicates the number of times the path is used in the oscillation). **a** Oscillation 1, **b** Oscillation 2, **c** Oscillation 3

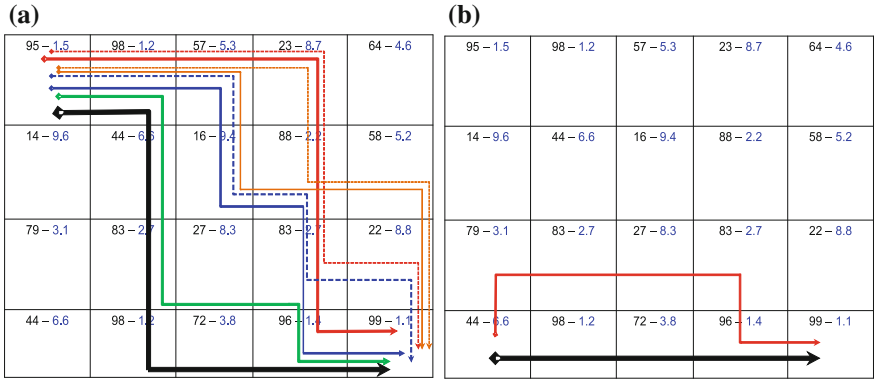


Fig. 7 Optimal path evolution for two sources: **a** paths used when objects enter through (0, 0) and **b** paths used when objects enter through (3, 0)

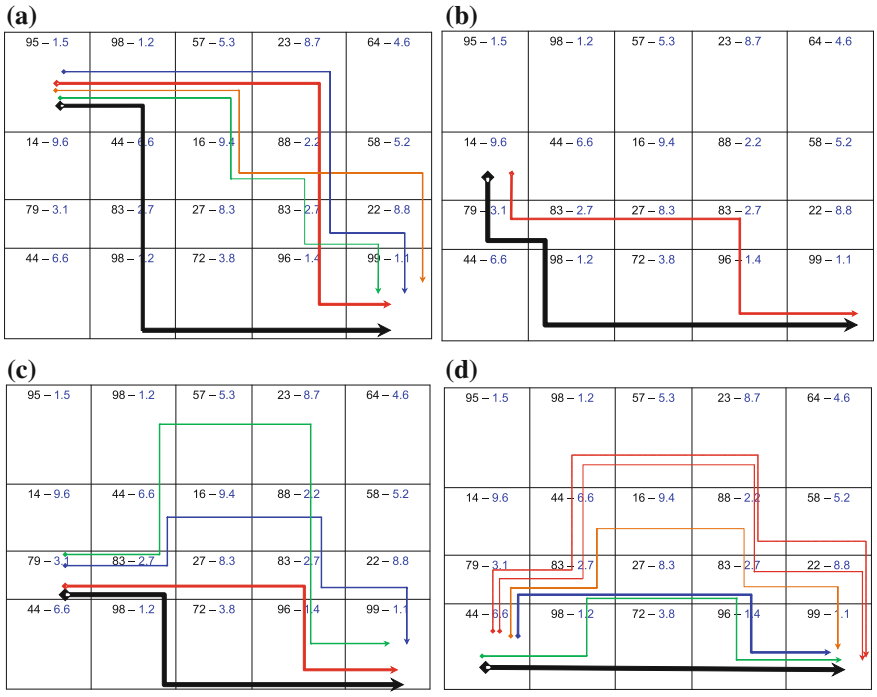


Fig. 8 Optimal path evolution for several sources: **a–d** present all the paths used for the four sources

These simulations show that:

- *Number of optimal paths*: During usage, a first optimal path is used for a number of times, afterwards a new optimal path appears and the two optimal paths are used for a number of times, afterwards another new optimal path appears and the three optimal paths are used etc.
- *Oscillation among optimal paths*: We define an oscillation as the interval of time where the optimal path oscillates (alternates) among several paths, for example between paths p_1 and p_2 from time t_1 to t_2. In case of one source, once a new optimal path appears, it oscillates with the previous optimal paths: in oscillation 1, there is only one path and no oscillation, in oscillation 2 the optimal path oscillates between this one and a new one, in oscillation 3 the optimal path alternates among the two paths and a new one etc. This phenomenon is seen less as the number of sources increases. For example, no oscillation of the optimal path is seen in the case of 4 sources. The reason is that optimal paths used for a given source change when blocks are used by objects entering through other sources.
- *Duration of usage of each optimal path during one oscillation*: In case of one source, in the oscillation 1, the first optimal path is used a number of times, in oscillation 2, the two optimal paths are used fewer times each, in oscillation 3 the three optimal paths are used even fewer times each, etc. Formally:

$$u(i) > u(i+1), \quad i > 0 \tag{6}$$

  where $u(i)$ is the number of times *each* optimal path is used during oscillation number i.
- *Duration of usage of each optimal path during the whole simulation*: The first optimal path is the most used, afterwards the second optimal path etc. This is because once an optimal path appears, it is used until the end of simulation. For example, in one source the first optimal path is used in all the oscillations, the second path is used for oscillations 2, 3, …, the third path is used for oscillations 3, 4, …, and so on.

3.4 Optimal Number and Position of Sources

The same three previous simulations are used.

Figure 9a shows the evolution of the total transfer time as a function of object number reaching the destination, which measures the conveying speed of the surface. It is found that the transfer time is greater with one source than with two or four sources. Intuitively, the reason is that source (0, 0) is at distance of 7 blocks from destination, whereas for two or four sources, source (3, 0) is sometimes used, and it is nearer to destination than (0, 0), at distance of 4 blocks.

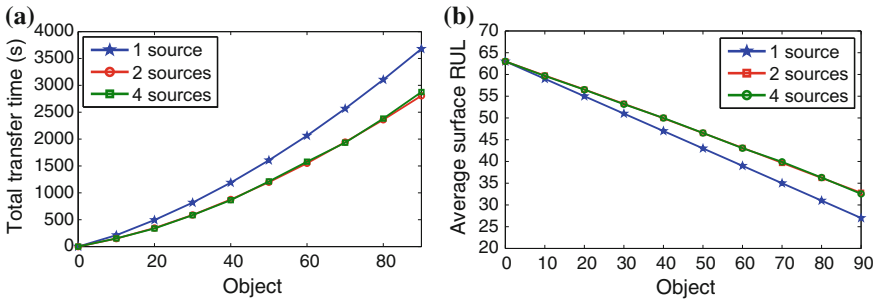


Fig. 9 **a** Total transfer time and **b** average surface RUL as a function of number of objects for different number of sources

Figure 9b shows the evolution in time of the average surface RUL (counting all the blocks) which measures the degradation, i.e. the health state, of the surface: the smaller the average surface RUL, the more degraded the surface. We note that the average surface RUL is smaller when objects enter only through (0, 0). The reason is the same as previously: the source (0, 0) is farther than other sources.

To confirm that the reason is not the number of sources, but their position, a forth simulation is done which uses only the source (3, 0). This simulation takes less time and the average RUL on the surface is greater than all the three simulations, which confirms that the important parameter is the position of the source. However, in this fourth simulation only 44 objects can be sent because the RUL of the source is 44 cycles. More importantly, as a general fact, the source (and the destination) are the most used blocks and their degradation increases greater than other blocks: the nearer to the source or destination, the greater the degradation of a block.

As a conclusion, in order to increase the lifetime (allow to send more objects) and the conveying speed of the surface, we propose to place sources at convenient positions and use them alternatively. In a future work, we will study source positions and object scheduling among these sources.

4 Conclusion and Future Works

In this paper, we have considered a surface for conveying fragile and tiny micro-objects based on air-jet technology. The surface is composed of an array of decentralized blocks, each one containing an electro-thermally actuated MEMS valve.

We took into account the optimal path, using criteria related to MEMS degradation, which avoids degraded blocks on the surface. We analyzed the evolution of the optimal path in dynamic conditions in order to maintain as long as possible a good performance of the conveying surface. We noticed that optimal paths alternate

or simply change during usage depending on the number of sources, some paths are used much more than others, and the greater the number of sources, the greater the number of optimal paths.

Future works include the scheduling of objects through the different sources, the extension of the algorithm to allow several concurrent objects on the surface, and the use of 4 MEMS per block, one block for each direction.

This work is only a step towards the realization of a contactless distributed MEMS-based conveyor. The ongoing results of the PHM part will be implemented and carried out on an experimental centimeter scale self-reconfigurable smart blocks conveyor which is being manufactured.

Acknowledgments This work has been supported by the Région Franche-Comté and the ACTION Labex project (contract ANR-11-LABX-0001-01).

References

1. Baraldi, P., Mangili, F., Zio, E.: A kalman filter-based ensemble approach with application to turbine creep prognostics. IEEE Trans. Reliab. **61**(4), 966–977 (2012)
2. Blough, D.M., Santi, P.: Investigating upper bounds on network lifetime extension for cell-based energy conservation techniques in stationary ad hoc networks. In: International Conference on Mobile Computing and Networking, pp. 183–192, Atlanta, GA, USA (Sep 2002)
3. Dahroug, B., Laurent, G.J., Guelpa, V., Fort-Piat, L., et al.: Design, modeling and control of a modular contactless wafer handling system. In: 2015 IEEE International Conference on Robotics and Automation (ICRA), pp. 976–981. IEEE (2015)
4. Floréen, P., Kaski, P., Kohonen, J., Orponen, P.: Lifetime maximization for multicasting in energy-constrained wireless networks. IEEE J. Sel. Areas Commun. **23**(1), 117–126 (2005)
5. Fukuta, Y., Chapuis, Y.A., Mita, Y., Fujita, H.: Design, fabrication, and control of mems-based actuator arrays for air-flow distributed micromanipulation. J. Microelectromech. Syst. **15**(4), 912–926 (2006)
6. Huang, H., Gao, S.: Optimal paths in dynamic networks with dependent random link travel times. Transp. Res. Part B **46**(5), 579–598 (2012)
7. Javed, K.: A robust and reliable data-driven prognostics approach based on extreme learning machine and fuzzy clustering. Ph.D. thesis, University of Franche-Comté, Besançon, France (2014)
8. Jay, L., Fangji, W., Wenyu, Z., Masoud, G., Linxia, L., David, S.: Prognostics and health management design for rotary machinery systems reviews, methodology and applications. Mech. Syst. Signal Process. **42**(1), 314–334 (2014)
9. Kang, I., Poovendran, R.: On lifetime extension and route stabilization of energy-efficient broadcast routing over MANET. In: International Network Conference, pp. 81–88, London, UK (Jul 2002)
10. Kirby, B.T., Ashley-Rollman, M., Goldstein, S.C.: Blinky blocks: a physical ensemble programming platform. In: CHI'11 Extended Abstracts on Human Factors in Computing Systems, pp. 1111–1116. ACM, Vancouver, Canada (2011)
11. Konishi, S., Fujita, H.: A conveyance system using air flow based on the concept of distributed micro motion systems. J. Microelectromech. Syst. **3**(2), 54–58 (1994)
12. Kurokawa, H., Tomita, K., Kamimura, A., Kokaji, S., Hasuo, T., Murata, S.: Distributed self-reconfiguration of M-TRAN III modular robotic system. Int. J. Robot. Res. **27**(3–4), 373–386 (2008)

13. Matmat, M., Koukos, K., Coccetti, F., Idda, T., Marty, A., Escriba, C., Fourniols, J.Y., Estève, D.: Life expectancy and characterization of capacitive RF MEMS switches. Microelectron. Reliab. **50**(9), 1692–1696 (2010)
14. Medjaher, K., Zerhouni, N.: Hybrid prognostic method applied to mechatronic systems. Int. J. Adv. Manuf. Technol. **69**(1–4), 823–834 (2013)
15. Medjaher, K., Skima, H., Zerhouni, N.: Condition assessment and fault prognostics of microelectromechanical systems. Microelectron. Reliab. **54**(1), 143–151 (2014)
16. Salemi, B., Moll, M., Shen, W.M.: SUPERBOT: a deployable, multi-functional, and modular self-reconfigurable robotic system. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 3636–3641, Beijing, China (Oct 2006)
17. Shea, H.R.: Reliability of MEMS for space applications. In: MOEMS-MEMS 2006 Micro and Nanofabrication, pp. 61110A–61110A. International Society for Optics and Photonics (2006)
18. Skima, H., Medjaher, K., Varnier, C., Dedu, E., Bourgeois, J.: Hybrid prognostic approach for Micro-Electro-Mechanical Systems. In: IEEE Aerospace Conference, pp. 1–8. 36, Big Sky, Montana, USA (Mar 2015)
19. Tanner, D.M.: MEMS reliability: where are we now? Microelectron. Reliab. **49**(9), 937–940 (2009)
20. Yin, S., Zhu, X.: Intelligent particle filter and its application on fault detection of nonlinear system. IEEE Trans. Industr. Electron. **62**(6), 3852–3861 (2015)

The Issue of Analyzing Measurement Data of Driving Speed in Large Urban Areas

Emilia Skupień and Agnieszka Tubis

Abstract The measurement of speed is the subject of analysis and publication of many researchers around the world. This issue is analyzed from the point of view of movement engineering, urban planning, logistics, but also the city management. The researches, initiated by the authors in this paper, are focused on measuring the time of vehicle journeys on the selected section of one of the main streets of Wroclaw. The aim of the chapter is to present the issues related to the analysis of measurement data from the ITS, which are aimed at supporting decision-making processes of those responsible for organizing transport in the city. For this purpose, the authors presented the basic theoretical issues of the study area and the approximate principles of operation of ITS systems. Then on the basis of quantitative analysis, the authors characterized the difficulties that may be encountered while drawing conclusions from the analysis of the collected data.

Keywords Movement engineering · Data analyzing · ITS

1 Introduction

The most important attribute of the road traffic is the speed of vehicles. It is a measure of the quality of movement and its consequences, both for its individual participants, as well as the general public dimension. The speed determines the comfort, convenience, traffic safety and the economics of transportation and the rank of traffic impact on the environment [14]. In large urban areas the speed of vehicles in addition to the usual groups of factors discussed in the literature section,

E. Skupień (✉) · A. Tubis
Wroclaw University of Technology, 27 Wybrzeze Wyspianskiego Street, Wroclaw, Poland
e-mail: emilia.skupien@pwr.edu.pl

A. Tubis
e-mail: agnieszka.tubis@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_36

is also affected by the action of the Intelligent Transportation System, which, among other determines the applicable cycles of traffic lights.

The measurement of speed is the subject of analysis and publication of many researchers around the world. This issue is analyzed from the point of view of movement engineering, urban planning, logistics, but also the city management. In fact it is also a point of interest to institutions responsible for traffic management in urban areas, who use data on average speed to increase road safety and improve passenger comfort. A properly conducted measurement, and above all accurate analysis of the data obtained, will affect the correctness of the decisions made by the authorities.

The researches, initiated by the authors in this paper, are focused on measuring the time of vehicle journeys on the selected section of one of the main streets of Wroclaw. The aim of the chapter is to present the issues related to the analysis of measurement data from the ITS, which are aimed at supporting decision-making processes of those responsible for organizing transport in the city. For this purpose, the authors presented the basic theoretical issues of the study area and the approximate principles of operation of ITS systems. Then on the basis of quantitative analysis, the authors characterized the difficulties that may be encountered while drawing conclusions from the analysis of the collected data.

The research presented in the chapter is preliminary and is only the first stage initiated research.

2 Literature Review

The issue of traffic measurement is of interest to the researchers examining the various transportation systems [9, 12]. Traffic studies are conducted and used for very different short-term and long-term purposes. The literature emphasizes that the primary objectives of measurements and studies of traffic is to gather information on its condition. This will include the setting of such characteristics as [2]: traffic flow in vehicles per hour or per day; fluctuations in traffic during the day, week or year; traffic structure; traffic burden; traffic flows at junctions and intersections (the directional structure); speed of movement; intervals between the vehicles; loss of time when traveling along the roadway or intersection; the size of forming traffic jams; types of traffic (source, target, through, internal); road trips (route); pedestrian traffic.

Based on these data it is possible to prepare [3]:

- study of transport for planning urban transportation systems, together with the traffic forecasts;
- plans and projects of roads and crossroads;
- traffic organization, capacity assessment and traffic conditions;

- economic analyzes;
- accident analysis;
- arrangements for traffic noise levels, air pollution and other environmental impacts.

Traffic studies provide data for analysis, discussion and decisions in planning, design and management of transport solutions and controlling the traffic [6]. Their aim is to support the decision-making process targeting the best possible use of transport infrastructure and to ensure the safe and efficient movement of people and cargo.

As noted in the introduction, the most important attribute in road traffic is the speed of vehicles [14]. Speed is an essential parameter in describing traffic conditions in the transport network. This parameter can be used to estimate the quality of travelling in the area. Moreover, with the development of Intelligent Transport Systems, vehicle speed and vehicle flow play an important role in advanced traffic management systems. The speed, along with other parameters, such as traffic flow, provide information about the current situation in the transport network [1]. Traffic speed data is one of the most important information sources for ITS, Advanced Traveler Information Systems (ATIS), and Advanced Traffic Management Systems (ATMS) [16].

Previous studies [3, 8, 15] indicate that the choice of speed by drivers in a free traffic is influenced by 6 sets of factors: (1) the characteristics of vehicles; (2) the characteristics of drivers; (3) the characteristics of the road and its environment; (4) weather conditions; (5) the legislation concerning speed limits and penalties for exceeding the speed limits; (6) the exact moment, the year, the season and time of the day.

In the case of researches carried out in city traffic, one should take into account additional factors that can affect the measured speed.

Among the different speed-related parameters known in the literature, average speed is very often used as measure for safe driving, mainly because elevated crash risk and severity have been related to an increase in average speed [4, 13]. It should also be noted that the formulas given in the literature to determine the average speed of vehicles, predominantly refer to vehicles in free traffic (e.g. Kempa's, Szczuraszek's and Kempa's equations) [6].

Some of the authors [5, 11] also emphasize that the use of aggregate statistics fails to recognize the probability distribution of the individual observed values. The apparent improvement in explaining the variation of the parameter is shown by the aggregation speed data, which may reduce the individual extreme values and overestimate the consistency and safety level. Therefore, individual speed profiles should be examined. Consequently, aggregate values also should be analyzed in order to find out if the difference between the individual and aggregate values is statistically significant.

3 The Use of Intelligent Transportation Systems in Traffic Management

Name of the Intelligent Transportation Systems means systems which represent a broad collection of diverse technologies (telecommunications, information technology, automatic and measuring), and management techniques for transport purposes in order to protect the lives of road users, improving the efficiency of the transportation and the protection of natural resources [17]. Traffic management using ITS technologies is divided into two subsystems: urban traffic management and traffic management on highways. Due to the scope of the study authors attention will be focused only on the first subsystem.

In urban traffic management most often used systems are [10]: (1) traffic management systems in the network of streets; (2) the automatic traffic surveillance; (3) the automatic charge collection.

These systems are used on a large scale in cities and urban areas. Also, the city, which is the subject of presented research is fitted with a system for ITS.

The main task of ITS is to improve traffic and safety of cars and public transport vehicles [18]. The main functional areas of the ITS system are [7]:

- the possibility of a dynamic and adaptive traffic management, responding to changing traffic conditions at any real-time;
- simultaneous traffic control at various levels: prioritizing public transport, liquefaction vehicular traffic, improving of openings of pedestrian crossings, crisis management;
- the ability to collect, process and distribute information to support the decision of the residents in the planning and implementation of the travel services and decisions on categorizing and prioritizing remedial action and maintenance of urban infrastructure.

These functions are realized by the system by continuously gathering information about the situation on the roads and intersections, by the infrastructure permanently embedded in the city. The data used by the ITS system to analyze the situation on the road, relate mainly to traffic at crossroads covered by its operation. Collected in the database values are obtained through detectors identifying and counting the vehicles, mounted next to the traffic lights at crossroads.

4 Specificity and Problems of Measurement Data Analysis

Customizing traffic lights by the Intelligent Transportation Systems is based on automatically collected and processed data. They therefore contain arising in such circumstances outliers and this can lead to undesirable settings. Knowing the way of collecting measurement data on the main roads of Polish cities, prior to their

analysis, one can distinguish the expected reasons for outliers. These reasons include:

- taking into account by the system, emergency vehicles and other with a speed higher than allowed, whose traveling time will be significantly lower than the average values of the group,
- vehicles stop on the route (e.g. as a result of a collision or crash) or deviating from the route and returning after some time, their time of travel will be longer.

Analysis of traveling time of cars crossing the sections of roads in urban areas, poses many difficulties. Facilitation for inferences about a set data is the ability to match them to a known probability distribution. For this reason, the authors have made an attempt to find a universal method of outlier data rejecting, which allows fitting of the remaining data to a known probability distribution.

The tests were conducted for two groups of measuring data, collected for journey times on the same stretch of road in the same direction during morning and afternoon rush hours. A summary of the raw data containing all the measurements from mornings and afternoons rush hours are shown in Table 1 and Fig. 1.

Because of the predicted outliers and inability to adjust the raw data into any of the known statistical distribution, authors have made attempts to reject outliers in such a way as to obtain the universal method of data filtering.

Most of the known methods of rejecting outliers is used for normal distributions, but in the examined case measurement data did not show this pattern. Therefore, attempts have been made to find methods tailored to the individual specificities of the phenomenon.

Table 1 A summary of the raw data (time of traveling through the investigated section)

	Morning	Afternoon
Sample size	687	471
Average	1023	1102
Standard deviation	857	1412
Max value	9549	9645
Min value	360	374
Median	848	550

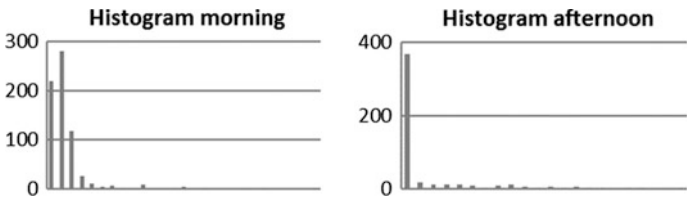


Fig. 1 Histograms transit times through the tested section at the morning and afternoon rush hours

The first attempt was to reject the maximum passes time until the median value was 0.95 times the average. Summary of the left data is shown in Table 2 and Fig. 2.

The histogram for the afternoon rush hours can be considered similar to the log-normal distribution, but the data obtained using the same method for the morning rush hours, do not allow adaptation to any known statistical distributions. Therefore, the result was not considered satisfactory.

A second attempt the rejection of outliers was based on the exclusion of times of travel at a speed of less than 0.33 of the maximum speed. The results are shown in Table 3 and Fig. 3.

The conclusions from the rejecting outliers are similar to previous ones. The results obtained in the afternoon rush hours differ significantly from the results of the morning. The histogram for the morning rush indicates the possibility of the coexistence of two groups of overlapping data. This phenomenon however has not been recorded for the afternoon rush hours.

Table 2 A summary of the data, for median 0.95 times the average

	Morning	Afternoon
Sample size	660	386
Average	363	141
Standard deviation	2684	1273
Max value	360	374
Min value	881	550
Median	837	524

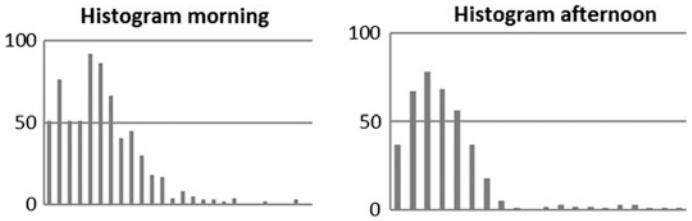


Fig. 2 Histograms transit times through the tested section at the morning and afternoon rush hours, for median 0.95 times the average

Table 3 A summary of the data, exclusion V < 0.33 Vmax

	Morning	Afternoon
Sample size	505	380
Average	727	540
Standard deviation	192	116
Max value	1079	1098
Min value	360	374
Median	848	550

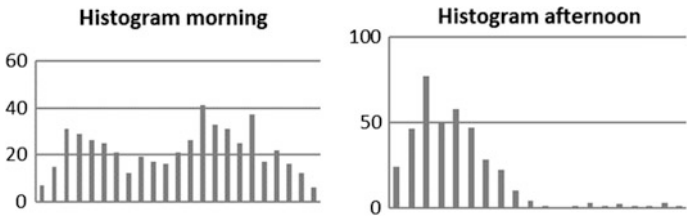


Fig. 3 Histograms transit times through the tested section at the morning and afternoon traffic rush hours, exclusion V < 0.33 Vmax

Due to the fact that the authors wanted the queried data rejection method to be universal, it cannot be regarded as satisfactory, so there have been no attempts of implementing for other data.

Another attempt differ from previous ones, this time rejected has been both the highest and lowest results. The authors rejected travel times for speeds less than 0.75 of average speed and above 1.5 average speed. The obtained data are shown in Table 4 and Fig. 4.

The proposed method of discarding the lower and upper values, has made the two histograms difficult to interpret. For this reason, also this method could not be considered satisfactory.

Another proposed criterion for rejecting data become the longest and shortest forecasted travel time. The measurements of vehicles traveling above the permitted speed (50 km/h) and below 20 km/h (adding a possible waiting time at intersections

Table 4 A summary of the data, exclusion 0.75 V max < V < 1.5 V max

	Morning	Afternoon
Sample size	381	345
Average	791	532
Standard deviation	139	76
Max value	1042	764
Min value	523	411
Median	804	525

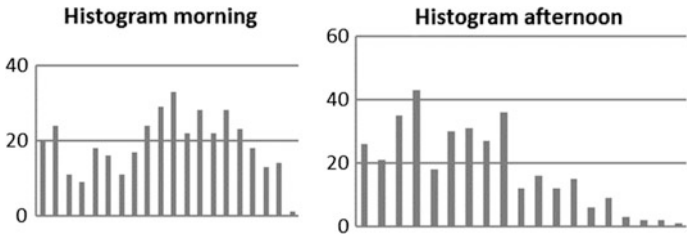


Fig. 4 Histograms transit times through the tested section at the morning and afternoon rush hours, exclusion 0.75 Vmax < V < 1.5 Vmax

Table 5 A summary of the data, exclusion 20 km/h < V < 50 km/h

	Morning	Afternoon
Sample size	597	360
Average	821	566
Standard deviation	250	150
Max value	1391	1323
Min value	415	416
Median	814	533

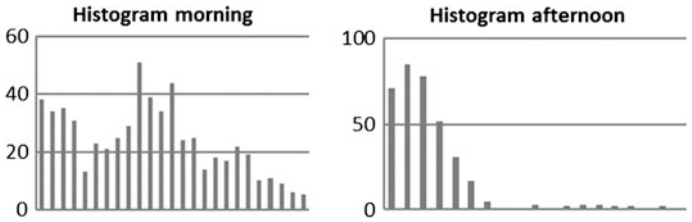


Fig. 5 Histograms transit times through the tested section at the morning and afternoon rush hours, exclusion 20 km/h < V < 50 km/h

with traffic lights by 30 s) has been rejected. Obtained in this way results are presented in Table 5 and Fig. 5.

These data confer similar to those occurring in the prior, difficulties of interpretation.

On the basis of this chapter the results of rejecting outliers tests, it can be stated that the analysis of data from measurements of travel times is a extremely complicated and demanding task and requires to take into account many factors. The very characteristics of the examined road sections undermines the credibility of some measurements. Registered traveling time, concerned road traffic on one of the main arteries of Wroclaw, connecting the western part of the city with the city center. On the investigated road is a provincial hospital, for this reason there are often recorded ambulance rides, whose speed exceeds the legal limit. This can affect an average speed of travel of all measured vehicles. However greater important can be attributable to the facilities, which are locate in the vicinity of the test route and may significantly understate measured travel time. These are—one of the largest shopping malls in Wroclaw and other shopping centers, business centers (collecting banks and other service companies), two large health clinics and the Municipal Social Centre. These objects are a place of temporary stop, often not exceeding 60 min. This causes the re-inclusion in the urban traffic of a vehicle in the studied time window, and the travel time is extended by the temporary stop.

The solution to this problem could be comparation each subsequent measurement time with the average travel time. This average travel time should be determined by the method of moving average with n values. The parameter n should be

dependent on the frequency of the occurrence of the next recorded measurements (a large number of measurements recorded in the adopted unit of time will increase the value of n; a small number of recorded measurements will reduce the number of n parameters). However the proposed method is characterized by a large computational complexity. High complexity of the method requires a properly prepared calculation algorithm, supported by appropriate IT tool. The possibility of its use will be the subject of further study of the authors.

5 Conclusions

Large urban agglomerations invest in ITS in order to improve traffic flow and transport safety. Analyses of data collected by these systems are intended to support decision-making processes of people responsible for the organization of transport in cities. However, as shown in the paper: the historical data analyzed with a simple statistical methods, can lead to erroneous conclusions. It is necessary to carry out detailed analyzes that will allow to clean up the original data series. Only this way it is possible to select the appropriate data for further quantitative analysis. Additionally when given only historical data from the ITS system, the analyst is unable to carry out a credible inference. For a complete analysis one needs additional information, e.g.: weathering, random events (e.g. car accidents), events planned (e.g. a narrowing streets due to road works). Excluding the knowledge of conditions which accompany measurements, created plans for perfecting the urban traffic management may be based on false assumptions.

In the beginning of this chapter it was pointed out that the presented results are introducing the studies initiated by the authors. The purpose defined at the outset there was a presentation of the model analysis of the speed of vehicles based on historical data. The intention of the authors was to present the complexity and difficulty of quantitative analyzes based on primary data obtained from the ITS. For this reason, as the summary it should be stated that: (1) it is impossible to give a general statistical distribution that matches the analyzed historical data; (2) preliminary analysis of data should take into account the process of cleaning the raw data of erroneous or incomplete measurements; (3) the correctness of the performed inference should take into account the conditions associated with the measurement; (4) the person responsible for the analysis should be familiar with the characteristics of the assessed routes and common behavior patterns of drivers on the examined sections.

Therefore, the next stage of the researches is to develop a model procedure for treatment of primary data, enabling to obtain a satisfactory level of reliability of the data analysing for supporting traffic management in the city.

References

1. Birr, K., Jamroz, K., Kustra, W.: Travel time of public transport vehicles estimation. Transportation Research Procedia **3**, 359–365 (2014)
2. Datka, S., Tracz, M.: Przewodnik do ćwiczeń z inżynierii ruchu. Politechnika Krakowska, Kraków (1974)
3. Datka, S., Suchorzewski, W., Tracz, M.: Inżynieria ruchu. WKŁ, Warszawa (1999)
4. European Commission: MASTER Managing speeds of traffic on European roads (No. RO-96-SC.202). European Commission, Finland (1999)
5. Freedman, D.A.: Ecological inference and the ecological fallacy. Int. Encycl. Soc. Behav. Sci. **6**, 4027–4030 (1999)
6. Gaca, S., Suchorzewski, W., Tracz, M.: Inżynieria ruchu drogowego: teoria i praktyka. WKŁ, Warszawa (2014)
7. Hadas, A.: Zastosowanie zintegrowanego systemu transportu w aglomeracji miejskiej. Przegląd Komunikacyjny **4**, 8–13 (2015)
8. Kempa, J., Szczuraszek, T.: Design speed, actual speed and road designing. In: 2nd International Symposium on Highway Geometric Design. Transportation Research Board, Road and Transportation Research Association, Mainz, German (2000)
9. Kierzkowski, A., Kisiel, T.: The simulation model of logistic support for functioning ground handling agent, taking into account the probabilistic time of aircrafts arrival. In: Carpathian Logistics Congress—Congress Proceedings, pp: 463–469 (2013)
10. Oskarbski, J., Jamroz, K., Litwin, M.: Inteligentne systemy transportu– zaawansowane systemy zarządzania ruchem. www.pkd.org.pl/pliki/referaty/oskarbski,_jamroz,_litwin.pdf. Accessed January 2016
11. Park, Y., Saccomanno, F.: Evaluating speed consistency between successive elements of a two-lane rural highway. Transp. Res. Part A: Policy Pract. **40**, 375–385 (2006)
12. Restel, F.J.: Train punctuality model for a selected part of railway transportation system. In: Safety, Reliability and Risk Analysis: Beyond the Horizon—Proceedings of the European Safety and Reliability Conference (2014)
13. Shinar, D.: Traffic Safety and Human Behavior. Elsevier, Oxford (2007)
14. Szczuraszek, T.: Prędkość pojazdów w warunkach drogowego ruchu swobodnego. Komitet Inżynierii Lądowej i Wodnej PAN, Warszawa (2008)
15. Watanatada, T.: A Model for Predicting Free-flow Speed Based on Probabilistic Limiting Velocity Concepts: Theory and Estimation. Transportation Research Record, vol. 1091 (1986)
16. Zhang, J., Wang, F.Y., Wang, K., Lin, W.H., Xu, X., Chen, C.: Data-driven intelligent transportation systems: a survey. IEEE Trans. Intel. Transp. Syst. **12**(4), 1624–1639 (2011)
17. www.itspolska.pl. Accessed January 2016
18. http://zdium.wroc.pl/view/document/185. Accessed November 2015

Implication of Availability of an Electrical System of a Wind Farm for the Farm's Output Power Estimation

Robert Adam Sobolewski

Abstract Availability of an electrical system of a wind farm plays a crucial role among factors affecting the power output of a farm. The availability is determined by an internal collection grid topology and reliability of its components, e.g. generators, inverters, transformers, cables, switch breakers, protective relays, and busbars, to name a few. A wind farm availability's quantitative measure can be: (i) the probability distribution of combinations of "in operation" states of the farm's wind turbines and (ii) the expected power to be delivered to the power system. The "in operation" state of a wind turbine involves the availability of the wind turbine and other equipment necessary for the power transfer to the external grid. They can be used for the analysis of the impact of various topologies and the reliability of individual components on the availability. The second kind of analysis may be supported by the importance ranking of the components. The paper presents the approach to formulating the reliability models that is based on Bayesian networks, useful importance measures of the components and the case study that illustrates the approach application.

Keywords Wind energy · Reliability · Bayesian networks

1 Introduction

A wind farm's output power depends on, among others, the farm's availability. The availability is determined by the reliability of components included in a wind farm (WF), e.g. electrical components: wind turbine generators, transformers, inverters, cables, circuit breakers, protective relays and busbars, to name a few. Some of these components constitute an internal collection grid, i.e. an electrical system that integrates wind turbines (WTs) and the collector hub (CH). The grid can be

R.A. Sobolewski (✉)
Faculty of Electrical Engineering, Bialystok University of Technology,
Wiejska 45D, 15-351 Bialystok, Poland
e-mail: r.sobolewski@pb.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_37

designed in several ways, depending on the WF's location (on-shore or off-shore), size (the number of WTs and total power installed) and the desired level of redundancy (reliability). The most common designs are radial and ring ones.

Failures of the WF's components can cause their outages, and sometimes, be spread to adjacent components. Failures can spread in the following situations: (i) a circuit breaker's or a protective relay's failure (missed tripping when required), or (ii) circuit breakers' and protective relays' location in the internal grid, which doesn't allow for the outage of the damaged component only. Failure spreading may result in the need to outage either the feeder with the damaged component or the WF. The consequence of these outages can be losing the power (partially or totally) to be either generated by WT or transferred from WTs to a point of common connection (PCC).

Implication of availability of an electrical system for WF's output power can be evaluated quantitatively relying on: (i) probability distribution of the occurrence of a combination of WT operating states, (ii) expected value of WF's output power and (iii) importance rankings of the electrical components. Operating state of a WT indicates its ability to generate power and output it to the PCC, which means the availability of the WT and of other components necessary to transfer power to the PCC. Importance rankings can be used to measure the effect of the reliability of individual components on the WF's output power (to identify the weaknesses).

The relevant literature offers some approaches, which were used for WF availability representation in models that describe the characteristics of WT and WF power and energy output [1–6]. The models feature a simplified representation of the impact of multiple factors on WF capacity. The main reason is taking into account two consequences of reliability states of the WF components only ("up" state—in operation, "down" state—outage). In fact, another consequence of reliability states of some components should be considered additionally, i.e. "down" state—failure spread.

Since the output power of WF can vary due to its availability, it is essential to estimate profits of individual designs of electrical system relying on accurate probabilistic models. The paper presents such models that, to a greater extent than before, take into account the important factors (e.g. spreading of the component failures) that determine the availability of WF's electrical system and allow one to obtain the probability distribution of the occurrence of combinations of both "in operation" and "failed" states of the WTs. Bayesian networks (BN) were used to formulate the models. Moreover, to evaluate how the particular component contributes to a WF availability, and how the presence of a component affects the contributions of other components in the WF, some importance measures are introduced in the paper as well. These are the following: Risk reduction worth, Risk achievement worth, Fussell-Vesely, and Birnbaum. A case study presents two BN models and the results of the availability analysis of an electrical system of demonstrated WF of both radial and ring internal collection grid topologies.

2 Electrical System of a Wind Farm

A WF consists of many electrical components, needed for: power generation, power transfer to the external grid and protection of the components and external grid against abnormal conditions (e.g. failure, short circuit, overvoltage, and so on). Some of them form the internal collection grid. There are mainly two different conceptual designs of the grid that can be utilized: radial and ring design.

Figure 1 shows the radial topology (RAT) of an internal collection grid. WTs are connected in series to one feeder and collected at the CH. The main advantage of this design is that the total array cable (CA) length is smaller than for the other options. Also, it is simple to control, and it provides the possibility to taper the CA capacity as the distance from the hub increases. The major disadvantage with this design is the poor reliability provided. CA or circuit breaker of the feeder (CBF) faults at the hub side of the feeder will lead to the loss of power from all downstream turbines in the feeder.

Figure 2 shows the ring topology (RIT) of internal collection grid. Two feeders are connected in parallel to provide redundancy. This means that for two feeders, the CA length will only increase by the distance between the turbines at the end of the feeders. Nevertheless, in the case that a fault occurs at the cable between the first turbine of a feeder and the CH, the full output power of the WT in the faulted feeder must be diverted through the other one, meaning that the cable at the hub end of the latter needs to be dimensioned for the power output of the double number of WTs.

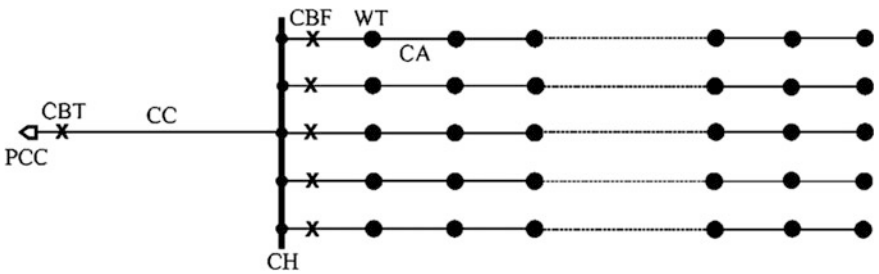


Fig. 1 Radial topology of an internal collection grid of the WF. All the designations in the text

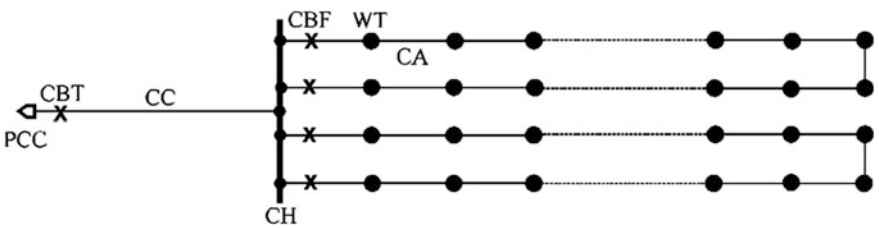


Fig. 2 Ring topology of an internal collection grid of a WF. All the designations in the text

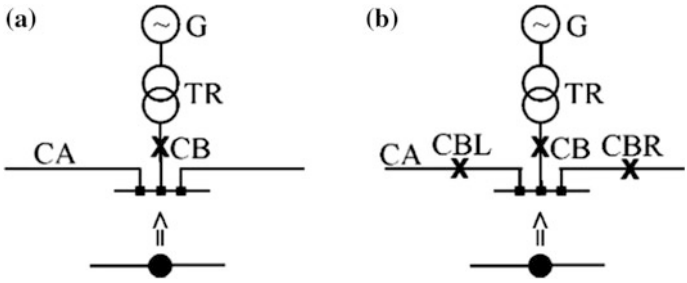


Fig. 3 Schematic of the WT connection to the feeder of WF internal collection grid topology: **a** radial and **b** ring one (single and double sided). All the designations in the text

The connection of the WT to the internal collection grid of WF is depicted in Fig. 3a for RAT (Fig. 1) and in Fig. 3b for RIT (Fig. 2). The main difference between these connections is the number of circuit breakers in CA sections, i.e. in the grid of RAT there are not any, and in the grid of RIT—there are two (CBL and CBR) at both opposite sides of WT connection point. A generator (G) fault or step up transformer (TR) fault, the corresponding circuit breakers (CB) should switch off the WT. The remaining WTs may stay connected to the feeder and still generate power. If CA section fails, the feeder should be switched off by circuit breaker of the feeder (CBF, Fig. 1) as there are not any circuit breakers with the CA sections in the grid of RAT. The same fault should be separated by two circuit breakers with the CA section in the grid of RIT because these circuit breakers are installed at both sides of the CA section. All the WTs will still keep in power generation, supplying in both directions through the operative CA sections.

The failures of some components can be spread to the adjacent ones. Failure spreading may result in the need to outage either the feeder with the damaged component or the entire WF. For example, the fault of G requires the operation of CB (Fig. 3). If the CB or protective relay is failed (missed tripping when required) then the fault should be separated by CBF (in RAT) or circuit breakers at of the CA sections closest to WT in question (in RIT). Thus the feeder will be out of operation (in RAT) or redundancy will be lost (in RIT). The fault of CA section in the feeder requires operation of CBF (in RAT) or operation of the circuit breakers of the adjacent CA sections (in RIT). If the circuit breakers in question or protective relays associated with them are failed (missing tripping when required) that the fault should be separated by the circuit breaker of the connection cable (CC) (CBT at Figs. 1 and 2).

3 Implementation of Bayesian Networks and Importance Measure in the Availability Analysis of an Electrical System

3.1 Bayesian Networks

The methodological basis for probabilistic modelling using Bayesian networks (BN) can be found in literature, such as [7, 8]. Formulation of a BN model of availability of an electrical system of WF entails the following elements: (i) identification of nodes, (ii) development of network structure, (iii) formulation of conditional probability tables (CPTs) for nodes, and (iv) setting of probabilities of the occurrence of individual reliability states of the components. The nodes identification includes: (i) root nodes, (ii) intermediate nodes representing the combination of the states of parent nodes, and (iii) a leaf node representing the electrical system availability.

To the root node, CPT is assigned that includes two reliability states "up" and "down" of a component (one or a set of the components) and the probabilities of the states occurrence. For example, the CPT of root node CB (Fig. 3) is shown it Table 1.

To the intermediate node, CPT is assigned that includes: (i) states of parent nodes, (ii) the operating states of the intermediate node itself and (iii) the conditional probabilities of the occurrence of operating states of the node. The number of states of the parent nodes can be two or three, depending on the components and their location in the internal collection grid. It is appropriate to distinguish between parent nodes with: (i) two states—"up" and "down" (root nodes) and "in operation" or "failed" (intermediate nodes), and (ii) three states—"in operation", "failed" and "failed with failure spreading", (intermediate nodes). The number of states for the intermediate node in question can be either two or three. The CPTs of the intermediate nodes and leaf node are to be deterministic ones (every probability is either 1 or 0). For example, the CPT of node A, that represents the states of CB and G (Fig. 3) is shown in Table 2.

Table 1 The CPT of the root node CB

Node CB state	P(CB)
cb = up	P(*cb* = up)
cb = down	P(*cb* = down)

Table 2 The CPT of the intermediate node A that represents the combinations of reliability states of components CB and G (Fig. 3)

	Node G state	*g* = up		*g* = down	
	Node CB state	*cb* = up	*cb* = down	*cb* = up	*cb* = down
Node A state	*a* = in operation	1	0	0	0
	a = failed	0	1	1	0
	a = failed with failure spreading	0	0	0	1

The CPT assigned to the leaf node contains all combinations of both operating and failed states of the turbines. WT operating state "1" means an ability of the WT and the components in at least one feeder, required to transfer the power to the external grid. Failed state "0" means the condition opposite to state "1" (Table 3).

Finally, the probability of the rth combination of WTs that are simultaneously in operation π_r can be calculated based on BN. The number of these probabilities equals 2^n, where n is the number of WTGs in a WF.

Relying on the probabilities π_r and expected values of the output power of WTs $\mathrm{E}(P_{WTa})$ one can evaluate the expected output power of the WF, i.e.

$$\mathrm{E}(P_{WF}) = \sum_{r=1}^{2^n-1} \left[\sum_{a \in J(r)} \mathrm{E}(P_{WTa}) \right] \cdot \pi_r \tag{1}$$

where $a \in J(r)$ when WT operating state is "1" in rth combination of WTs states.

3.2 Importance Measures of Components' Reliability

The purpose of important measures is to assess how much a single electrical component of WF contributes to the unavailability risk of a system. The risk involves the amount of power to be lost because of components' failures. This contribution may be analysed relying on structural importance measures, e.g. Risk reduction worth, Fussell-Vesely importance, Risk achievement worth and Birnbaum importance.

In Table 4 the following definitions are used: R(base)—the present risk level that refers to amount of WF's output power taking into account a power lost because of a'priori components failure, $\mathrm{R}(x_i = 1)$—the risk level with the component i assumed to be perfectly reliable, $\mathrm{R}(x_i = 0)$—the risk level with the component i assumed to be failed. The RRW(i) represents the maximum decrease in risk for an improvement to the component i unreliability. The FVI(i) represents the direct effect of the component i unreliability on the risk level. The RAW(i) presents a measure of "worth" of the component in "achieving" the present level of risk and indicates the importance of maintaining the current level of the component i reliability. The BI(i) is completely dependent on the structure of the system and the reliability of other components, and is independent of the current reliability of the component i. If BI(i) is large, a small change in the reliability of component i will result in a comparatively large change in the system risk.

Table 3 The CPT of leaf node that represents the combinations of operating states of 2 WTs

Node WT1 state	*wt*1 = o			*wt*1 = f			*wt*1 = fs		
Node WT2 state	*wt*2 = o	*wt*2 = f	*wt*2 = fs	*wt*2 = o	*wt*2 = f	*wt*2 = fs	*wt*2 = o	*wt*2 = f	*wt*2 = fs
WT1 ∩ WT2	1	0	0	0	0	0	0	0	0
WT1	0	1	0	0	0	0	0	0	0
WT2	0	0	0	1	0	0	0	0	0
NONE	0	0	1	0	1	1	1	1	1

Designations of the operating states: “o”—in operation, “f”—failed, “fs”—failed and failure spreading

Table 4 Risk importance measures [9]

Measure	Abbreviation	Formula
Risk reduction worth	RRW(i)	$\frac{R(base)}{R_{(x_i=1)}}$
Fussell-Vesely importance	FVI(i)	$\frac{R(base)-R_{(x_i=1)}}{R(base)}$
Risk achievement worth	RAW(i)	$\frac{R_{(x_i=0)}}{R(base)}$
Birnbaum importance	BI(i)	$R(x_i = 0) - R(x_i = 1)$

4 Case Study

4.1 Internal Collection Grid Topology

The example calculation refers to the availability analysis of WF's electrical system with two internal collection grid topologies—radial and ring. It was assumed that: the WF consists of 4 WTs with rated power $P_{Rj} = 2$ MW (j = 1, 2, 3, 4) each, the expected values of WT output powers are obtained based on real data, and the reliability of electrical system components' data are gathered from related literature.

Figure 4 depicts two diagrams of a WF internal collection grid topology. The designations in Figs. 4 and 5 are as follows: WT1, …, WT4—the wind turbines (every WT substitutes generator G and step up transformer TR, Fig. 3), CA1, …, CA5—array cable sections; CBF1, CBF12, …, CBF5, CBF51—circuit breakers in feeders (include protective relays); CB1, …, CB4—WT circuit breakers (include protective relays), and CH–PCC—the component representing the connection between CH and PCC.

The expected WTs output power obtained relying on real data are the following: E[P_{WT1}] = 519.3 kW, E[P_{WT2}] = 581.4 kW, E[P_{WT3}] = 527.4 kW and E[P_{WT4}] = 505.3 kW [10]. The probability of "down" state of electrical system

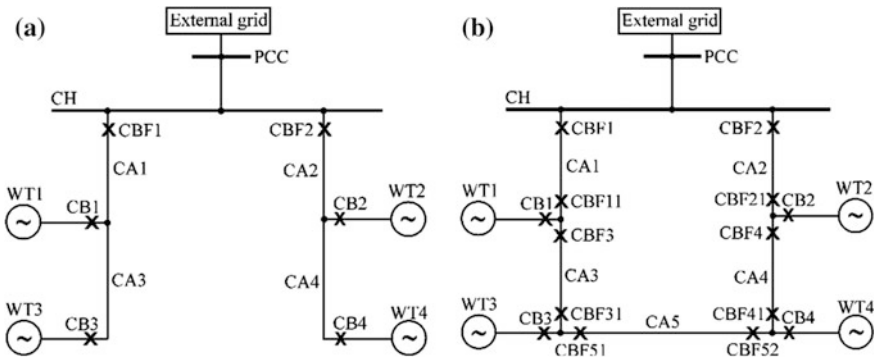


Fig. 4 Diagram of a WF internal collection grid with topology: **a** radial and **b** ring

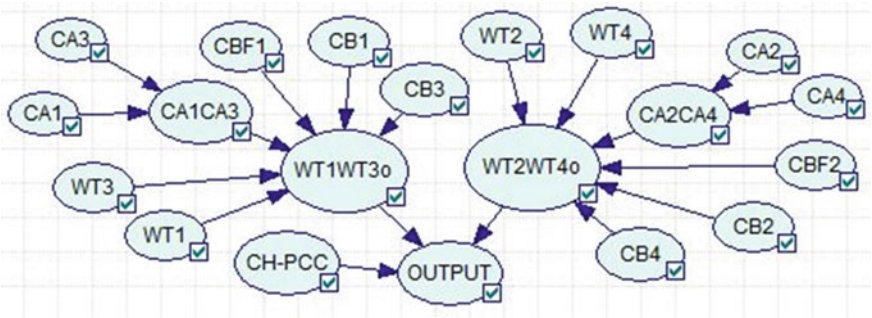


Fig. 5 BN of availability of an electrical system with radial internal collection grid topology of WF. The node designations in the text

components refers to FOR (Forced Outage Rate) parameter, commonly used in reliability analysis of systems. Parameters FOR assumed for the components are the following: 0.920—for wind turbines, 0.950—for array cable sections, 0.999—for circuit breakers in feeders, 0.995—for WTs circuit breakers, 0.940—for the component CH-PCC. The values of parameter FOR refers to the electrical system components of on-shore WF.

4.2 Bayesian Network Models

Figure 5 shows the BN model of availability of an electrical system with radial internal collection grid topology of WF. The root node designations refer to the WF components' designations (Fig. 4a). Two intermediate nodes represent combinations of operating states of the components. OUTPUT node is the node that takes into account the combination of the states of nodes WT1WT3o, WT2WT4o and CH-PCC.

Figure 6 shows the BN model of availability of an electrical system with ring internal collection grid of WF. The root node designations refer to the WF equipment designations (Fig. 4b). 28 intermediate nodes mapping combinations of operating states of the components represented by root nodes and other intermediate nodes. OUTPUT node is the node that takes into account the combination of states of nodes WT1o, WT2o, WT3o, WT4o and CH-PCC.

4.3 Results of Modeling

In the case of a'priori reliability of WF electrical components the probability of all WTs "in operation" are higher within RIT (0.6442) as compared to RAT (0.5365)

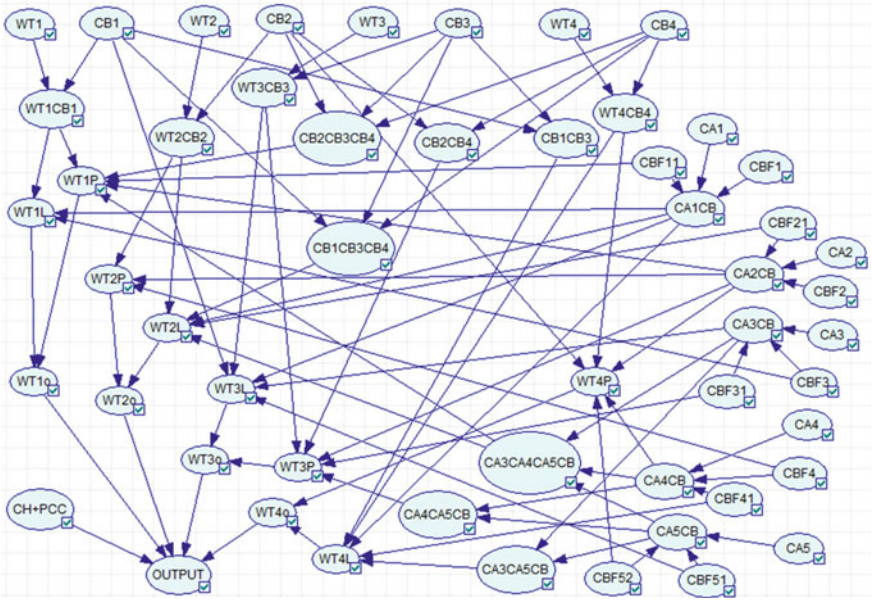


Fig. 6 BN of availability of an electrical system with double sided ring topology of WF. The node designations in the text

by 11 %. Moreover, the probabilities of all combinations of three WTs "in operation" are higher within RIT in comparison to RAT. The probability of NONE combination is a little bit lower within RIT (0.0672) comparing to RAT (0.0737). Assuming WT1 failed one can expect the probability of the combination of three WTs (of numbers 2, 3 and 4) is higher by 12.3 % within RIT (0.7063) as compared to RAT (0.5832). If L1 failed, one can expect nonzero probabilities of combinations of 4 WTs and 3 WTs within RIT. All these probabilities equal 0 for RAT. The probability of NONE combinations is lower by 5 % within RIT as compared to RAT. Assuming CB1 failed, the nonzero probabilities can be expected for one combination of 3 WTs (of numbers 3 and 4) and 3 combinations of 2 WTs (of numbers: 2 and 3, 2 and 4, 3 and 4) within RIT. On the contrary, the nonzero probability can be expected for one combination of 2 WTs (of numbers 2 and 4) within RAT.

Using the expected WT output powers provided in Sect. 4.1 and the calculated probabilities of a combination of WTs that are simultaneously in operation in both topologies, expected WF output power can be determined. Table 5 shows the expected output powers of WF with radial and ring internal collection grid topologies.

It follows from Table 5 that the WF components' variants of reliability states (a priori, WT1 failed, L1 failed, CB1 failed) impact to varying degrees the expected WF output power. This impact also depends on the adopted internal collection grid topology. In every variant of the component states, the expected power is higher for

Table 5 Expected output power of WF for radial and ring internal collection grid topology and four options of the components' state

Expected output power of the WF (kW)							
Radial topology				Ring topology			
a'priori	WT1 fail.	L1 fail.	CB1 fail.	a'priori	WT1 fail.	L1 fail.	CB1 fail.
1644.42	1244.74	837.95	837.95	1812.43	1370.41	1629.42	1260.76

Table 6 Importance measures of WF components: WT1, L1 and CB1, for RAT and RIT

Importance measure	Value of the component importance					
	Radial topology			Ring topology		
	WT1	L1	CB1	WT1	L1	CB1
RRW(i)	1.08	1.10	1.01	2.38	1.57	2.72
FVI(i)	0.07	0.09	0.01	1.66	1.03	1.01
RAW(i)	1.82	2.65	2.65	0.40	0.03	0.01
BI(i)	435.12	849.82	811.19	569.18	192.52	554.54

RIT rather than RAT. Occurrence of WT1 failure can decrease the output power by 24 % for RAT and RIT. But L1 or CB1 failure can reduce the output power by almost 50 % for RAT. On the contrary, failures of components L1 or CB1 can decrease the output power by respectively 10 and 30 %.

The investigation of the probabilities of the occurrence of the operating states of WTs and the expected output power of WF can be carried out for reliability states of every component of an electrical system. The results presented so far confirm that both closing the internal collection grid of WF by the line (CA5, Fig. 4b) and updating the grid with circuit breakers at both sides of every array cable section improve significantly the availability of a WF electrical system.

Table 6 shows the important measures of three components WT1, L1 and CB1 of an electrical system of WF calculated for RAT and RIT.

RRW(i) indicates that maximum decrease in risk (loss of the WF output power) can be expected from improving the reliability of L1 for RAT (CB1 for RIT). As regards FVI(i), the reliability of L1 for RAT (WT1 for RIT) has the highest effect on the risk level. RAW(i) indicates that maintaining the current level of reliability of L1 and CB1 for RAT (WT1 for RIT) is most crucial for receiving the present level of risk. As regards BI(i), the relatively small change in reliability of L1 for RAT (WT1 for RIT) will result in a comparative large change in the risk.

The investigation of importance measures of components of electrical system of WF can indicate components' impact on electrical system availability and support decision making process when the availability is not satisfying.

5 Conclusion

This paper presents probabilistic models using BN availability analysis of a WF electrical system with different internal collection grid topologies. The main advantage of the models is taking into account the most important technical factors that determine WT ability to generate power and its output to the external power grid. An important feature of the developed models is also an easy inference on the probabilities of a combination of WTs that are simultaneously in operation for assumed operating states of WF components. Obtained probabilities can also be successfully used to calculate the expected WF output power in view of its reliability and the measures of the components' importance.

References

1. Sannino, A., Breder, H., Nielsen, E.K.: Reliability of collection grids for large offshore wind parks. In: 9th International Conference on Probabilistic Methods Applied to Power Systems KTH, Stokholm, Sweden, 11–15 June 2006
2. Quinonez-Varela, G.: Electrical collector system options for large offshore wind farms. IET Renew. Power Gen. **1**(2), 107–114 (2007)
3. Bahirat, H.J., Mork, B.A., Hoidalen, H.K.: Comparison of wind farm topologies for offshore applications. In: IEEE Power and Energy Society General Meeting (2012)
4. Segura-Heras, I., Escriva-Escriva, G., Alcazar-Ortega, M.: Wind farm electrical power production model for load flow analysis. Renew. Energy **36**, 1008–1013 (2011)
5. Mabel, M.C., Raj, R.E., Fernandez, E.: Analysis on reliability aspects of wind power. Renew. Sustain. Energy Rev. **15**, 1210–1216 (2011)
6. Ali, M., Matevosyan, J., Milanovic, J.V.: Probabilistic assessment of wind farm annual energy production. Electr. Power Syst. Res. **89**, 70–79 (2012)
7. Darwiche, A.: Modelling and Reasoning with Bayesian Networks. Cambridge University Press (2009)
8. Kjaerulf, U.B., Madsen, A.L.: Bayesian Networks and Influence Diagrams. A Guide to Construction and Analysis. Springer
9. van der Borst, M., Schoonakker, H.: An overview of PSA importance measures. Reliab. Eng. Syst. Safety **72**, 241–245 (2001)
10. Sobolewski, R.A.: Wind farm reliability modelling using Bayesian networks and semi-Markov processes. ActaEnergetica **3**, 24 (September 2015)

CPU Utilization Analysis of Selected Genetic Algorithms in Multi-core Systems for a Certain Class of Problems

Jakub Sobuś and Marek Woda

Abstract This work was carried out in order to examine and compare selected models of genetic algorithms (through the implementation), using the latest tools and libraries that allow for multithreaded programming in a .NET environment. Implemented algorithms were then tested for the use of available resources, such as CPU cycles/cores consumption and the time at which they are able to provide the quality results at acceptable pace. With a choice of multi-core processors—allowing for parallel calculations on their cores, as well as genetic algorithms, one should think about how to implement the chosen algorithm so as to avoid the deadlocks and bottlenecks to make optimal use of the computing power of cores. There are many approaches to deal with such issues—a lot of tools and software libraries facilitate the implementation of such algorithms. This paper tries to address two essential questions what algorithms fit the best into multicore architecture, and which one benefits the best from available logical/physical cores producing the best possible results.

Keywords Implementation · Multi-threaded computation · Genetic algorithms · Resources consumption

1 Introduction

Access to multi-core systems has become lately so easy that most electronic devices—computers, mobile phones and even televisions are equipped with processors with more than one core. As the result, there is a possibility to use multi-threaded algo-

J. Sobuś (✉)
NeuroSYS Sp. z o. o., Rydygiera 2a/27, 50-249 Wroclaw, Poland
e-mail: j.sobus@neurosys.pl

M. Woda
Department of Computer Engineering, Wroclaw University of Technology, Janiszewskiego 11-17, 50-372 Wroclaw, Poland
e-mail: marek.woda@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_38

rithms written by a wide group of people who would gladly have used them to optimize problems that they have to cope with in a daily work. Genetic Algorithms (Gas) support scientists and engineers to cope with different design issues (e.g. create electronic circuits) for years [2, 7]. Genetic algorithms proved to be a justified mean in solving many problems in a fast and effective manner. There are still many issues (e.g. traveling salesman problem, that almost every logistics company deals with) that hampers productivity, simply because their solution space is too large to find in a seamless and fast way the optimal results. Having access to raw computer power, and introducing parallelism in genetic algorithms one can make them even more handy and widely used on a daily basis. Genetic algorithms from the beginning have aroused an interest in a wide range of researchers due to variety of implementation ways. There are many publications [1, 4, 6, 9] describing the distribution of genetic algorithms (GAs) in different subclasses. Muhlenbein [9] introduces a division into coarse-grained model, relying on partitioning of a single population on multiple processors. The other [4] assumes that coarse-grained model have many populations. The exchange of information occurs in this model thanks to an additional migration operator. It envisages the exchange of selected individuals between populations. In this way, a new model called the Island model, has been developed [4]. Other ongoing research, do not rely only on comparing different implementations, but also try to find the best genetic operators, comparing them in various respects. *Selection* operators can be compared in terms of reducing the diversity among the individuals selected as the next generation. The worst turns to be a selection the *truncation* scheme [3, 4]. This is due to the lack of a chance to survive of less adapted individuals. It turns out that despite the weak value of fitness function, they may carry in themselves chromosome fragments, allowing to reach the optimal solution of the problem. Therefore, it is wise to pass some of less adapted individuals to the next generation (*crossover*). Standard features of genetic algorithms may seem sufficient, but there are studies [5, 12] to check whether there are additional operations, enabling acceleration in getting results. Elitism is such an additional operator which gives decent results. The majority of individuals' selection patterns, assumed operation of the random factor in selecting a group of parents. The result is that there can be eliminated in a subject comprising the optimal solution. Elitism is assumed that before the start of the selection, you must make sure that the best individuals in the population shall be included into a next generation. It helps greatly to reduce the time needed to find an optimal solution [13]. Among the various studies [7, 11], one can also find these focused on the parallelization of algorithms. The fact is that a smaller number of synchronization points, leads to a higher efficiency during computation. That is why researchers aim at finding the best possible asynchronous implementations. However, it is worth to mention that the lack of synchronization between individuals or populations may lead to slow solutions propagation among individuals. The methods allowing for the exchange of information in a particular asynchronous implementation must cause so-called empty runs. This means that some processes run in parallel can execute tasks that have already been completed. Therefore, one should take this into account and check if for a given problem is better to make of synchronization or allow for unnecessary CPU cycles to be consumed. Everything depends of course on the tasks which genetic

algorithm is about to optimize [1]. Along with the release of software libraries enabling parallel programming using GPUs, first genetic algorithms implementations started to occur [2]. Graphics cards consist of many more cores than it does on the processor (CPU). GPU programming enables faster and more efficient way to performs tests on the parallel algorithms, thanks to the ability to implement complex models requiring many interacting threads. Yet again, it is worth to mention that the architecture of graphics processors is different from CPUs. Depending on GPU vendor an algorithm requires a different implementation. The implementation workload is far greater because there are not too many implementations that one can base the research on. Nonetheless the GPU programming is worth to spend time on, since the sheer number of parallel tasks is far greater than on any modern CPU and parallel tasks are executed tens or even hundred times faster [2]. The cited studies are only a small segment of existing research related to genetic algorithms. As it was mentioned earlier received results heavily depends on a nature of considered problem and implementation of the algorithm. The scientists continually test new solutions to form a more optimal selection operators and models of distribution of individuals in a population space. The idea of genetic algorithms will always have their group of supporters, since similar rules of operation occurs in the nature itself. The availability of new types of GPUs will constantly be making a new scope for development in this branch of science.

2 Implemented Algorithms

This section contains descriptions of models of genetic algorithms implemented for the purposes of this study. Selected models differ in the number of populations, threads synchronization and a way of implementation of some genetic operators. Criterion for selection of algorithms was their diversity Table 1.

Table 1 General comparison of implemented algorithms

	Model of genetic algorithm			
	Classical	Synchronous	Asynchronous	Island
# of threads	1	Many	Many	Many
# of populations	1	1	1	Many
Method of operation	–	Synchronous	Asynchronous	Mixed
# of sync points	–	Many	N/A	Few
Complexity	Easy	Easy	Moderate	Moderate

2.1 Classical Model

The first implemented was the classical model. It works as a single-threaded algorithm. This model consists of a single population. The method of selection chosen for its implementation is a tournament selection [6]. The tournaments are realized as a binary that means a tournament group consists of two random individuals from the population. Then the value of the fitness function is calculated. An individual with a higher value is copied to the population is transient, and the other individual returns to the initial pool population. Tournaments are held, until the expected amount of individuals in the target generation is reached. Less fit individuals, which failed to pass the tournament, are removed. The next step is the crossover. In our case it was one-point crossover. Crossover is about to draw two individuals from the transient population, chosen through the tournament from the previous step. After the selection of these individuals, the point of intersection is randomized that will influence creation of two individual children. The code snippet showing the loop method responsible for the rise of a new generation shown in the following listing. This is the simplest of the models implemented. It does not benefit out of multi CPU cores. This implementation will be a basis for comparison with other GAs (Fig. 1).

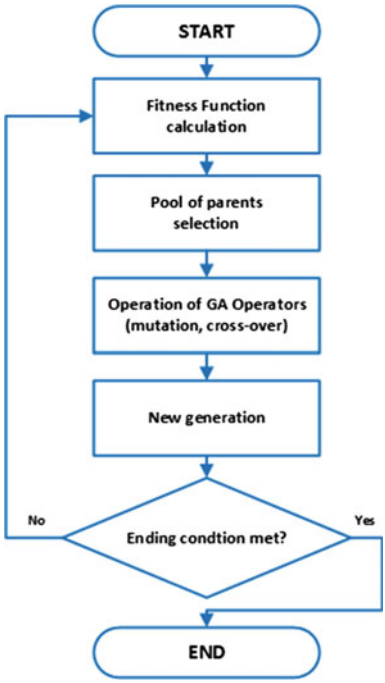


Fig. 1 Classical model—a block diagram

2.2 Synchronous Approach

Another GA model implemented for the purpose of this work is a synchronous one. The principle of its operation is based on a model known in programming called Master-Slave. The actual structure of the algorithm is very similar to the classical model. This model comprises of only one population. The steps of the algorithm are the same, but divided into several separate threads (Fig. 2). This model utilizes all available cores in the system, by creating Slave threads. During the algorithm, there is only one management thread, which hampers the speed of the whole algorithm. This thread deals with time-consuming partitioning and data collection. This is the simplest multi-threaded implementation—it does not introduce too many changes to the classical GA model. The only change is the division of tasks and their implementation by worker threads. Unfortunately, this implementation presupposes the existence of a number of synchronization points, which will translate into slower performance of the algorithm.

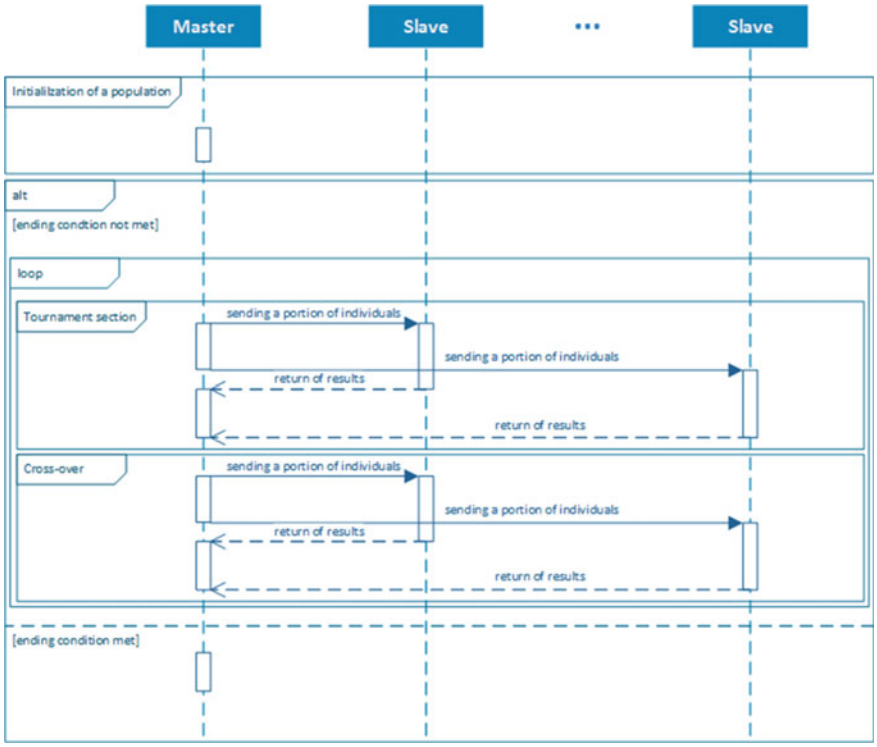


Fig. 2 Synchronous model—sequence diagram

2.3 Asynchronous Model

Another implemented model is asynchronous one. As the name implies, it assumes no synchronization points. For this reason, in this algorithm exists so-called empty runs, is due to the nature of its operation. Just like as in previous models, so this one comprises of a single population. A block diagram presents (Fig. 3) contains a diagram of a single thread version. Please note that this implementation is one of many, which may vary in terms of execution. The first step of the algorithm is initialization of the first population. Only after this step, the algorithm becomes fully asynchronous. In this particular implementation other than in all other models operators of selection and mutations were used. This is due to the lack of communication between, running in parallel threads. Selection and cross-over is carried out in a single method. This implementation uses a single population and a number of concurrent threads. In contrary to synchronous model, there is not distinct threads. There is no manager involved in tasks distribution. All operations take place without the synchronization. For this reason, there is a chance that another thread deletes the individual, which in another is going to be replaced. The more individuals a population has, it is less likely to eliminate the same individual. Therefore, one should keep in mind that this approach will work well (with a minimal empty runs) only if the population is large enough.

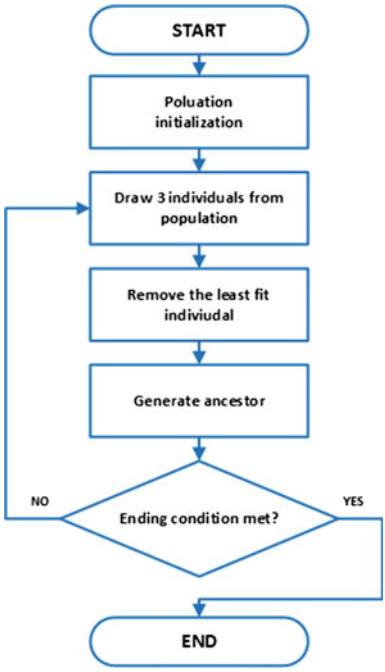


Fig. 3 Asynchronous model—a block diagram

Lack of synchronization in this model eliminates the necessity of any thread to wait for another before it "joins". Thanks to that CPUs should be utilized optimally (constant load assured). On the other hand, the lack of synchronization during the removal of least fit individuals from the population causing the empty runs. Although the calculations load processors well, in the case of drawing the same individual but more than one thread, the job of another will be futile. This asynchronous model operates on a single population. In this approach, there is no generation concept. Threads independently modify individuals, which are processed a different number of times. For this reason, the ending condition may not be the number of generations. The best solution, appears to be run an algorithm for a predetermined timeframe. When that time is passed, an individual with the highest value of fitness function is being chosen from the final population.

2.4 Island Model

The principle of its operation is based on the classical model. All operations within a single population take place sequentially. The most important difference compared to other models, is the existence of many populations evolving independently. This approach wouldn't change a lot a general operation, in a relation to the classical model, in some instances of the program. For the sake of faster solutions exchange within populations, an additional genetic operator was introduced (migration). Its operation is based on a random transition of an individual from one to another population. Jointly with cross-over and mutation, they contribute to accelerate diversification of currently viewed solutions subspace. It is worth to mentioned about a concept of neighborhood, which defines mutual location of the populations. In the implementation following model of the neighborhood was taken.

Migration is dependent on the location in the neighborhood, the connection between populations allow individuals to move between them. As shown in Fig. 4, all populations are interconnected. This allows for free flow of individuals between

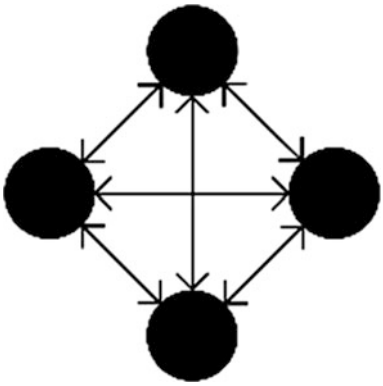


Fig. 4 Island model—populations in a neighborhood

them. Thanks to this operation, spreading solutions between populations may be faster. Each island has been implemented as a separate thread. The number of threads directly translates into a number of available CPU cores. Due to penalty hit when switching a context—it is not worth to use too many threads. This function is run by each thread spawn. There is no distinction between different types of threads, since this model does not have a thread supervisor. Each population sequentially run the algorithm similar to the classical one (Fig. 5).

In order to implement the migration operator .NET collection type *ConcurrentQueue* was used. Its construction resembles a regular queue, but it is available in all in parallel threads. Environment takes care of the synchronization and make it at the right moment so as not to slow down the program. Migrations are the only points of synchronization. "Good" solutions propagation between populations wouldn't be possible without them. In such a case, the model would work as classical model run a number of times. Please note that this model of genetic algorithm runs on many parallel populations. For this reason, their amount should be adjusted to the available CPU cores in a system. Switching threads is an expensive operation, increasing number of populations we will decrease performance of the algorithm.

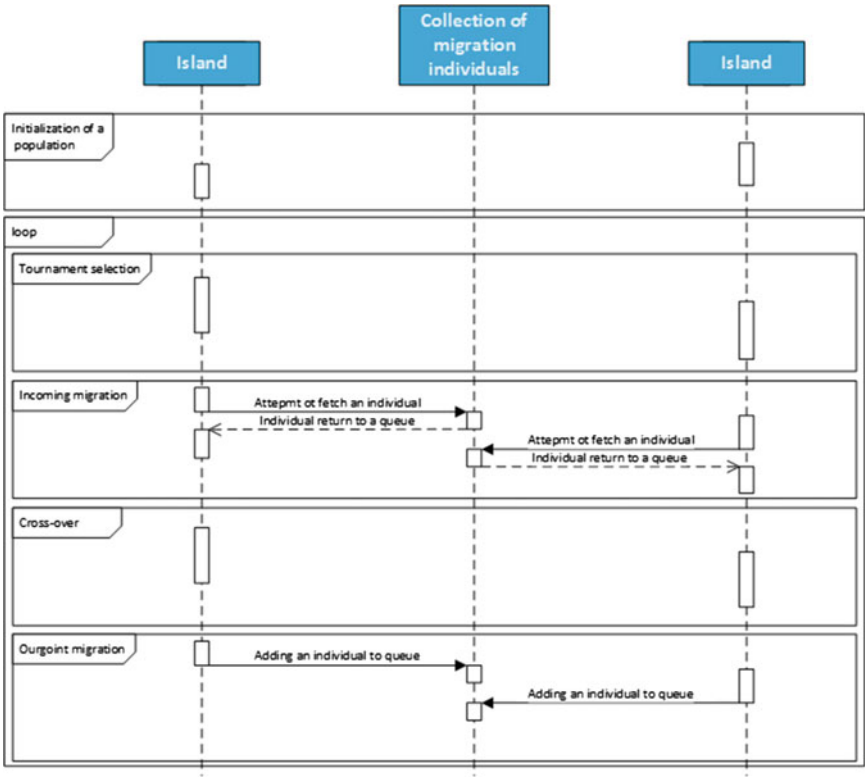


Fig. 5 Island model—a sequence diagram

3 Experiments

As a basis for experiments on implemented algorithms well know Traveling Salesman Problem [8, 10] (its asymmetric version) was chosen. Representation of the chromosome is a combination of vertices that a salesman traverses (403 points). The optimal solution equals to 2465. Asymmetry means that the distance from point A to point B is different from the way from point B to point A. Such formulated problem was sufficient to observe the difference in consumption of the resources by implemented algorithms. All GAs were implemented in .NET and tested on two 64-bit Intel CPUs—i5-760 and i7-2630QM. All tests were carried out on newly installed 64 bit Windows 8.1 Enterprise operating system. The latter CPU was tested with and without Hyper Threading (with 4/8 logical units) (Fig. 6).

First experiment was based on gradual increase of size of required generation with the fixed population size (500). Asynchronous model was not included in that experiment due to the fact that there is no notion of generation defined for that model. The execution time was assumed as resource (Fig. 7).

Next experiment, assumed generation count to be fixed at 1000, and population vas variable (Fig. 8).

Following experiment tested CPU usage based on a GA model used. Tests were conducted with Hyper Threading on and off (4 core CPU with HT on/off = 4/8 Threads) (Fig. 9).

Asynchronous GA performed the best due to lack of sync points, Island algorithm has just a few synchronization points, that is why it performed comparably to the asynchronous one. Number of populations in Island model were equal to number of logical cores. Next experiment tested number of logical processors influence on performance of Island GA. Main idea was to check whether the amount of the islands should be equal to the number of logical processors. As it could be expected

Fig. 6 TSP—Optimal route for 16862 Italian cities

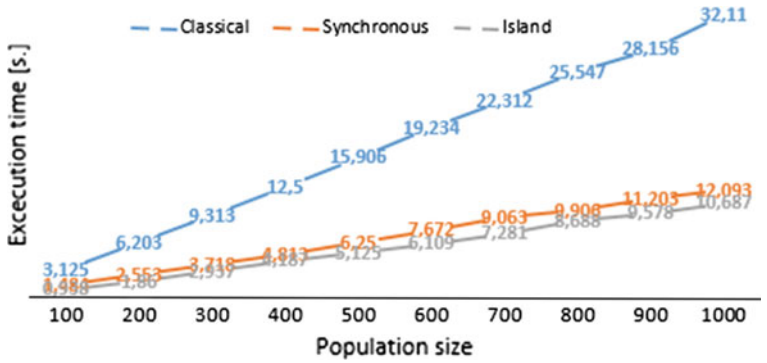


Fig. 7 Results for fixed population (500) and variable generation

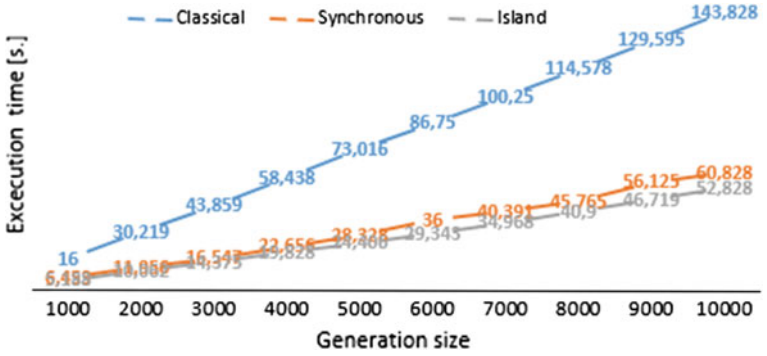


Fig. 8 Results for fixed generation (1000) and variable population

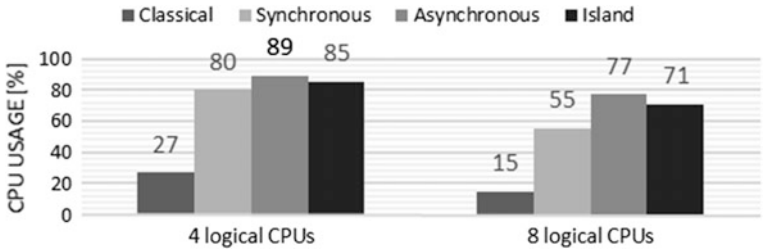


Fig. 9 CPU usage—GA Models versus 4/8 logical CPUs

increased number of islands amplified CPU usage and increased significantly computation time (when number of island > logical cores) (Fig. 11) this is due to Migration Operator used in Island GA that enforces parallel operations on populations. When the number of populations > number of logical CPUs then computation time increases noticeably caused by a context switching (with multiple threads).

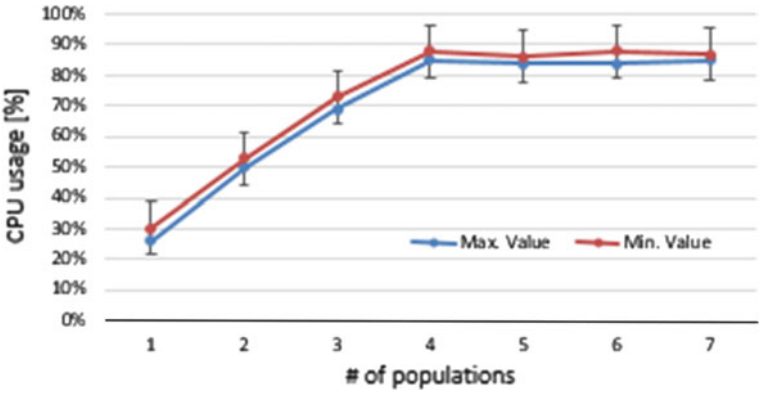


Fig. 10 Island model—CPU usage (4 logical CPUs)

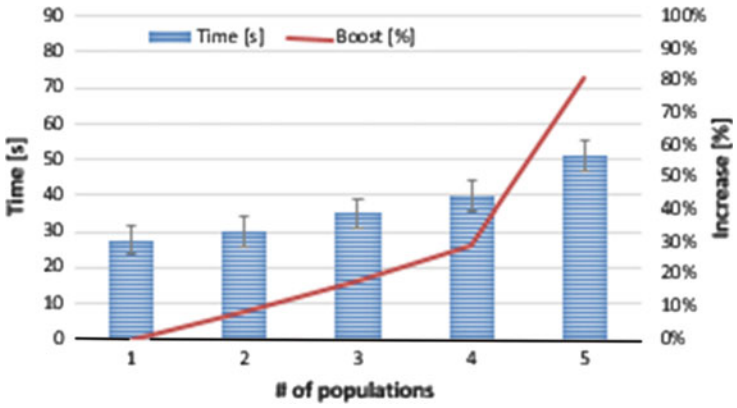


Fig. 11 Island Algorithm—running time/% increase (4 logical CPUs)

CPU consumption generally never goes beyond 90 % however with every additional thread more than number of logical cores will make algorithm running time considerably longer (Fig. 10).

As it was in the case tests with CPU with 4 logical processors the tendency holds. In order to test the hypothesis about number of thread equaled to number of logical cores Island algorithm running time was measured a number of times. When the population number exceeds 4—it is clearly visible for both 4 and 8 logical processors that time required for processing ceased be linear—there is an overhead related to Hyper Threading, additionally with more than 8 populations processing time lengthen even more (Figs. 12 and 13).

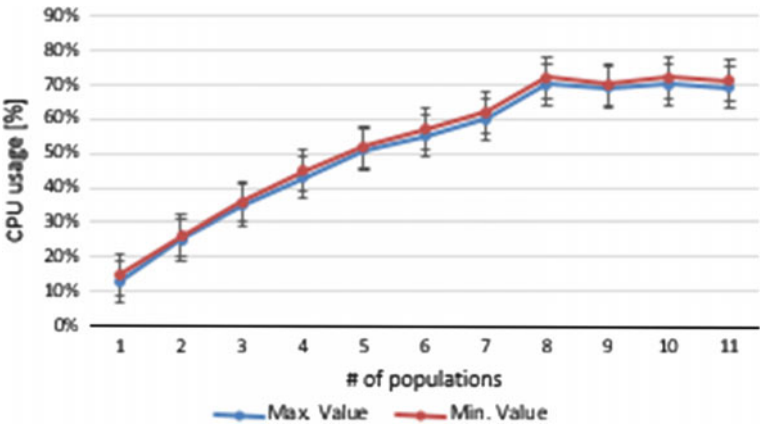


Fig. 12 Island model—CPU usage (8 logical CPUs)

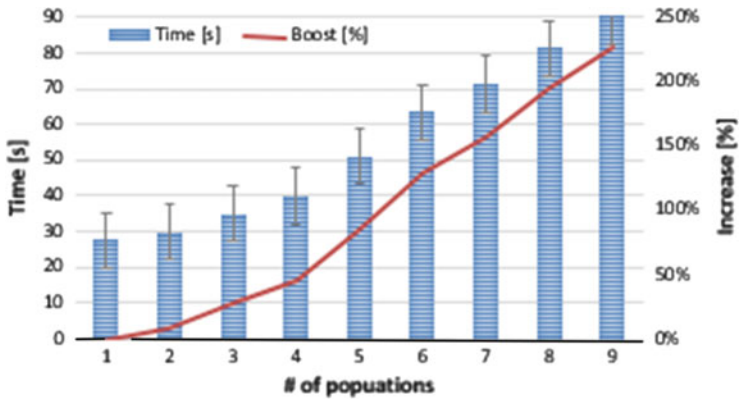


Fig. 13 Island Algorithm—running time/% increase (8 logical CPUs)

4 Conclusions

The work presented in the paper was devoted to the study genetic algorithms resources consumption and efficiency. During its implementation we were able to check out some aspects of these algorithms, but also managed to generally examine the applications running on systems with multicore processors. Analysis of the results allowed us to draw several noteworthy conclusions. Multithreaded applications are not the easiest way to implement the genetic algorithms. A lot of time one has to spend on their design. The first step should be to select target architecture on which the program is implemented. Each one may work differently and one should know how to implement the algorithm so as to make best use of available resources.

The actual implementation of the program should be preceded by a phase of design and research aimed at clarifying the best possible implementation of an algorithm. There are many approaches as to how genetic algorithms should be implemented and it is difficult to unanimously tell which one is an optimal one for a given problem. It all depends on the knowledge and experience of the person implementing an GA for a given problem. In addition to the many ways to achieve genetic operators, there are also many models of the same algorithm in multithreaded version. In the paper, authors dealt with a number of implementations which provided a set of comparable results. All the test workstations, on which the tests were conducted, were equipped with different type of multicore processors. It turned out that such an architecture is not well suited to work with implemented GAs (that use a number of threads in parallel) due to heavy penalty hit caused by threads switching. Processors with Hyper Threading enabled retuned even worse results than the same one with this feature turned off, due to a known fact that processors that contain the same number of logical processors, as physical ones, are better suited for constant heavy load of all cores. Generally speaking, the number of threads that an algorithm run should not exceed the amount of physical processors available for the system in order not to cause excessive loss of computational power.

References

1. Alba, E., Troya, J.M.: Analyzing synchronous and asynchronous parallel distributed genetic algorithms. Future Gener. Comput. Syst. **17**(4), 451–465 (2001)
2. Arora, R., Tulshyan, R., Deb, K.: Parallelization of binary and real-coded genetic algorithms on GPU using CUDA. In: 2010 IEEE Congress on Evolutionary Computation (CEC). IEEE (2010)
3. Blickle, T., Thiele, L.: A comparison of selection schemes used in evolutionary algorithms. Evol. Comput. **4**(4), 361–394 (1996)
4. Cantú-Paz, E.: A survey of parallel genetic algorithms. Calculateurs paralleles, reseaux et systems repartis **10**(2), 141–171 (1998)
5. Gen, M., Cheng, R.: Genetic Algorithms and Engineering Optimization, vol. 7. Wiley (2000)
6. Golub, M., Jakobović, D.: A new model of global parallel genetic algorithm. In: Proceedings of the 22nd International Conference on Information Technology Interfaces, 2000 (ITI 2000). IEEE (2000)
7. Gordon, V.S., Whitley, D.: Serial and Parallel Genetic Algorithms as Function Optimizers. ICGA (1993)
8. Hoffman, K.L., Padberg, M., Rinaldi, G.: Traveling salesman problem. Encyclopedia of Operations Research and Management Science, pp. 1573–1578. Springer, US (2013)
9. Muhlenbein, H.: Evolution in Time and Space-the Parallel Genetic Algorithm. Foundations of Genetic Algorithms (1991)
10. Pospichal, P., Schwarz, J., Jaros, J.: Parallel genetic algorithm solving 0/1 knapsack problem running on the gpu. In: 16th International Conference on Soft Computing MENDEL, vol. 2010 (2010)

11. Renner, G., Ekárt, A.: Genetic algorithms in computer aided design. Comput. Aided Des. **35**(8), 709–726 (2003)
12. Thierens, D.: Selection schemes, elitist recombination, and selection intensity. ICGA (1997)
13. Vasconcelos, J.A., et al.: Improvements in genetic algorithms. IEEE Trans. Magn. **37**(5), 3414–3417 (2001)

Monitoring Reliability of Embedded Systems

Janusz Sosnowski and Karol Zakrzewski

Abstract The paper deals with the problem of monitoring reliability issues in embedded systems. In particular, we concentrate on data reported during development, testing, operation in the field and service. For this purpose a special tool has been developed which provides the capability to collect data on-line and perform various analyses. The usefulness of this approach has been illustrated for some real embedded systems produced and serviced for several years in a commercial company. We present the interpretation of the obtained results, which proved their practical significance.

Keywords Reliability · Monitoring system development and maintenance

1 Introduction

The recent progress of nano technology resulted in proliferation of complex embedded systems in all domains of human life, science, medicine, automotive industry, controllers, etc. These systems are becoming more and more intelligent and their practical significance and roles are increasing. Hence, many traditional applications can be enhanced with new and more sophisticated functionalities. Moreover, various cyber-physical systems are getting much interest [2] and economically justified. All these systems are characterized by strong interaction with the environment via various sensors, control elements and communication channels. Depending upon the application we face some specific requirements such as high performance, low energy consumption, high dependability, safety, security, etc. Various hardware components are integrated with sophisticated software to

J. Sosnowski (✉) · K. Zakrzewski
Institute of Computer Science, Warsaw University of Technology,
Nowowiejska 15/19, 00-665 Warsaw, Poland
e-mail: J.Sosnowski@ii.pw.edu.pl

K. Zakrzewski
e-mail: karol_zak@wp.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_39

assure different kinds of signal and information processing quite often operating in real time fulfilling tight schedules.

The paper focuses on the methodology of tracking the problems during development and maintenance (operation) of embedded systems to assure their high dependability. We also take into account resilience requirements which result from changes in the environment, new operational schemes, new functional demands, etc. There is a lot of literature on monitoring schemes targeted at big computer systems, e.g. monitoring event [1, 15] or performance logs [11, 13]. They can be adapted to embedded systems. However, we have to face particular problems related to embedded systems, e.g. limited device resources, communication problems, impact of external environment and application diversity. An important issue is to take into account the specificity of different phases of the system life cycle and create a common knowledge database. The paper is based on our experience gained in developing several systems (measurement, industrial, satellite and automotive on board controllers [5, 8, 17, 18]).

Section 2 discusses different aspects of monitoring various features in correlation with different phases of the system life cycle. Section 3 presents the developed ERP sub-system for monitoring development, operational and service logs of embedded systems. Some practical results are given in Sect. 4. They relate to collected data within a SME company producing intelligent measurement devices and industrial controllers. Final conclusions are summarized in Sect. 5.

2 The Scope and Goals of Monitoring

Various monitoring techniques are widely used in general purpose computers. They are targeted at specific problems, e.g. software debugging or testing [7], system performance, software development and maintenance [9, 14]. Many systems provide event logs describing operating system, application, security mechanisms activities and problems. These logs comprise huge amount of data describing the status of system components, operational changes related to initiation or termination of services, configuration modifications, software upgrades, execution errors, etc. Hence, extracting interesting data is a challenge (we discussed this in [15, 16]). In this process some measures of log quality can be introduced, e.g. based on checking the entropy of log files [3]. To reduce log verbosity some coding can be included [12]. We can also monitor various performance parameters by activating appropriate counters [11, 20]. While designing log strategies and rules some care is needed taking into account two goals: detection and diagnostics of a problem, tracing the propagation of fault effects.

It is not possible to transfer directly monitoring schemes from general purpose computer systems to embedded systems due to combined hardware and software problems, limited system resources, specificity of applications, interaction with complex environment, etc. We faced this problem in practice in relevance to developed embedded controllers for gas counters (long time and massive production),

satellite on board controller and some mobile specialized equipment. In these applications an important issue was imprecise initial characterization of operational profiles and environment impact. Moreover, collecting data from the field was a challenge. Hence, an important issue was to integrate various dimensions of monitoring over all life cycle phases and create a common knowledge data base.

The possibilities of monitoring depend upon the life cycle phases. In the initial phase of the project we can use specially instrumented software to collect various problem oriented data, e.g. time distribution of performed tasks, power usage. In subsequent phase we rely more on reports generated by the involved personnel (e.g. testers, service). However, some embedded monitoring mechanisms in the system can improve diagnostic or data acquisition processes. Some special monitoring techniques are also targeted at debugging (e.g. software trace recording tools [10, 19]), they provide a bulk of data, e.g. tenths of thousand messages per second and are used to diagnose faults. In our research we are interested in higher level properties related to system reliability and functionality in different operational profiles.

Tracing techniques should minimize memory, processing and communication overhead. The data probes can be defined and instrumented in code based on developers and testers knowledge (e.g. specific function calls and exits). Monitored data can be further analyzed off-line, however some critical situations can be specified and detected on line including automatic reaction, e.g. unauthorized access, approaching to battery capacity limit. Such data may show possibilities of further performance improvements or facilitate debugging [19]. More detailed tracing can be instrumented during development or testing and then significantly reduced during the operational phase. However, we can admit higher degree of monitoring for a sample of devices contracted with some customers (users) to get better knowledge from the field.

During exploitation we are interested in three goals; collecting data on environment interaction (e.g. temperature changes, power disturbances, external interference), operational profile, critical events and their contexts (e.g. system restarts, triggering predefined alarms). Depending on the application these data are accessible directly in a continuous way (e.g. internet) or in a limited way, e.g. communication with an orbiting satellite. This knowledge is useful for developing new releases, and versions. In the case of long life systems we can mitigate system deficiencies by their reconfigurations. However, this process may need special care if performed remotely [8]. Tracing data from the field and remote reconfiguration impose some additional payload in communication, which in practice can be significantly limited.

In systems with reliable connections (e.g. flow counters over gas pipes communicating via RS 485 or internet channels) we faced the problem of the allowed maximal payload for monitoring due to limitations in processing, transmissions and power usage (for battery systems). Moreover, due to various legislation requirements the monitoring and reconfiguration logic or software have to be proven to be isolated form sensitive parts of the device. Here, some authorization is needed in relevance to different classes of data and users. Similar issues we have encountered in intelligent electric power meters widely used in houses and industry.

Limited overhead of embedded monitoring mechanisms in the target system results in careful selection of monitored points and data collection techniques. In particular, we have to generate aggregated information, identify anomalous states, generate alarms, etc. They are application and environment dependent. Some operational situations are of low probability or may relate to appearing faults or workload stresses. Hence, we should also monitor system behavior in such situations. They can be created in an artificial way by simulating various environment behaviors [2, 17] or injecting faults [2, 14]. We have developed special tools for this purpose and used them also for embedded systems (e.g. [4, 5, 7, 17]).

While searching for system anomalies we can target on some specific problems and correlate them with appropriate events, environment conditions, performance variables, etc. Some of these problems can be obvious (e.g. system failures), others may need some practical experience (e.g. those related to performance aspects). Another issue is defining unique and effective symptoms which assure high accuracy and precision of forecasting, this can be a challenge in some cases, e.g. cyber-attacks [20]. One step further is looking for unknown problems, which in fact can be hidden and not harmful at least for some time, however they grow systematically and result in dangerous situations. Looking for such potential threats needs observing many variables in the system, finding their profiles, trends, etc. Monitoring various objects and selected variables we can identify operational profiles in relevance to different system workloads, operational times, etc. We can look for statistical deviations of the monitored variables, e.g. increase or decrease in the mean value as compared with reference conditions. Such deviations have to be referred to system operation profiles.

In the case of known problems we can optimize problem detectors by selecting most sensitive measures which are not disturbed by the system workload. In the case of unclear problem symptoms we have to observe a wide spectrum of parameters. Their selection needs long time experience with existing systems. We have developed a special tool for visualization of collected data [15]. It is based on our previous experience [11] and involves fine/coarse grained views, various time perspectives, result aggregations, correlations, etc. In practice, we can use multi-dimensional monitoring, extract characteristic data and create some kind of data warehouse (knowledge data base). Performing retrospective analysis we focus also on possible global and local correlations (between objects of different dimensions). Combining this with service and user remarks we can refine anomaly detection rules, etc.

3 ERP System for Monitoring

Having gained some experience within several projects of embedded systems we have developed a system (ProductSystem) for monitoring device lifecycle adapted to the specificity of a small company. It is some kind of ERP systems targeted at controlling activities related to development, manufacturing and service.

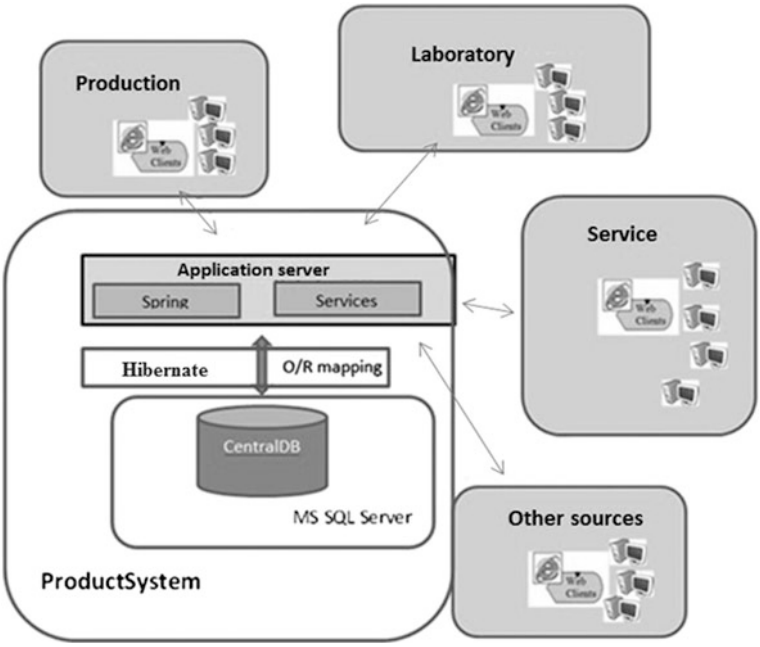


Fig. 1 The structure of global monitoring system

In particular, it can register reports from various company departments related to production (development and testers, measurement laboratory), service, and end users. Most reports are generated manually via electronic forms, others in a semiautomatic or automatic way (e.g. measurement reports, system logs). The structure of the system is shown in Fig. 1.

The relational database (CentralDB) has been developed using SQL DDL scripts. The data base model has been generated with Microsoft SQL Server Management Studio. It comprises many tables related to production, laboratory measurements and service/exploitation information. In particular, we have data for each produced device on the configuration, shipment to the user, detailed report of calibration measurements, reported problems, service history, etc. Hence, this database integrates data related to the whole lifecycle of each device. The developed supervising application uses Spring technology and smartGWT library to provide electronic forms accessible to various system users and used for loading and analyzing data. The intermediate layer based on Hibernate architecture interconnects data base with the supervising application. Different system users have different access capabilities needing authentication, this can be performed using typical web browsers. In general, we distinguish 4 classes of users: authorized personnel in production, service, laboratory and others. Within the laboratory various specialized measurement are performed, in particular calibration and accuracy checking (including legalization performed by special personal), electromagnetic interference resistance checks, etc.

Some data in laboratory are generated in an automatic way and can be transferred to the database using special data conversion scripts. Within the group of other users we have customers, company administration, etc. Here, we can also include the stream of data collected from device monitoring (various logs).

Designing the database and the supervising systems we assured high flexibility. Hence, we provide the ability of adding new tables to the data base model, creating new or modifying existing electronic forms, redefining or extending list of faults, etc. In this processes various general templates have been prepared. This feature resulted from the gained experience during long time activity of the considered company which showed changes in production assortment and used technologies, company organization, administration, client needs, etc.

The integrated data base stores collected data over a long period of production and servicing various products, mostly gas flow counters. These counters are dedicated for supervision of gas pipe systems and gas receivers. They cooperate with various sensors and perform sophisticated calculations in accordance with national and international norms. So, the collected data can be used to assess various aspects of manufacturer activities, product operation in the field and its quality. For an illustration we present some selected results. As opposed to, classical ERP systems we deal with heterogeneous data related to many domains. Having filled database with the available data and performed analysis we have identified some lacking information, which could be very useful for better assessment of manufacturing and servicing processes, this has been previewed in the developed system. All identified faults were classified in over 50 groups (however their distribution was uneven 0.1–10 %). Deeper analysis allows us to correlated faults with different users, versions, external environment (e.g. open air or air conditioned places).

4 Practical Results

We have monitored various embedded systems developed and serviced by a medium size company. We give some results characterizing reliability issues. Figure 2 presents cumulative plots of produced and serviced devices of type A for subsequent quarters within the period of 3 years and a half. In this period about 2600 devices have been manufactured and 435 repaired in service. Within the repaired devices only 312 devices have been produced before the considered time period, so we can assume that 4.7 % of faulty devices returned from exploitation.

These reliability statistics can be complemented with mean time to failure (MTTF) parameter. MTTF distribution (in percent) for device of type A is presented in Fig. 3. Here, almost 8 % of devices returned to the service after 50 days, then we have some peak at 350 and 500 days. The average time to service was 220 days. Taking into account the fact that the considered devices perform accurate measurements we were interested in stability of these measurements over time. The result of such analysis is shown in Table 1 for two types of devices (A, B). The

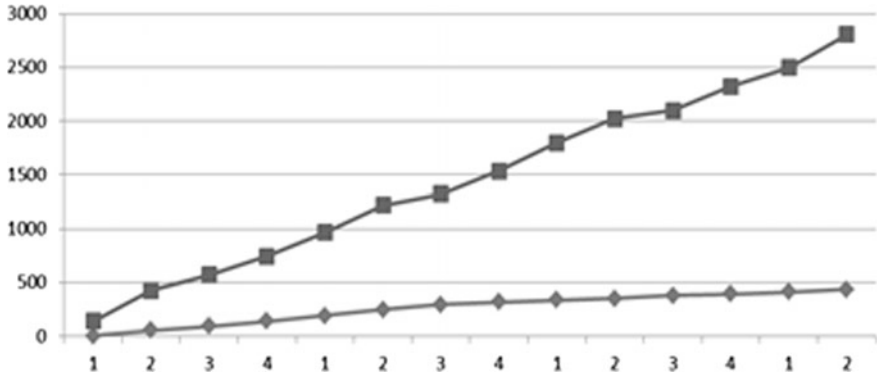


Fig. 2 Cumulated number of produced (*upper plot*) and serviced (*lower plot*) devices of type A

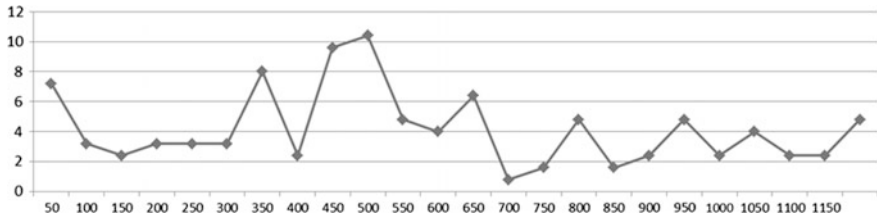


Fig. 3 Distribution (in percent) of MTTF (in days) for type A devices

Table 1 Measurement errors

Temp.	Absolute error averaged over all devices of the same type					
	A(T)	A(T)*	A(P)	A(P)*	B(T)	B(T)*
−25°	0.016	0.010	0.070	0.011	0.450	0.245
+23°	0.043	0.045	0.011	0.014	0.460	0.245
+55°	0.017	0.026	0.012	0.014	0.052	0.365

table specifies absolute errors of measuring temperature (T) at points −23, 23 and 55 °C (for devices of type A and B) and pressure (P) in kPa (for devices of type A) just at the moment of the first shipment to the users and after receiving for the service (columns marked with *). Gas pressure was measured in the range 0–600 kPa. The relative measurement errors were in the range 0.25–0.5 % depending upon the measured point. It is worth noting that before calibration for about 10 % of devices the relative error exceeded 0.5 % for middle range measured values, for extreme values we observed this for 40 % of devices. The accuracy of calibrated devices was quite stable, only small percentage of serviced devices showed some higher deviation.

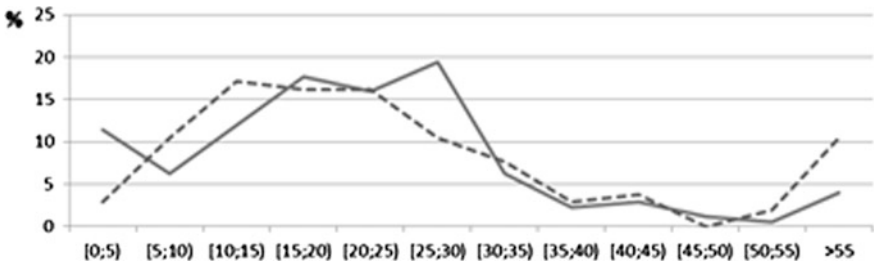


Fig. 4 Distribution (in percent) of repair time (in days) for C1 and C2 devices

Figure 4 presents distribution of repair time in service (the x axis specifies time slots in days, y axis shows percentage of devices attributed to the time slot). Most devices have been repaired within 10–25 days. Originally collected data specified only the moments of receiving faulty device and sending them out after repair. The introduced electronic forms allow us to include more data, e.g. time of analysis, time of device calibration, legalization, waiting time for spare elements. Some complementary views on service effectiveness are presented in the sequel.

It is worth noting that devices leaving the service are not handled in the same order as they are coming. This is illustrated in Table 2, the numbers in odd rows (in bold) show the order number of the received devices, the even rows show the service delay. The first serviced device was the 6th received, the 1st received device was serviced after 4 devices which were received later (6, 2, 3, 4). Here, the longest delay was 41 days and the shortest 7 days. Unfortunately, the reasons of this situation are not presented due to the above mentioned lacks in reports. Figure 5 shows the distribution of the number of accepted devices for service (3 plots covering 3 types of devices) for 4 years with 3 month (quarter) period resolution. More detailed view can be generated with finer resolution. This is illustrated in Fig. 6 for devices of type A.

It is also interesting to correlate service requests with individual users. Over 50 % of serviced devices appeared only once and 20 % twice, 15 % three times. One of the devices has been repaired 16 times. Within the serviced devices dominated those which operated in specific open air conditions.

The embedded monitoring mechanisms within devices collect and aggregate various data. In particular, they detect communication breaks with sensors, alarm

Table 2 Service delays (in days)

6	**2**	**3**	**4**	**1**	**7**	**10**	**8**	**11**	**12**	**13**	**14**	**15**
7	15	10	10	21	14	14	16	15	14	14	14	14
19	**21**	**5**	**20**	**22**	**25**	**18**	**23**	**26**	**30**	**24**	**16**	**17**
12	13	29	15	15	13	22	16	14	14	28	41	41

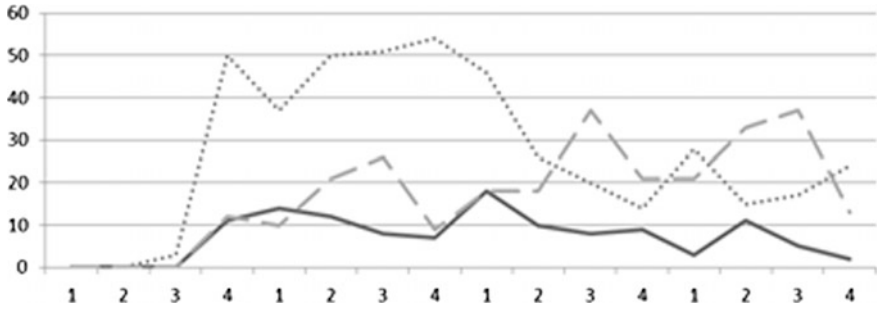


Fig. 5 The number of serviced devices A (*upper dotted plot*), B (*bold lower plot*) and C (*dashed line*) in subsequent quarters of 4 years

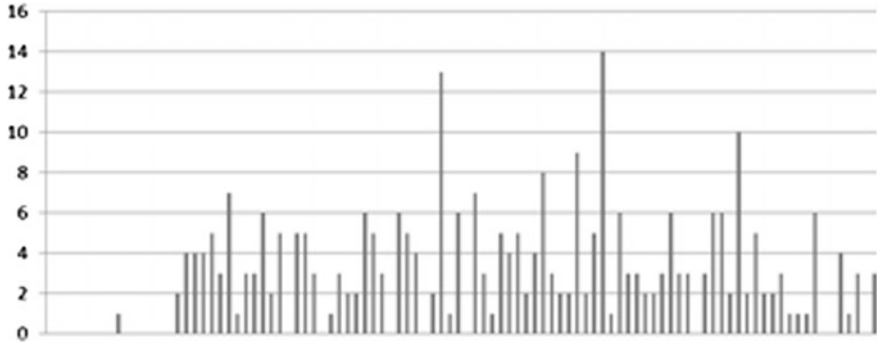


Fig. 6 The number of devices of type A accepted for service within 100 weeks

situations caused by non-acceptable values of specified variables, approaching to critical level of battery capacity, unacceptable activities, e.g. unauthorized tries to access critical data, violations of allowed functionalities, physical violations (e.g. breaking sealed accesses, changing critical parameters), statistics of communication faults, number and sources of restarts, unsuccessful archiving, etc. Depending upon device configuration and user the monitored data is collected in some external database or locally in the device. In the last case higher degree of compaction and aggregation is involved. The collected data is sent either directly to the ERP system or manually loaded periodically or during the service time. In this way we can provide various statistics and derive some common problems, correlate them with other contexts (e.g. software versions, environment specificity) and trigger improvement actions within the device life cycle. The developed system is flexible and reconfigurable, so it can be easily adapted to other products (devices) or manufacturing and service schemes.

5 Conclusions

System monitoring is useful to prevent or detect system anomalies and trigger appropriate handling procedures within the system working in the field. Moreover, it extends our knowledge on system operational profiles, interactions with the environment, etc. The collected data (related to different perspectives and system life cycles) is helpful in improving product quality in subsequent releases, improvement of development and maintenance procedures, appropriate upgrades and reconfigurations of the working devices, etc. In well-designed systems anomalies occur scarcely so assessment of the effectiveness of the monitoring schemes involves long operation observations from many system instances. This problem is becoming more critical in non-massive applications (e.g. satellite missions). Hence a good alternative is to verify the effectiveness by fault (anomaly) injections and extensive simulations. We used this techniques in some developed projects and found it as very useful [4, 5, 17].

In embedded systems we have various limitations, so the monitoring probes (hardware or software) have to be designed carefully (low intrusiveness) taking into account application specificity and various environment interactions. Moreover, direct contacts with users are of greater importance, especially in applications integrated with various physical environment. This results in the need of creating specialised knowledge database. It will be useful in deeper analysis of fault sources, their types, severity levels and mitigation effectiveness (compare [6]). Typically, general purpose systems provide us with a bulk of highly redundant logs, so the main problem is data exploration. In the case of embedded systems we are more concentrated on designing application and problem dependant logs in close cooperation with the potential users.

Our future research work will focus on deeper investigation of collected data logs from the field in relevance to various devices. We also plan to improve information content and accuracy of collected data. Combining monitoring tasks with on-line system reconfiguration is another challenge.

References

1. Baccanico, F., et al.: Event logging in industrial development process: practices and engineering challenges. In: IEEE International Symposium on Software Reliability Engineering Workshops, pp. 10–13 (2014)
2. Backer, M., Kuznik, Ch., Mueller W.: Virtual platforms for model-based design of dependable Cyber-Physical system software. In: Proceedings of Euromicro Conference on Digital System Design, pp. 246–253 (2014)
3. Cinque, M., et al.: Assessing direct monitoring techniques to analyze failures of critical industrial systems. In: IEEE 25th Symposium on Software Reliability Engineering, pp. 212–221 (2014)
4. Gawkowski, P., Ławryńczuk, M., Marusak, P., Sosnowski, J.: Software implementation of explicit DMC algorithm with improved dependability. In: Tarek, S. (ed.) Novel Algorithms

and Techniques in Telecommunications, Automation and Industrial Electronics, pp. 214–219. Springer, Berlin (2008)
5. Gawkowski, P., Pawełczyk, P., Sosnowski, J., Gajda, M.: LRFI—Fault injection tool for testing mobile software. In: Ryżko, D., et al. (eds.) Emerging Intelligent Technologies in Industry, Studies in Computational Intelligence, vol. 369, pp. 269–282. Springer (2011)
6. Hamil, H., Goseva-Popstojanova, K.: Exploring fault types, detection activities, and failure severity in an evolving safety-critical software system. Software Qual. J. **23**, 229–265 (2015)
7. Iwiński, M., Sosnowski, J.: Testing fault susceptibility of a satellite power controller. In: Zamojski, W., et al. (eds.) Dependable Computer System, Advances in Intelligent and Soft Computing, vol. 97, pp. 63–74. Springer (2011)
8. Iwiński, M., Graczyk, R., Sosnowski, J.: Dependability issues in the PPLD-PSU subsystem for the BRITE-PL Hevelius microsatellite. Pomiary-Automatyka-Kontrola, PAK **60**(7), 447–449 (2014)
9. Janczarek, P., Sosnowski, J.: Investigating software testing and maintenance reports: case study. Inf. Softw. Technol. **58**, 272–288 (2015)
10. Kraft, J., Wall, A., Kienle, H.: Trace recording for embedded systems: lessons learned from five industrial projects. In: Runtime Verification, LNCS, vol. 6418, pp. 315–329. Springer (2010)
11. Król, M., Sosnowski, J.: Multidimensional monitoring of computer systems. In: Proceedings of IEEE UIC-ATC 2009 Symposium and Workshops on Ubiquitous, Autonomic and Trusted Computing in conjunction with the UIC'09 and ATC'09 Conferences, pp. 68–74 (2009)
12. Schuster, H., et al.: A log tool suite for embedded systems. In: 7th International Conference on Advances in Circuits and Microelectronics, pp. 16–20 (2014)
13. Sirbu, A., Babaaoglu, O.: Systematic analysis of cluster computing log data: the case study of IBM BlueGene. arXiv:1419.4449v2 (2015)
14. Skarin, D., Kaarlsson, J.: Software implemented detection recovery and soft errors in a break-by-wire system. In: Proceedings of 7th EDCC Conference, IEEE Comp. Soc. pp. 145–154 (2008)
15. Sosnowski, J., Kubacki, M., Krawczyk, H.: Monitoring event logs within a cluster system. In: Zamojski, W., et al. (eds.) Complex Systems and Dependability, Advances in Intelligent and Soft Computing, vol. 170, pp. 259–271. Springer (2012)
16. Sosnowski, J., Gawkowski, P., Cabaj K.: Exploring the space of system monitoring. In: Bembenik, R., et al. (eds.) Intelligent Tools for Building a Scientific Information Platform: Advanced Architectures and Solutions, Studies in Computational Intelligence, vol. 467, pp. 501–517 (2013)
17. Trawczyński, D., Sosnowski, J., Gawkowski, P.: Analyzing fault susceptibility of ABS microcontroller. Lecture Notes in Computer Science, LNCS **5219**, 360–372 (2008)
18. Trawczyński, D., Zalewski, J., Sosnowski, J.: Design of reactive security mechanisms in time-triggered embedded systems. SAE Int. J. Passenger Cars – Electron. Electr. Syst. SAE **7**(2), 527–535 (2014)
19. Trumper, J., Voigt, S., Dollner, J.: Maintenance of embedded system: supporting program comprehension using dynamic analysis. In: IEEE SEES Conference, pp. 58–64 (2012)
20. Ye, N.: Secure Computer and Network Systems. Wiley, Chichester. ISBN 978-0-470-02324-2 (2008)

Implementation Efficiency of BLAKE and Other Contemporary Hash Algorithms in Popular FPGA Devices

Jarosław Sugier

Abstract BLAKE is a cryptographic hash function which was one of the 5 finalists in the SHA-3 competition. Although it ultimately lost to KECCAK the cipher was repeatedly acclaimed during evaluation for its both good cryptographic strength and great performance especially in software realizations, and it is still selected as a hash function of choice in contemporary computer systems. The aim of this paper is to investigate how the elaborated particularities of BLAKE internal processing and data multiplexing are handled by automatic implementations tools when two low-cost FPGA platforms are targeted—the standard, well established Spartan-3 and the newer Spartan-6 devices from Xilinx, Inc. The cipher was implemented in high-speed architectures built from the standard iterative one with loop unrolling and (optionally) pipelining. Total of 5 different organizations were created and results of their implementations are compared with analogous results obtained for another two contemporary hash algorithms: Salsa20, which inspired BLAKE core processing, and KECCAK. Presented data illustrate how the fundamental mechanisms of loop unrolling with or without pipelining work in case of this particular cipher.

Keywords BLAKE hash function · Salsa20 · Keccak · Loop unrolling · Pipelining

1 Introduction

Contemporary computer information systems are complex amalgamates of technical, information, organization, software and human resources (users, administrators, technical support, etc.). In this context, ensuring appropriate security and confidentiality of information processing (at data acquisition, transmission, storage and retrieval levels) constitute one of the main and one of the most important

J. Sugier (✉)
Faculty of Electronics, Wrocław University of Science and Technology,
Janiszewskiego St. 11/17, 50-372 Wrocław, Poland
e-mail: jaroslaw.sugier@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_40

challenges in their design, implementation and maintenance. Malfunctions in system operation caused by security violations are so common that in dependability analysis they are treated similarly as "classic" failures in traditional reliability theory. To meet the challenge of their eradication it is necessary to apply appropriate cryptographic methods and in this paper we consider one class of such methods which is based on so called hash functions.

The aim of this work was to investigate implementation efficiency of BLAKE hash function—one of the five finalists in the SHA-3 contest—in popular Field Programmable Gate Arrays. In literature there is an extensive number of speed- or area-efficient implementations of this algorithm [3–5, 12] and it was not the goal of this work to develop yet another ones which would incrementally improve the parameters by some additional margin. Instead, our aim was to investigate how the elaborated particularities of BLAKE internal processing and data multiplexing are handled by automatic implementations tools when two low-cost FPGA platforms are targeted—the standard, well established Spartan-3 and the newer Spartan-6 devices from Xilinx, Inc. The cipher was implemented in high-speed architectures built from the standard iterative one with loop unrolling and (optionally) pipelining. Total of 5 different organizations were created and results of their implementations are compared in this paper with analogous results obtained in our previous works for another two contemporary hash algorithms: Salsa20, which inspired BLAKE core processing, and KECCAK—the winner of the SHA-3 contest.

Contents of this paper is organized as follows. In the next chapter we will briefly outline BLAKE background, define the algorithm and present its high-speed loop unrolled and pipelined architectures which were in scope of interest of this work. Then, in Sect. 3 we will discuss the results obtained after their implementation in the two FPGA chips and compare them with equivalent results of Salsa20 and KECCAK algorithms.

2 BLAKE Hash Algorithm

2.1 The SHA-3 Contest

The Message-Digest Algorithms developed by Ronald Rivest in the years 1989–92 were the first widely recognized cryptographic hash functions standardized by Internet Society as recommendations RFC 1319-21 (MD2, MD4 and MD5) and later used in the Secure Hash Algorithm SHA-1—a U.S. FIPS PUB 180-1 standard published in 1995. Although in 2002 an extended SHA-2 standard was announced, constant increase in available computational power eventually made them vulnerable to brute force attacks so in November 2007, foreseeing the need of an entirely new approach in hash design in order to compete with recent advances in cryptanalysis, the U.S. National Institute of Standard and Technology announced an

open contest for development of a novel SHA-3 standard. 14 of the proposed submissions passed the initial verification and, eventually, the best 5 algorithms—BLAKE, Grøstl, JH, Keccak and Skein—were selected for the final round in December 2010. From this group by NIST decision Keccak was selected as the winner.

Although BLAKE ultimately lost in the SHA-3 contest the cipher was repeatedly acclaimed in all stages of competition for its good cryptographic strength and great performance especially in software realizations. For these advantages it is still often selected as a hash function of choice in contemporary computer systems: for example, its variant was chosen as a checksum/validation method in a recent extension to RAR file archive format, or it was selected for password-based key derivation function NeoScrypt which is intended to become an informational RFC recommendation.

2.2 Definition of the Algorithm

The subject of this study is BLAKE-256—the variant of the cipher in which size of the words is 32b (leading to 512b state) and which produces 256b digest. To compute the hash [1], the message m of length $l < 2^{64}$ bits is first padded with bit string "10…01$[l]_{64}$" so that its total bit length is a multiple of 512 (where $[]_{64}$ means 64-bit unsigned big-endian representation) and then it is split into 512b blocks m^0, …, m^{N-1}. The hash value $h(m)$ is computed iteratively:

$h^0 = \text{IV}$
for $i = 0 \ldots N-1$
  $h^{i+1} = compress\,(h^i, m^i, s, l^i)$
return h^N

where upper index ^{i} denote ordinal number of the message block, and:

IV	initial value of the hash (a 256b constant same as in SHA-2 standard),
s	so called *salt*, a user-supplied unique 128b string parametrizing the hash,
l^i	number of message bits in m^i,
compress(h, m, s, t)	a *compression function* which handles one block of data.

Thus the processing of the message stream is reduced to the application of the compression function *compress*(h, m, s, t) and its implementation was the main subject of this work.

The function uses another 512b constant divided into 16 words $c_0 \ldots c_{15}$, and 10 permutations $\sigma_0 \ldots \sigma_9$ of the message words $m_0 \ldots m_{15}$—both are statically defined in

the specification. The state is organized in a 4 × 4 matrix of words $v_0 \ldots v_{15}$ which is initially filled with input data, in part xor'ed with constants $c_0 \ldots c_7$:

$$\begin{pmatrix} v_0 & v_1 & v_2 & v_3 \\ v_4 & v_5 & v_6 & v_7 \\ v_8 & v_9 & v_{10} & v_{11} \\ v_{12} & v_{13} & v_{14} & v_{15} \end{pmatrix} = \begin{pmatrix} h_0 & h_1 & h_2 & h_3 \\ h_4 & h_5 & h_6 & h_7 \\ s_0 \oplus c_0 & s_1 \oplus c_1 & s_2 \oplus c_2 & s_3 \oplus c_3 \\ t_0 \oplus c_4 & t_0 \oplus c_5 & t_1 \oplus c_6 & t_1 \oplus c_7 \end{pmatrix} \quad (1)$$

Then the state goes through $n_r = 14$ rounds. Each round modifies twice all v_i words by applying a so called *G function* for four sets of words:

$$\begin{aligned} &G_0(v_0, v_4, v_8, v_{12}); \quad G_1(v_1, v_5, v_9, v_{13}); \\ &G_2(v_2, v_6, v_{10}, v_{14}); \quad G_3(v_3, v_7, v_{11}, v_{15}); \end{aligned} \quad (2)$$

and then

$$\begin{aligned} &G_4(v_0, v_5, v_{10}, v_{15}); \quad G_5(v_1, v_6, v_{11}, v_{12}); \\ &G_6(v_2, v_7, v_8, v_{13}); \quad G_7(v_3, v_4, v_9, v_{14}). \end{aligned} \quad (3)$$

The first applications of the G function operate on words from each column of the matrix, while the second—on words from diagonals. This corresponds to column and row rounds in Salsa20 while the G function is equivalent to Salsa's quarterround ([2]).

Finally—at the lowest level—each function $G_i(a, b, c, d)$ for a round number $r = 0 \ldots 13$ is a sequence of the following operations:

$$\begin{aligned} a &= a + b + (m_{sr'(2i)} \oplus c_{sr'(2i+1)}) \\ d &= (d \oplus a) \gg 16 \\ c &= c + d \\ b &= (b \oplus c) \gg 12 \\ a &= a + b + (m_{sr'(2i+1)} \oplus c_{sr'(2i)}) \\ d &= (d \oplus a) \gg 8 \\ c &= c + d \\ b &= (b \oplus c) \gg 7 \end{aligned} \quad (4)$$

where $r' = r \bmod 10$ and the operators denote the following transformations of the words:

$\oplus$ bitwise xor of two bit vectors,

$+$ addition mod 2^{32} of two bit vectors (i.e. regular 32b addition ignoring carry out),

$\gg$ right rotation by a constant number of positions.

After 14 iterations, the state produced by the last round is xor'ed with the input h and the salt s to give the final value of the compression h':

$$h'_i = h_i \oplus s_{i \bmod 4} \oplus v_i \oplus v_{i+8}, \quad i = 0 \ldots 7. \tag{5}$$

2.3 Organization of the Algorithm in Hardware

The aim of this study was to evaluate high speed variants of BLAKE organization and to verify scalability of the cipher with respect to the number of rounds implemented in hardware, so for the test suite 5 architectures were selected as listed below. Their naming convention and internal organization follow taxonomy defined in [3, 4].

- "x1": the basic iterative organization with one round implemented in hardware and the state being passed though it repeatedly in 14 clock cycles (i.e. each complete round is computed in one clock tick);
- "x2": modification of the above with a combinational cascade of two rounds implemented in hardware with total computation done in 7 clock cycles (the loop is unrolled by factor 2 and in each clock tick the state is propagated through two rounds);
- "x5": the cascade is built from 5 rounds and 3 clock cycles are required for complete computation (the final result is taken from the fourth round in the cascade so that $n_r = 2 \times 5 + 4$);
- "PPL2": the modified x2 organization with pipeline registers added after each round: two chunks of data are processed in parallel (increasing the throughput twice) but the completion needs again 14 clock cycles since in one clock tick the state is transformed by one round;
- "PPL5": the pipelined x5 organization with 5 chunks of data processed in parallel and consequently higher throughput.

To implement in hardware one complete round of the compression function—which is the core of all the 5 architectures—one must first instantiate eight blocks of G functions, each with different permutations of the state words loaded to its inputs. Realization of this function was presented in detail in [9].

3 Results

3.1 Parameters of the Implementations

In all five architectures the hardware module computing the compression function was equipped with basic input/output registers providing means for iterative hashing of the message in 512b chunks. The same code was automatically

Table 1 Implementation parameters of the five architectures

		min. T_{clk} (ns)	Throughput (Gbps)	Levels of logic	% routing delay	Size: LUTs	Size: slices
Spartan-3	x1	45.7	0.80	66	50.6	9155	5415
	x2	88.9	0.82	118	51.2	16928	10039
	x5	244.1	0.70	258	61.7	41923	23232
	PPL2	47.3	1.55	69	50.7	16817	10375
	PPL5	47.4	3.86	50	55.4	18066	11882
Spartan-6	x1	28.7	1.28	35	64.3	5621	2460
	x2	67.7	1.08	70	72.3	10150	4409
	x5	185.1	0.92	131	78.1	24368	8833
	PPL2	30.4	2.40	38	65.6	10918	4144
	PPL5	35.1	5.21	44	68.9	24236	9816

synthesized and implemented by Xilinx ISE software with XST synthesis tool for two devices—Spartan-3 XC3S5000-5 [10] and Spartan-6 XC6SLX150-3 [11] which were selected to be sufficiently large to accommodate the most sized x5 or PPL5 architectures. The same methodology was applied in our previous works on Salsa20 [6] and Keccak [7] so an already existing test platform was uniformly extended to accommodate BLAKE cipher and to produce comparable results.

Table 1 presents results obtained after implementation. Speed aspect is represented by the value of the minimum clock period as it was estimated after static timing analysis of the final, fully routed design. From this parameter maximum theoretical throughput of the message stream was derived taking into account number of clock cycles needed to compute the hash and, in case of pipelined variants, number of data streams processed simultaneously in pipeline stages. Two middle columns provide parameters which illustrate effectiveness (or difficulties) of the implementation process, i.e. how the complex logical transformations of the algorithm were realized with programmable resources of the array: for the longest combinational path in the design the third column gives number of logic elements it contains (LUT generators) and the fourth—percentage of the propagation delay incurred by the routing resources (and not logic elements). Any significant rise in the latter parameter above 50–70 % indicates problems with routing of tracks between logic resources in the array: if the connections are too dense versus distribution of logic elements, routing becomes congested and implementation in the FPGA array becomes problematic. Size characteristics for each design are reported in the last two columns of the table which give the total numbers of utilized LUT generators and slices.

Detailed evaluation of these implementation was included in [9], while the following discussion will concentrate on comparison with equivalent results obtained for Salsa20 [6] and Keccak [7].

3.2 Comparison with Salsa20

The core of BLAKE algorithm—the compression function—was introduced by its authors as an extension of Salsa20 hash functions with two modifications in the *G* function (Eq. 4): (1) customized order and assortment of elementary operations as compared to the Salsa quarterround; (2) additional mixing of the first state word with message bits and the c_i constants as seen in the first and the fifth Eq. (4). Of these two alterations the latter had substantially greater impact on data flow in hardware.

In Salsa20 and in majority of other hash functions (including KECCAK) the message bits are only loaded as an input to the first cipher round in parallel with other data like salt, counter or nonce. That is, the message bits enter only beginning of the round cascade and are not routed to each round separately. In BLAKE two m_i words must be sent to each *G* function so effectively all 16 words are used in the whole round.

The authors introduced this new aspect as a relatively minor extension and indeed it may be so in software implementation: even if each *G* function operates in a separate thread of CPU execution, extra reads of RAM locations which store the message words did not alter the overall scheme of data handling and just added other operations to the sequence of already running ones. In hardware, though, this led to creation of a completely new, 512b wide data path which was not needed neither in Slasa20 nor in KECCAK organizations [6, 7]. This effectively doubled the total width of the data path running along the round cascade from 512b (the state) to 1024b (the state plus the message bits). Distribution the message words was additionally complicated by permutations $\sigma_0 \ldots \sigma_9$ used by the algorithm: each round needs different permutation of m_i so switching between them required supplementary multiplexers controlled by the round counter.

The effects can be seen in Fig. 1 which compares minimum clock period and size of the 5 organizations between the two ciphers: all BLAKE implementations are greater than equivalent Salsa20 ones by 97 ÷ 169 % in Spartan-3 and by 83 ÷ 113 % in Spartan-6. But, on the other hand, since the additional message tracks run in parallel to the main processing paths with state bits they did not produce significant increase in propagation time and had little impact on the minimum clock

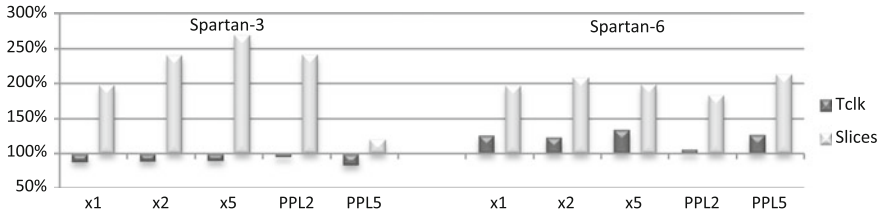


Fig. 1 Speed (minimum clock period) and size (number of occupied slices in the array) of BLAKE implementations as percentages of Salsa20 parameters

period. In the Spartan-3 device all BLAKE architectures run 5–15 % faster than Salsa but in Spartan-6–6 ÷ 34 % slower.

This disparity, on the other hand, is a noteworthy observation: more powerful logic resources found in the new FPGA family (LUT generators with number of inputs increased from 4 to 6) offered less of speed improvement for atomic BLAKE transformations (specifically for wide, 32b adders which are constructed from independent dedicated resources) than for those in Salsa20. Still, implementations of BLAKE run on average 30 % faster in Spartan-6 than in Spartan-3; in case of Salsa20 just this reduction was even greater.

3.3 *BLAKE, Salsa20 and Keccak: Scaling with the Loop Unrolling Factor*

Because implementation parameters are not directly comparable across completely different ciphers, in order to evaluate efficiency of loop unrolling and pipelining mechanisms in each of them a different approach, similar to that developed in [7] and [6], will be applied.

In every cipher the x1 architecture will be used as a point of reference in evaluation of the remaining cases. Size of every unrolled architecture should increase proportionally to the number of rounds implemented in hardware (the unrolling factor) and we will estimate

$$\begin{aligned} Size_{\mathrm{x}k} &\approx Size_{\mathrm{x1}} \cdot k \\ Size_{\mathrm{PPL}k} &\approx Size_{\mathrm{x1}} \cdot k \end{aligned} \tag{6}$$

Additional registers which are added in the pipelined organizations usually do not introduce any extra burden in the FPGA arrays and therefore the above equations are assumed identical for both x*k* and PPL*k* cases. Maximum frequency of operation—or the minimum clock period—depends on the other hand on the number of rounds the state must go through in one clock cycle, i.e.:

$$\begin{aligned} Tclk_{\mathrm{x}k} &\approx Tclk_{\mathrm{x1}} \cdot k \\ Tclk_{\mathrm{PPL}k} &\approx Tclk_{\mathrm{x1}} \end{aligned} \tag{7}$$

In the analysis actual parameters of all 4 derived architectures in 3 ciphers were compared to the estimates from the above equations and Fig. 2 illustrates the ratios *actual_value/ estimation*: the lower the ratio, the faster (shorter T_{clk}) or the smaller (lower number of slices) was the actual design in comparison to what could be expected from the x1 case. The value of 100 % is the threshold separating "better than" from "worse than" expected.

Speed parameters of all three ciphers on the Spartan-3 platform are close to expectations: in this aspect BLAKE behaves similarly to Salsa with negligible

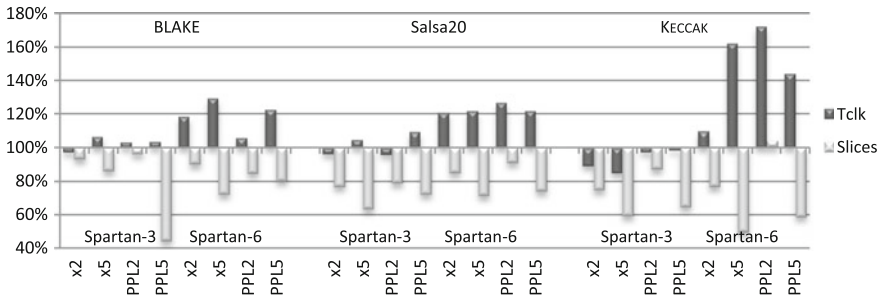


Fig. 2 Speed (minimum clock period) and size (slices) of the derived implementations compared to the estimations calculated form the respective basic x1 organization

variations –3 ÷ +7 %; in Spartan-6, on the contrary, actual clock period is noticeably worse than estimated, with x5 organization being nearly 30 % slower. This is a situation analogous to Salsa but not as bad as that in KECCAK, where routing congestion (affecting even the pipelined designs) could impair performance up to 60 ÷ 70 % as we have discussed in [7, 8].

Regarding number of occupied slices, it was already shown that large, unrolled designs present additional opportunities for optimizations in LUT usage [8] and size parameters can be significantly better than simple estimations in Eq. (6) predict. In Spartan-3 implementations of BLAKE this optimization gave actually the smallest effects across all tested cases (with an exception of PPL5 case, as it was noted in [9]) so unrolling of this cipher is not as effective as of Salsa and KECCAK with regard to this aspect.

3.4 *Efficiency of Implementation on Spartan-3 and Spartan-6 Platforms*

Figure 3 presents ratios of the speed and size metrics between the two platforms: parameter of Spartan-6 implementation was divided by the value for Spartan-3 and the quotient is displayed in percent.

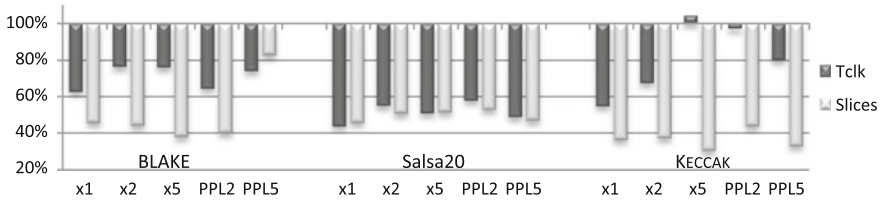


Fig. 3 Reductions in minimum clock period and number of occupied slices in Spartan-6 implementations as a percentage of Spartan-3 values

Using the device with more advanced architecture and manufactured in faster technology, Spartan-6 implementations are unquestionably expected to use significantly less slices and offer shorter clock periods, thus both the quotients should reach rather low values. This indeed is the case, although with one notable exception: realization of the x5 KECCAK architecture is actually slower in the new chip—an unusual and unexpected situation attributed again to routing congestion in Spartan-6 devices when the new SHA-3 standard is implemented [8].

In this picture BLAKE is located between the two other ciphers: reductions in its clock period are not as successful as those in Salsa (−25 ÷ −35 % vs. −45 ÷ −55 %) but no example of the above mentioned "negative progress" is observed. Regarding the size the order is reversed: BLAKE reductions (again with the exception of PPL5 case which was very efficiently implemented already in Spartan-3) are second to the best ones of KECCAK (−55 ÷ −62 % vs. −56 ÷ −67 %) and better than in Salsa (−47 ÷ −54 %).

4 Conclusions

Although BLAKE has not won the SHA-3 competition the interest in this cipher continues mainly due to its good software efficiency. In this work we evaluated its various high-speed hardware implementations including the case when the cipher loop is unrolled with up to 5 rounds instantiated in parallel. Preparing in total 5 different organizations and targeting them to two low-cost FPGA platforms we were able to analyze internal characteristic of BLAKE implementations generated automatically by the tools and to identify how the fundamental mechanisms of loop unrolling with or without pipelining work in case of this particular cipher.

The results confirmed close relationship to Salsa20 algorithm which has very similar internal organization but, at the same time, some disparities were identified: significantly larger size (due mainly to specific extension in internal data propagation which at first seems minor but is consequential in hardware realizations) and undecided (between the two ciphers) speed advantage which depends on targeted FPGA family. Additional comparison was made with previous results obtained for the same organizations of KECCAK algorithm so conclusions regarding loop unrolling efficiency and benefits brought by the newer Spartan-6 platform were generalized by including the case of this cipher.

References

1. Aumasson, J.P., Henzen, L., Meier, W., Phan, R.C.W.: Sha-3 proposal BLAKE, version 1.3. https://www.131002.net/blake/blake.pdf. Accessed March 2016
2. Bernstein, D.J.: The Salsa20 family of stream ciphers. In: New Stream Cipher Designs. Springer, Heidelberg (2008)

3. Gaj, K., Kaps J.P., Amirineni, V., Rogawski, M., Homsirikamol, E., Brewster, B.Y.: ATHENa—Automated Tool for Hardware EvaluatioN: Toward Fair and Comprehensive Benchmarking of Cryptographic Hardware Using FPGAs. In: 20th International Conference on Field Programmable Logic and Applications, Milano, Italy, (2010)
4. Gaj, K., Homsirikamol, E., Rogawski, M., Shahid, R., Sharif, M.U.: Comprehensive evaluation of high-speed and medium-speed implementations of five SHA-3 finalists using Xilinx and Altera FPGAs. In: The Third SHA-3 Candidate Conference. IACR Cryptology ePrint Archive, 2012, 368 (2012)
5. Junkg, B., Apfelbeck, J.: Area-efficient FPGA implementations of the SHA-3 finalists. In: 2011 International Conference on Reconfigurable Computing and FPGAs (ReConFig), pp. 235–241. IEEE (2011)
6. Sugier, J.: Low-cost hardware implementations of Salsa20 stream cipher in programmable devices. J. Polish Safety Reliab. Assoc. **4**(1), 121–128 (2013)
7. Sugier, J.: Low cost FPGA devices in high speed implementations of KECCAK-*f* hash algorithm. In: Zamojski, W., Mazurkiewicz, J., Sugier, J., Walkowiak, T., Kacprzyk, J. (eds.) Proceedings of the 9th International Conference on Dependability and Complex Systems DepCoS-RELCOMEX. AISC, vol. 286, pp. 433–441. Springer, Heidelberg (2014)
8. Sugier, J.: Popular FPGA device families in implementation of cryptographic algorithms. In: Zamojski, W., Mazurkiewicz, J., Sugier, J., Walkowiak, T., Kacprzyk, J. (eds.) Theory and Engineering of Complex Systems and Dependability. Proceedings of the 11th International Conference on Dependability and Complex Systems DepCoS-RELCOMEX. AISC, vol. 365, pp. 485–495. Springer, Heidelberg (2015)
9. Sugier, J.: Implementing SHA-3 candidate BLAKE algorithm in Field Programmable Gate Arrays. J. Polish Safety Reliab. Assoc. **7** (2016)
10. Xilinx, Inc.: Spartan-3 Family Data Sheet. DS099.PDF. www.xilinx.com. Accessed March 2016
11. Xilinx, Inc.: Spartan-6 Family Overview. DS160.PDF. www.xilinx.com. Accessed March 2016
12. Yan, J., Heys, H.M.: Hardware implementation of the Salsa20 and Phelix stream ciphers. In: Proceedings of the Canadian Conference on Electrical and Computer Engineering CCECE 2007, pp. 1125–1128. IEEE (2007)

Risk Analysis of Interference Railway GSM-R System in Polish Conditions

Marek Sumiła

Abstract The paper presents the problem of interference of GSM-R receivers as a result of coexistence with MFCN (public operators mobile networks). In the following sections presents necessary to understand the nature of the problem description of the GSM-R, causes and effects of interference and results of simulations considered interference phenomena. In the main part of the paper shows the results of evaluation a part of Polish railway line by the determining the distance of nearby MFCN's transmitters.

Keywords Railway safety · GSM-R · System interference

1 Introduction

GSM-R (GSM-Railway) is a wireless communication system based on the public GSM ETSI standard [9] but adopted to the needs and applications of nowadays railway transport companies. The standard is a one of primary subsystems of ERTMS (European Rail Traffic Management System) that allow to introduce interoperability in communication field between various European railway companies. For the users GSM-R is a secure platform for voice and data communication between railway operational staff, including drivers, dispatchers, shunting team members, train engineers, and station controllers.

In addition to that GSM-R is use as an important part in railway traffic signaling, control and train protection systems known as ETCS (European Train Control System). It is used to transmit data between trains and railway regulation centers with level 2 and 3 of ETCS and it is only way to announce permission to enter next track and new max. speed because at those levels of ETCS is not implemented traditional signs and semaphores near trackside. From the correct operation of the

M. Sumiła (✉)
Warsaw University of Technology, Koszykowa 75 str., 00662 Warsaw, Poland
e-mail: sumila@wt.pw.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_41

system depends, therefore, the safety of train traffic. In this way GSM-R is consistent with older, national systems of communication as was listed in [1, 12].

Since 2007 in Europe has been investigated problem of interference GSM-R systems by public MFCN (Mobile/Fixed Communications Networks). The experience of western Europe countries, where GSM-R has already been implemented, indicates that the public transmitters can cause interference and directly affect on the GSM-R terminals and it's receiver's circuits. The issues related to the coexistence of GSM-R and MFCN networks elaborate the UIC, the European Railway Agency (ERA), as well as a CEPT committee the ECC (Electronic Communications Committee). The UIC's working document [16] and test reports carried out by CEPT [2–4] shows the scale of the phenomenon and its consequences on the work of the GSM-R system and its terminals. Due to safety aspects the ECC Report [4] indicate two types of radio terminals susceptible to interference. There are Cab Radios and EDOR's terminals. The Cab Radio is use by a driver of a train and/or by other on-train system to voice communication. EDOR (ETCS Data Only Radio) is use by the ETCS train control application and transmit only data. Loss of communication by these types of terminals has a great impact on a operational and commercial consequences, such as: bad economic and commercial effects, impossible to make voice communication—especially Railway Emergency Calls (REC)—can lead to dangerous situations, jammed traffic—major delays, performances of line, etc. [14].

Polish implementation of GSM-R system is in progress and it will be connected with ETCS. The system is planned for the approximately 15,000 km of railway lines. The idea of the paper was formed as a result of previous author research and as no Polish Government regulations relating to the issue of interference of systems (GSM-R and MFCN). This article consider a real part of Polish railway line (E-20) and assess the risk of interference.

2 Causes and Classification of Interference

Public MFCN was designed and implemented a few years earlier than GSM-R. Public operators wants to provide continuous access to their network and covered the most area of the country included railway areas, too. For this reasons, some MFCN transmitters are located in places directly adjacent to the railway lines. In such condition, after implementation GSM-R system, in the railway area we can meet BTS of MFCNs. Another important premise for this claim is the fact that the GSM-R band is directly adjacent to the band of public GSM, UMTS and LTE networks.

In general, the classification of causes of radio interference in adjacent networks can be classified according to:

- quality of the devices used radio-broadcasting, including:
  - effectiveness of filtration spurious emission,

– antennas propagation characteristics of transmission,

- power of transmitters,
- geographic location of base stations interfering with the network,
- size of the separation between the interfering channels,
- levels of the interfering signal at the edge of cells (the quality of the radio coverage).

The direct adjacent band of those systems can produce out-of-band signals and lead to receivers blocking and cause the intermodulation products.

Definition of receiver blocking is a measure of the capability of the receiver to receive a modulated wanted input signal in the presence of an unwanted input signal. The frequencies of the unwanted signals are other than those of the spurious responses or the adjacent channels. The unwanted input signals causing a degradation of sensitivity of the receiver beyond a specified limit [6].

The intermodulation interference originates from interfering signals in neighboring channels due to the frequency selectivity of the antennas and the receiver equipment [6]. Those signals are from the coexistence MFCN and as proved in [13] do not have to be in closed adjacent channels. The receiver non-linearities produce an intermodulation product E_{if} of third order at the frequency. To consider a service with a desired signal at frequency f_0, a channel separation Δf and interfering signals E_{i1} and E_{i2} at frequencies $f_0 + n\Delta f$ and $f_0 + 2n\Delta f$, respectively.

$$f_0 = 2(f_0 + n\Delta f) - (f_0 + 2n\Delta f) \quad n = \pm 1, \pm 2, \ldots \tag{1}$$

The signal strength E_{if} of the intermodulation product is given by:

$$E_{if} = kE_{i1}^2 E_{i2} \tag{2}$$

with some constant k to be determined. For signal levels the Eq. (2) reads

$$L_{if} = 2L_{i1} + L_{i2} + 20 \log k \tag{3}$$

ETSI Standard [6] defines via the intermodulation response L_{imr}, the interfering signal levels $L_{i1} = L_{i2}$ at which bit errors due to intermodulation just start to be recorded. The constant $20 \log k$ in Eq. (3) comes from the measurement procedure is described in mentioned [6].

3 Methodology of Risk Analysis

In both types of interferences has been assumed that the minimum carrier-to-interference ratio, C/I, is too high at the receiver input. In order to calculate the C/I experienced by the receiver, it is necessary to establish statistics of both the wanted signal and unwanted signal levels.

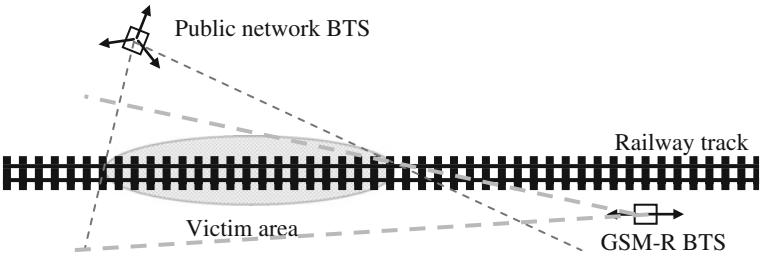


Fig. 1 Influence of public network BTS on railway track

The main parameter used to evaluate this ratio can be the distance between the MFCN transmitter and the railway line. According to this, it should be identified the BTS sites where the MCFN are in a short distance from the railway line as is was presented in Fig. 1.

This distance is different in different documents [4, 11, 17] but not exceed the ranges of 1000 m and is according to the difference between the signals, desired and unwanted, received by a GSM-R terminal. The distance is based on statistical analysis of reported cases of GSM-R networks' interference [4, 16] or the expected out-of-band signals in the railway area. In the case when we cannot precise wanted signal level we can take into account the minimum signal level e.g. −95 dBm for ETCS level 2 or −98 dBm for ETCS level 3. The minimum level of GSM-R signals are set out in the EIRENE SRS [5].

The signal strength in a distance from the transmitter can be calculated based on the modified Hata model available for outdoor-outdoor path loss calculations and equal

$$L = 69.55 + 26.16 \log(f) - 13.82 \log h_B - C_H + [44.9 - 6.55 \log h_B] \log d \,(\mathrm{dB}) \tag{4}$$

where:

L—median path loss (dB), f—frequency of transmission (MHz), h_B—base station antenna effective height (m), d—distance between the base and mobile stations (km), C_H—antenna height correction factor.

To simplify calculation it can be used the free-space attenuation formula when there are no obstacles in the signal path [10]. Then it can be used equation

$$L = 32.4 + 20 \log f + 20 \log d \,(\mathrm{dB}) \tag{5}$$

where:

L—level of signal attenuation in free-space (dBm), f—frequency (MHz), d—distance from the BTS to the point of a railway track (km).

Perform the risk assessment it should be considered and compare the signal from the MFCN transmitter and the level of wanted signal from the GSM-R base station.

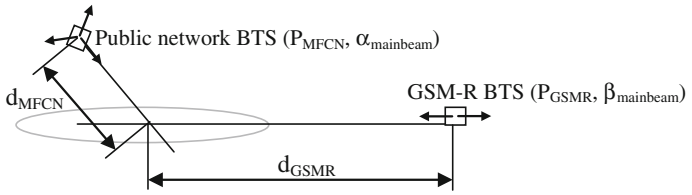


Fig. 2 Determination of the C/I at the point of the railway line

To accurately evaluate interfering signals from public networks it should consider such parameters as carrier types (2G, 3G, 4G) and a respective channel bandwidth, a frequency and an isotropic radiated power EIRP, a mounting height of an antenna, azimuth, a tilt, an antenna propagation characteristics.

The essence of the study was to evaluate the signal power comes from the MFCN transmitters and GSM-R sites in the rail track area. For this purpose it should be analyzed the transmitter's power (MFCN and GSM-R) and the attenuation on the distance between the transmitters and the point evaluated (Fig. 2).

Based on the figure we can therefore be concluded

$$P_{po} = P_{GSMR} - L(d_{GSMR}) - (P_{MFCN} - L(d_{MFCN})) \text{ (dBm)} \tag{6}$$

where
P_{po}—signal level at the point of railway, P_{GSMR}—GSM-R transmitter power [dBm], $L(d_{GSMR})$—attenuation at distance from GSM-R transmitter, P_{MFCN}—MFCN transmitter power (dBm), $L(d_{MFCN})$—attenuation at distance from MFCN transmitter.

This equation is valid under the assumption that the evaluated point is exactly in the axis of the main beam of the transmitted signal. If the evaluated point deviates from the main beam radiated signal power will be lower. It depends on the characteristics of the transmitter's antenna and is equal

$$P_{po} = k \cdot P_{GSMR} - L(d_{GSMR}) - (u \cdot P_{MFCN} - L(d_{MFCN})) \text{ (dBm)} \tag{7}$$

where
k, u—loss of power from main beam factor

When it is no knowledge about the exact location of the GSM-R transmitter it can assume the minimum value of the signal according to EIRENE SRS [5]. In case of implementation the ETCS level 2 we have

$$P_{po} = -95 - (u \cdot P_{MFCN} - L(d_{MFCN})) \text{ (dBm)} \tag{8}$$

4 Simulation of Harmful Interference

Previously mentioned phenomena causing interference GSM-R receivers can be evaluated in the simulation process. Assessment of the level of interfering signals at the point can be made with algebraic methods according to pre-specified methodology and allow to evaluate the probability of occurrence an interference in considered place. The use of previous formulas (7 or 8) allows to calculate the difference between useful and interfering signals.

The Fig. 3 shows the difference between signal's level in a function of distance from the transmitter MFCN to the point of assessment. For the accepted minimum signal level for ETCS level 2 (−95 dBm) in the distance of 1000 m, the level of difference is about 60 dB. To prevent the effect, the difference in signal levels cannot exceed 35 dB as it is concluded in the available reports [3, 4, 16]. That is a high probability that in assessed place we can meet with blocking of the input circuit of the GSM-R receiver.

Simulation of the intermodulation allow to assess where products appears. Results of such simulations are presented in Fig. 4 and shows that not only adjacent

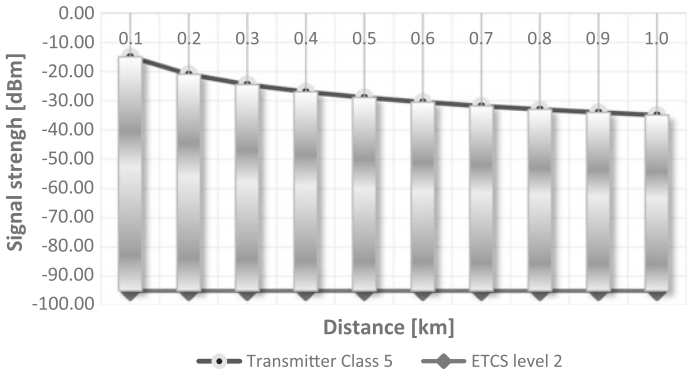


Fig. 3 The difference in signal levels in function of distance

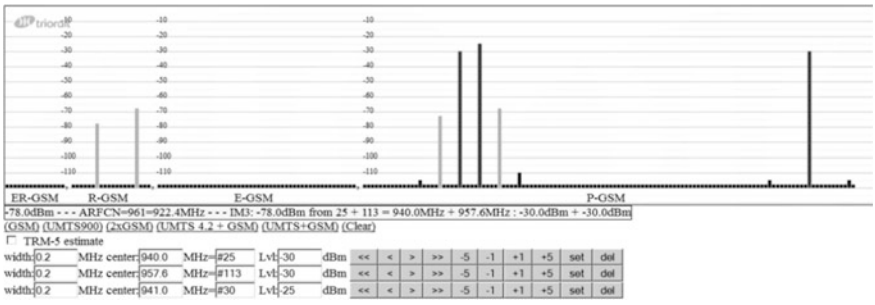


Fig. 4 The effect of intermodulation as a result presence of strong signals in the public GSM band. *Red bars* active channels of MFCN, *orange bars* intermodulation products

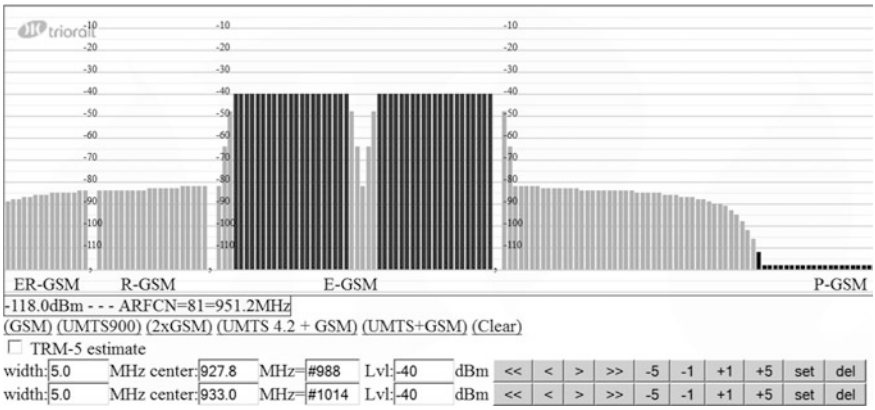


Fig. 5 Results of the simulation products of intermodulation (*orange bars*) from two broadband systems transmitting in the adjacent band

band signals influence on GSM-R band but is also function of far signals, too. Performed simulation has proofed that the intermodulation occurs for close and far signals.

For the wideband systems such as UMTS and LTE intermodulation products are a normal situation and the receiver generates it by mixing different spectral components corresponding to multiple GSM channels of 200 kHz. In practice, the vast majority of intermodulation cases are seen when the public BTS is closer than 250 m from the rail tracks (Fig. 5)

5 Risk Assessment of Influence of MFCN on Railway Line in Poland

In this section has been shown the results of evaluation the part of the line E-20 between stations Łuków and Biała Podlaska, where GSM-R system will be implemented. The selected part of the line E20 is shown in Fig. 6.

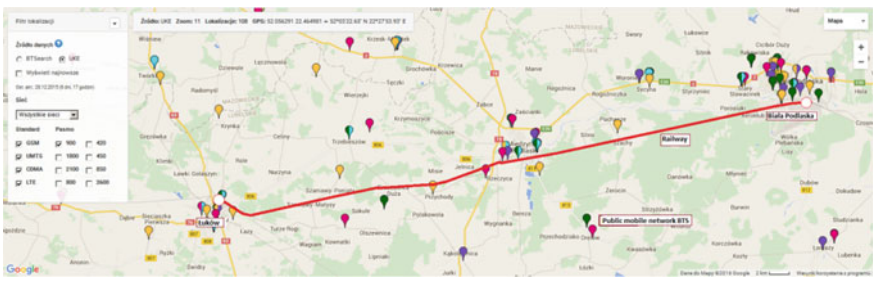


Fig. 6 Map of evaluated part of railway line (http://beta.btsearch.pl/)

Table 1 Results of analysis part of E-20 railway line

	Location	Occupied frequency band [MHz]					Dist. [m]
		925.1 – 930.1	930.1 – 935.1	942.3 – 948.5	948.5 – 953.1[1]	953.1 – 959.9[2]	
1	Łuków						170
2	Polskowola-Przychody						840
3	Rzeczyca						970
4	Międzyrzec Podl.						530
5	Międzyrzec Podl.						60
6	Międzyrzec Podl.						370
7	Szachy						560
8	Biała Podlaska						70
9	Biała Podlaska						60
10	Biała Podlaska						200
11	Biała Podlaska						570
12	Biała Podlaska						200

Based on the analysis of UKE data identified 22 MFCN transmitters located along the railway line in 12 locations. The distance between transmitter and railway line was obtained with using online tool http://beta.btsearch.pl/.

Results of analysis are shown in Table 1.

Based on the obtained data, it can be stated that the average distance to the railway track is about 383 m. The distance in seven out of twelve locations is less than this value. In six locations the MFCN transmitters broadcast technologies in adjacent bands and there are a wideband systems (Fig. 7)

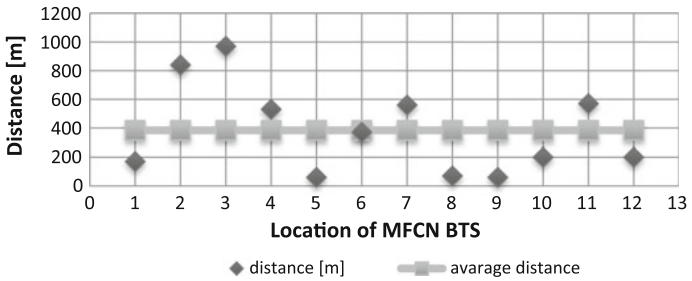


Fig. 7 Distance between MFCN BTS and the railway track according to average distance

To fully assess evaluated part of railway line it is necessary to obtain specification of MFCN BTSs and the information about locations of the GSM-R base stations.

6 Conclusion

GSM-R is a railway radio communication system and is an essential element to works of ETCS level 2 or 3. Unfortunately, the implementation of GSM-R takes place in time when public mobile networks (MFCN) are fully operate and its infrastructure exists. It should be emphasized that thea infrastructure was designed and built in an era when the ability of influence of GSM transmitters on the operation of the system GSM-R was not taken into account. The study of works [11, 16, 17] indicate that the newer radio GSM-R modems using improved filtration systems due to ETSI specification TS 102 933-1 [7] and TS 102 933-2 [8] at least version 1.2.1 are more resistant to the effects of blocking and intermodulation than the existing solutions. The use of new modems GSM-R allow to increase resistance by approx. 10 dB.

References to this article mentions a number of actions that could mitigate the impact of public networks for GSM-R. These are:

- filters in public operators' BTS,
- improved GSM-R coverage,
- fine tuning on engineering parameters,
- frequency band management,
- usage of ER-GSM band (restricted to some applications).

Finally, the solution to the problem of interfering with the network can be achieved only with effort on both sides.

References

1. Bester, L., Toruń, A.: Modeling of Reliability and Safety at Level Crossing Including in Polish Railway Conditions. Communications in Computer and information Science, vol. 471. Telematics—Support for Transport. Springer, Berlin (2014)
2. ECC Report 096: Compatibility between UMTS 900/1800 and systems operating in adjacent bands, Krakow (2007)
3. ECC Report 162: Practical mechanism to improve the compatibility between GSM-R and public mobile networks and guidance on practical coordination (2011)
4. ECC Report 229: Guidance for improving coexistence between GSM-R and MFCN in the 900 MHz band, CEPT (2015)
5. EIRENE System Requirements Specification: European Integrated Railway Radio Enhanced Network. GSM-R Operators Group. UIC CODE 951. Version 15.4.0, Paris (2015)
6. ETS 300 113-1: v1.6.1:2006: ETSI Standard. Electromagnetic compatibility and Radio spectrum Matters (ERM); Land mobile service; Radio equipment intended for the transmission

of data (and/or speech) using constant or non-constant envelope modulation and having an antenna connector; Part 1: Technical characteristics and methods of measurement (Question ITU-R 211/1)
7. ETSI TS 102 933-1 Railway Telecommunications: GSM-R improved receiver parameters. Part 1: Requirements for radio reception
8. ETSI TS 102 933-2: Railway Telecommunications (RT). GSM-R improved receiver parameters. Part 2: Radio conformance testing. V1.3.1 (2014-08)
9. ETSI TS 145 005 Digital cellular telecommunications system (Phase 2+); Radio transsmision and reception (3GPP TS 45.005 version 12.5.0 Release 12). Sophia Antipolis Cedex April 2015
10. ITU-R P.525-2: ITU-R Recommendation. Calculation of free-space attenuation (1994)
11. Radio Spectrum Committee: GSM-R interferences—contributions from delegations and ERA on issues. Statistics and best practices as a follow-up to the discussion in RSC#42. Working Document. European Commission. DG Communications Networks Content & Technology, Brussels, 26 February 2013
12. Rosiński, A.: Rationalisation of the maintenance process of transport telematics system comprising two types of periodic inspections. In: Selvaraj, H., Zydek, D., Chmaj, G. (eds.) Proceedings of the Twenty-Third International Conference on Systems Engineering, Given as the Monographic Publishing Series. Advances in Intelligent Systems and Computing, vol. 1089, pp. 663–668. Springer (2015)
13. Sumiła, M.: Wpływ interferencji pochodzących od publicznych sieci GSM na pracę odbiorników sieci GSM-R. Logistyka 4/2015. Poznań 2015, pp. 1005–1014. ISSN 1231-5478
14. Sumiła, M., Andrzej, M.: Analysis of the problem of interference of the public network operators to GSM-R. Tools of Transport Telematics/Mikulski Jerzy (red.), pp. 76–82. Springer, ISBN 978-3-319-24577-5 (2015)
15. Sumiła, M., Andrzej, M.: Evaluation of the impact of public networks to GSM-R receivers. In: Proceedings of 19th International Scientific Conference. Transport Means, 2015, pp. 223–226. Kaunas University of Technology, Lithuania. October 22–23, 2015, ISSN 1822-296X (print), ISSN 2351-7034 (online), Publishing House „Technologija“, Studentųst. 54, 51424, Kaunas (2015)
16. UIC O-8736-2.0. UIC Assessment report on GSM-R current and future radio environment; version of July 2014
17. UIC O-8740: Report on the UIC interference field test activities in UK. September 2013

Water Producers Risk Analysis Connected with Collective Water Supply System Functioning

Dawid Szpak and Barbara Tchórzewska-Cieślak

Abstract The paper presents threats to water producers related to the collective water supply systems (CWSS) functioning which is the background for safety assessment based on risk analysis. Risk is commonly used as a measure of systems safety. Risk analysis takes into account probability of adverse events occurrence, effects of these events and vulnerability to dangerous situations. The developed method is based on calculating an expected value of losses incurred by water company as a result of adverse events and calculating an expected value of water company financial profit. Its innovative character is to use incomplete or uncertain database. Water company profitability is one of the basic conditions for CWSS continued functioning. The developed methodology is presented on the application example for the county town in south-eastern Poland.

Keywords Collective water supply system · Risk analysis · Water producers

1 Introduction

The objective reality in the collective water supply system (CWSS) functioning is the possibility of the occurrence of various types of the undesirable events. These events do not appear without reason, they can be a result of a serious of events (emergency scenario), the so called "domino effect" [13]. The main threats to the CWSS safety are: errors associated with the fallibility of technology, natural disasters and errors associated with human activities [3, 4, 11, 14, 18]. They also occur

D. Szpak (✉) · B. Tchórzewska-Cieślak
Department of Water Supply and Sewage Systems, Rzeszow University of Technology, Al. Powstańców Warszawy 6, 35-959 Rzeszów, Poland
e-mail: dsz@prz.edu.pl

B. Tchórzewska-Cieślak
e-mail: cbarbara@prz.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*, Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_42

as a result of incorrect decisions, which cause the negative consequences during the CWSS functioning. For the right analysis of risk connected with the CWSS functioning an appropriate amount of different information, data recording and the possibility of their processing, which is not simple in practice, are necessary [2, 17].

Risk analysis connected with the CWSS functioning is often performed in the so called "uncertain information conditions", which is connected with uncertain (incomplete, imprecise or undependable) data concerning subsystem operating [6, 17]. The measure of data inaccuracy can be the so called quantitative uncertainty.

The paper contains the basic information concerning uncertainty in data analysis and the proposal of using the fuzzy logic theory to analyse water producers risk connected with CWSS functioning from economic point of view, if the uncertain data occur.

2 Risk in the CWSS Functioning

As a result of the occurrence in the CWSS the so called representative emergency scenario (RES), marked as S_i, water consumers and producers are subjected to the possibility of the loss of safety. Collective water supply system should have a high reliability level and have a low functioning costs [13].

According to the basic definition risk is understood as combination of frequency or probability of adverse events and consequences connected with it, or expected value of losses as a risky decision consequences [1, 5, 13]:

$$r = \{S, P, C\} \tag{1}$$

where:

S negative scenario described as a series of successive undesirable events,
P the probability of scenario S,
C losses caused by the scenario S.

This approach was subsequently extended to include a fourth parameter, called vulnerability to threats (V_i) or exhibition (E_i) [17].

Based on the review of the world literature [2, 8, 9, 14] and many years of authors' work on risk analysis and CWSS safety [16–18], the following safety measures for CWSS functioning are adapted: the risk from the water producer point of view r_p and the risk of water consumer r_c. Water consumer risk is associated with the possibility to consume poor quality water or the possibility of interruptions in water supply. The issue of water consumers risk is the subject of numerous works, among others [3, 12, 13], while the analysis of risk from the water producer point of view is not so widely recognized, therefore in this work we try to analyse this issue.

The measure of producers safety loss is risk connected with the possibility that the activity carried on will not be profitable. It was assumed that the risk (r) is a function of two parameters: an expected value of losses as a result of adverse events E(C) and an expected value of water company financial profit E(Z).

The formula to determine the size of water producers risk connected with the CWSS functioning is the following [12]:

$$r_p = \frac{E(C)}{E(Z)} \tag{2}$$

where:

E(C) an expected value of losses incurred by water company as a result of adverse events (losses associated with the lack of water sales, the need to repair and flushing water pipe network, possible compensation for water consumers) [15],

C(Z) an expected value of water company financial profit resulting directly from the water sale.

Expected value of losses incurred by water company as a result of adverse events is determined by the formula [15]:

$$\mathrm{E(C)} = \sum_{i=0}^{n} \mathrm{C_{Si}} \cdot \mathrm{P_{Si}}. \tag{3}$$

where:

C_{Si} the value of losses caused by the RES or a single adverse event, that may result in the water producers risk,

P_{Si} the probability of the RSA or a single adverse event, that may result in the water producers risk.

To determine a value of the probability that the given representative emergency scenario P_{Si} will occur, one needs to determine the probability that the particular events included in RES will occur, based on the statistical data and the assumed probability distribution. Calculation of the probability values P_{Si} can be made by means of the Event Tree Analysis, the Fault Tree Analysis or Bayesian networks [16].

Expected value of water company financial profit resulting directly from the water sale is determined by the formula:

$$E(Z) = \sum_{i=0}^{n} Z_{Si} \cdot P_{Si} \tag{4}$$

where:

Z_{Si} the value of water company financial profit resulting directly from the water sale during the RES or a single adverse event, that may result in the water producers risk

Table 1 shows a three scale of safety level (SL). It was assumed that risk, determined in accordance with formula (2), is a measure of water producers safety.

Table 1 Safety level

Safety level	r
Required safety level (RSL)	≤0.50
Acceptable safety level (ASL)	(0.50–1.0)
Unacceptable safety level (USL)	>1.0

3 Uncertainty in the CWSS Risk Analysis

Incidental undesirable events threatening CWSS safety are, among others [10, 12]:

- incidental pollution of water sources,
- natural disasters, e.g. drought, flood,
- interruptions in the supply of electricity,
- failure of the water supply network, for example, break of the water main,
- secondary water pollution in the water supply network,
- failures of pumping stations, water treatment plants (WTP),
- intentional action of the third parties, for example. vandalism, acts of terrorism.

The limitation of the effects of these undesirable events consists in [8, 12]:

- developing the emergency response plans, including the possibility to use alternative water sources,
- developing the alternative water treatment technology,
- developing a comprehensive system for water quality monitoring,
- current monitoring of the system through systematic inspections, renovations and modernization of the water distribution subsystem,
- increasing the emergency capacity of the network water tanks,
- efficient functioning of water supply and sewerage emergency service,
- comprehensive monitoring of operation of all the CWSS subsystems using SCADA and GIS,
- introduction to the strategy of the water company the risk of failure management for all the CWSS subsystems and based on it the implementation of safety procedures.

In the analysis of risk of the CWSS the most often the statistical uncertainty occurs, caused by the random nature of the analysed event, the impact of the external factors, as well as a time factor, which determines the change of the analysed undesirable event (failure).

There are some sources of uncertainty [16]:

- incomplete or imprecise definition of an analysed value (e.g. imprecise definition of failure in water pipe network),
- incomplete knowledge about the impact of environment on the analysed event (e.g. impact of the ground and water conditions on water pipe network failures),
- reading errors and accuracy class of instrument reading,
- inaccurate data obtained from the external sources (data about the CWSS operating obtained from waterworks),
- imperfection of the used research method.

To analyse uncertainty we usually use the probabilistic methods which require a large amount of data. In many cases data concerning the description of events, e.g. failures in water pipe network, are obtained on the basis of experts information (the CWSS users, experienced engineers or scientists). The most difficult is to chose the probability distribution. In practice, data concerning risk analysis in the CWSS are not only random but also unreliable (incomplete). Uncertainty of such data consists of many elements. Some of them are estimated on the basis of the assumed probability distribution, known from experience, or other information [17].

The following data, among others, are necessary to perform risk analysis in the CWSS [16]:

- data identifying analysed object (name and type of object and its basic technical data). Such data concern highly detailed studies,
- data about failures (undesirable events), repairs and other breaks in the CWSS operating (information about failure date, time, duration and its description),
- data concerning the reasons of the undesirable events occurrence,
- data concerning the consequences of those events.

The sources of the data necessary to analyse risk are:

- data collected from waterworks about the CWSS operating,
- measurement data (e.g. measurements of pressure and water flow in water pipe network, measurements of water leaks in water pipe network),
- data collected from experts.

The source of uncertainty in the mentioned above data analysis is, the most often, the incomplete or unreliable knowledge about:

- quantitative and qualitative data base concerning the CWSS failures,
- water pipe network technical condition assessment,
- imprecise and incomplete information concerning failures (the undesirable events) localization and identification, ground conditions, and so on,
- cause and effect failure assessment,
- experts opinions and expertises.

4 Utilization of the Fuzzy Sets Theory in the Water Producers Risk Analysis

The probabilistic methods for risk assessment (based on the undesirable events probability distributions) require unequivocal determination of statistical characteristics, which is connected with the necessity of having appropriate amount of highly reliable data. If the obtained data are “highly unreliable”, using such methods leads to getting faulty result of the performed analysis [16].

Having at disposal different types of data in the analysis of risk connected with CWSS functioning it is necessary to develop the method which would allow to use reliable (complete) data, as well as data which are unreliable or incomplete but are important from the risk assessment and analysis point of view. The method which can be useful to assess risk in such situation is the method using fuzzy risk analysis (FRA).

The notion of fuzzy sets was introduced in 1965 by Zadeh [19]. Unlike the conventional set, a limit of fuzzy set is not defined precisely, however, there is a gradual assessment from a complete lack of affiliation of an element in a set, through its partly affiliation, till its total affiliation. This gradual assessment is defined by means of the so called affiliation function μ_A, where A means a set of fuzzy numbers. Fuzzy sets can be used to describe different linguistic notions connected with risk analysis (little, medium, large, very large). A linguistic variable is such variable which characterises the so called fuzzy, imprecise notions, expressed by means of words, e.g. about number 1, high risk, low risk value [16].

The particular parameters, e.g. characterizing risk value are described by means of "n" linguistic variables. Then the particular linguistic assessments are assigned to fuzzy numbers x_j, which are defined as threes $x_j = (l_j, m_j, h_j)$, where: j = 1, 2, …n, with this condition satisfied:

$$0 \le l_j \le m_j \le h_j \le 1 \tag{2}$$

The fuzzy numbers values assigned to the particular linguistic variables are determined according to the following dependences:

- for j = 1

$$x_j = \left(0; 0; \frac{1}{n-1}\right) \tag{6}$$

- for $2 \le j \le n - 1$

$$x_j = \left(\frac{j-2}{n-1}; \frac{j-1}{n-1}; \frac{j}{n-1}\right) \tag{7}$$

- for j = n

$$x_j = \left(\frac{n-2}{n-1}; 1; 1\right) \tag{8}$$

where:

x_j a form of j fuzzy number,

n a number of linguistic variables describing given parameter (event, value, e.g. damage, loss, protection degree, probability, risk, safety),

j a successive number of linguistic variable, j = 1, 2, …n.

Linguistic type variables are often used to describe parameters characterizing size of risk, so there is the possibility to change linguistic variables into fuzzy numbers characterizing risk in the CWSS [14].

For the water producers risk r_p the following linguistic variables were assumed:

$$j = 1{-}\text{RSL}, \quad j = 2{-}\text{ASL}, \quad j = 3{-}\text{USL}.$$

For these assumed linguistic variables the fuzzy numbers values were determined, according to the formulas (6)–(8), which is presented in Table 2 [6].

In Fig. 1 a graphic interpretation of the affiliation function for water producers risk parameters connected with CWSS functioning is presented.

The transformation of a fuzzy set in the unfuzzy value can be performed using the method of singletons according to the equation [7, 17]:

$$r = \frac{\sum_{i=1}^{3} r_k \cdot \mu_A(r_k)}{\sum_{i=1}^{3} \mu_A(r_k)} \tag{9}$$

where:

$u_A(r_k)$ the value of membership function of a risk value to a fuzzy set A,

r_k risk value for every level, adopted as the middle value in the range.

Table 2 Fuzzy numbers for linguistic variables describing water producers risk r_p

Safety level	Fuzzy number
RSL	(0.0; 0.0; 0.5)
ASL	(0.0; 0.5; 1.0)
USL	(0.5; 1.0; 1.0)

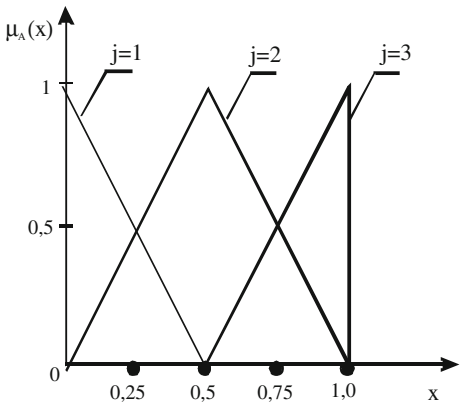


Fig. 1 The triangular affiliation function for producers risk r_p

5 Application Example

The developed methodology has been shown on the application example for the county-town in south-eastern Poland. The town area is 36.52 km^2, while the number of inhabitants is about 37 000. Average daily production of treated water in 2013 was 5630.78 m^3/d.

Table 3 shows the six RES related to the possible contaminated water consumption and the lack of water supply to consumers. Water company profit was 120819.16 € in 2014. Probabilities of the RES were estimated based on the failure book provided by the water company to an expert.

Table 3 Descriptive scale for RES [3]

No.	Description	P_i	C_i (€)	Z_i (%)
1	• The proper functioning of CWSS	0.65	0	100
2	• Slight deterioration of water organoleptic parameters that is not dangerous for consumers, • Local pressure reduction in water pipe network,	0.2	1 000	90
3	• Significant deterioration of water organoleptic parameters that is nuisance for consumers, • Daily water production decrease to 70 % Q_{maxd} or interruptions in water supply for 2 h, • Pressure reduction in water pipe network,	0.1	10 000	70
4	• Significant deterioration of water organoleptic parameters (high turbidity, smell), • Exceeded of water physicochemical parameters, • Single consumers poisoning associated with the contaminated water consumption, • Daily water production decrease to $0.5\ Q_{dmax} \leq Q_{dn} < 0.7\ Q_{dmax}$ or interruptions in water supply for 2–12 h, • Pressure reduction in water pipe network,	0.04	100 000	50
5	• Exceeded of bacteriological and physicochemical parameters of water quality, • Numerous consumers poisoning associated with the contaminated water consumption, • Daily water production decrease to $0.3\ Q_{dmax} \leq Q_{dn} < 0.5\ Q_{dmax}$ or interruptions in water supply for 12 to 24 h,	0.009	200 000	30
6	• Extensive water bacteriological contamination, presence of pathogenic microorganisms, exceeded of water physicochemical parameters, • Very widespread poisoning, necessity of hospitalization • Daily water production decrease to $Q_{dn} < 0.3\ Q_{dmax}$ or interruptions in water supply more than 24 h, • Water treatment plant, water pumping station or water main failure	0.001	1 000 000	0

Knowledge gained from the analysis may be an important source of information for understanding water value.

For the data presented in Table 3 water producers risk connected with the CWSS functioning was determined according to Eqs. (2)–(4):

$$\mathrm{E(C)} = 0.65 \cdot 0 + 0.2 \cdot 1000 + 0.1 \cdot 10\,000 + 0.04 \cdot 100\,000 + 0.009 \cdot 200\,000 + 0.001 \cdot 1000\,000 = 15200\,€$$

$$E(Z) = 0.65 \cdot 120819.16 + 0.2 \cdot 108737.24 + 0.1 \cdot 84573.41 + 0.04 \cdot 60409.58 + 0.009 \cdot 36245.75 + 0.001 \cdot 0 = 111479.84\,€$$

$$r_p = \frac{15200\,€}{111479.84\,€} = 0.136$$

Based on Fig. 1:

- r_p belongs to the fuzzy set j = 1 (RSL) with the membership function $\mu r_p(x) = 0.728$,
- r_p belongs to the fuzzy set j = 2 (ASL) with the membership function $\mu r_p(x) = 0.272$,
- r_p belongs to the fuzzy set j = 3 (USL) with the membership function $\mu r_p(x) = 0$.

The above values are interpreted as level with which risk value belongs to certain safety level. Based on the above, it is concluded that water producers safety level of analysed CWSS is RSL with the membership function $\mu r_p(x) = 0.728$ and ASL with the membership function $\mu r_p(x) = 0.272$.

The output value of the model, which is the basis for the assessment of water producer risk, calculated according to the formula (9) is:

$$r = \frac{0.728 \cdot 0.25 + 0.272 \cdot 0.5 + 0 \cdot 0.75}{0.728 + 0.272 + 0} = 0.318$$

CWSS condition is assessed as good. Despite this, water company should continuously improve CWSS functioning, which is a part of critical infrastructure, so that the risk remained at the same level or has been reduced to the lowest possible level.

6 Conclusions

- Analysis of water producers risk connected with the CWSS functioning should be one of the main element of complex CWSS risk management.
- For the correct and complete risk analysis and assessment it is necessary to possess large base of different data about subsystem operating. If we have access

to reliable and complete database, we should use the conventional methods of analysis and risk assessment (e.g. the matrix methods) [3, 11, 12, 20].

- If it is impossible to obtain accurate and complete data, then risk analysis is carried out in uncertainty conditions. A solution for this problem can be to apply risk analysis methods using fuzzy set theory (FRA).
- The risk can be defined as a fuzzy value, especially when it is estimated and not defined, which often takes place when the analysis of the undesirable events in the CWSS is performed.
- The present paper is a proposal of the utilization of known methods of fuzzy logic theory in analysis and assessment of water producers risk in which base of operating data is small, incomplete or with a low degree of reliability.
- The proposed method can be applied to assess water producers safety level in terms of producers profitability. The condition of method applicability is to have information about adverse events in CWSS and financial data about water companies activities.
- The presented method is the first stage of the study on the risk of water producers. Further works on the improvement of the method should take into account not only economic point of view but also the indicators of reliability.

References

1. Aven, T.: Conceptual framework for risk assessment and risk management. Summer safety and reliability seminars. J. Pol. Saf. Reliab. Assoc. **1**, 15–27 (2010)
2. Boryczko, K., Piegdoń, I., Eid, M.: Collective water supply systems risk analysis model by means of RENO software. In: Steenbergen et al. (eds.) Safety, Reliability and Risk Analysis: Beyond the Horizon, pp. 1987–1992. Taylor & Francis Group, London (2014)
3. Boryczko, K., Tchórzewska-Cieślak, K.: Analysis of risk of failure in water main pipe network and of delivering poor quality water. Environ. Prot. Eng. **40**(4), 77–92 (2014)
4. Fortmann-Roe, S., Iwanejko, R., Wójcik, W.: Dynamic risk assessment method–a proposal for assessing risk in water supply system. Archives Environ. Prot. **41**(2), 36–40 (2015)
5. Kaplan, S., Garrick, B.J.: On the quantitative definition of risk. Risk Anal. **1**(1), 11–27 (1981)
6. Kleiner, Y., Rajani, B., Sadiq, R.: Failure risk management of buried infrastructure using fuzzy-based techniques. Aqua—J. Water Supply: Res. Technol. **55**(2), 81–94 (2006)
7. Kluska, J.: Analytical Methods in Fuzzy Modelling and Control. Springer, Berlin (2009)
8. Mays, L.W.: Water Supply Systems Security. McGraw-Hill Professional Engineering, Texas (2004)
9. McGill, W.L., Ayyub, B.M., Kaminskiy, M.: Risk analysis for critical asset protection. Risk Anal. (Wiley Blackwell) **27**(5), 1265–1281 (2007)
10. Parafińska, K., Marcinkowski, J.T., Zimoch, I.: Hazardous events and hazard identification in water supply systems. Probl. Hig. Epidemiol. **96**(2), 349–356 (2015)
11. Pietrucha-Urbanik, K.: Failure analysis and assessment on the exemplary water supply network. Eng. Failure Anal. **57**, 137–142 (2015)
12. Rak, J.: Risk estimates for water supply systems. Ochrona Środowiska **25**(2), 33–36 (2003)
13. Rak, J.R.: Some aspects of risk management in waterworks. Ochrona Środowiska **29**(4), 61–64 (2007)

14. Sadiq, R., Kleiner, Y., Rajani, B.: Water quality failures in distribution networks—risk analysis using fuzzy logic and evidential reasoning. Risk Anal. **27**(5), 1381–1394 (2007)
15. Studziński, A.: Amount of labour of water conduit repair. In: Steenbergen, et al. (eds.) Safety, Reliability and Risk Analysis: Beyond The Horizon, pp. 2081–2084. Taylor & Francis Group, London (2014)
16. Tchórzewska-Cieślak, B.: Uncertainty in analysis of risk connected with water distribution subsystem functioning. XX Jubileuszowa Krajowa Konferencja, VIII Międzynarodowa Konferencja „Zaopatrzenie w wodę, jakość i ochrona wód", vol. 2, pp. 119–130, Gniezno (2008)
17. Tchórzewska-Cieślak, B.: A fuzzy model for failure risk in water-pipe networks analysis. Ochrona Środowiska **33**(1), 35–40 (2011)
18. Tchórzewska-Cieślak, B., Szpak, D.: A proposal of a method for water supply safety analysis and assessment. Ochrona Środowiska **37**(3), 43–47 (2015)
19. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**(3), 338–353 (1965)
20. Zio, E.: Computational Methods for Reliability and Risk Analysis. Springer, Berlin (2009)

Analysis of Reshuffling Cost at a Container Terminal

Justyna Świeboda and Mateusz Zając

Abstract Intermodal transport is a popular mode of freight transportation over long distances. Apart of reliability the economic efficiency becomes crucial issue of the intermodal operation. Economic efficiency and improve the efficiency of the transshipment operations is the challenge that face intermodal yard planners. The purpose of this article is an analysis of the cost of reshuffling and the proposal for its limitations. The paper includes analysis of moves productivity and cost estimation. There is propose of heuristic method in the paper, which helps to reduce cost of unproductive moves.

Keywords Inland terminal · Unproductive moves · Handling operations

1 Introduction

Intermodal transport [4] may be defined as passenger or freight transport from the initial point to the destination, using at least two means of transport. Such transport can be carried out with a various configuration of the means of transport, i.e. by truck, rail and ocean shipping. Intermodal transport refers to a multimodal chain of freight service; most frequently this is an integrated freight unit, namely a container. Such transport is carried out over long routes.

Functions of the container terminal include handling and storage of units. Figure 1 shows a diagram with a visual representation of possible operations. Loading operations in the inland terminal can occur with the following relations:

J. Świeboda (✉) · M. Zając
Wroclaw University of Technology, Wyb. Wyspianskiego 27,
50-370 Wroclaw, Poland
e-mail: justyna.swieboda@pwr.edu.pl

M. Zając
e-mail: mateusz.zajac@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_43

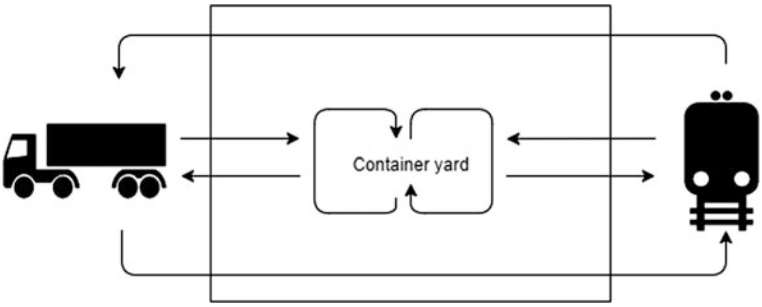


Fig. 1 Handling relations in the inland container terminal

- Road transport—rail transport or rail transport—road transport,
- Road transport—container yard or container yard—road transport,
- Rail transport—container yard or container yard—rail transport,
- Relocations of freight within the yard.

One of the main barriers for an effective operation at handling points is the lack of method of scheduling unloading operation. As a result, the operation time is extended and carrying out unnecessary operations within the yard generates additional and unnecessary costs.

2 Literature Review

The intermodal transport riches development permanently. Many authors have discussed the problems of scheduling tasks and container reshuffling at handling points. For instance, in the paper [18] the Authors develops a model in which strategies for storage and container handling scheduling. In the model, container arrangement time depends on the scheduling sequence and the level on which the given container is located. In paper [21], the authors develop a model consisting of a storage subsystem model and a yard crane scheduling subsystem model. In the paper by [8], the aspect of scheduling performed for three items of handling equipment available in ports—quay crane, yard crane and internal trucks—is discussed. This approach allow to keep balance between savings on time and energy. The optimisation algorithm integrates genetic algorithm (GA) and particle swarm optimisation (PSO) algorithm, where the GA is used for global search, and the PSO is used for local search. Paper [20] deals with a heuristic method referred to as the multi-layer genetic algorithm (MLGA) used for obtaining an optimum solution to the problem of scheduling at an automated container terminal. The algorithm involves three types of handling equipment, that is, quay cranes (QCs), automated guided vehicles (AGVs) and automated yard cranes (AYCs). In the paper by [11], a model relating to sea terminals and two items of handling equipment—quay cranes

and yard trucks—is developed. Compared to the mentioned method of operation scheduling for both types of the equipment, the integrated model takes into account the sequence between containers, blocking, quay crane interference and quay crane safety margin. The literature provides many papers devoted to the concept of crane operation scheduling, with a specific focus on minimisation of handling times. The paper by [7] develops a method whereby the energy-saving aspect was also taken into account. In the case of a similar system, that is a warehouse system, an optimisation model was developed, whereby the concept of energy savings [42] was taken into account, whereas the paper [36] involves an analysis of costs in production systems. Nevertheless, the genetic algorithm in the concept of transport management was presented in the paper [37, 38]. The paper [6] presented a model that optimises the processing operation in a waste disposal enterprise. This model optimises the operation time in addition to increasing the output of the entire system. In papers [10, 19], the authors developed an application for logistic system management, wherein a cost analysis was also included. In the paper [39, 40], a method of collecting containers sequence from the train was developed.

The author of the paper [2] discussed unproductive moves in container terminals. Unproductive movements of containers are called shifts, which are both time and money consuming. In the paper [1], an approach was presented for the arrangement of containers on board a ship so that effectiveness can be improved, with the number of unproductive movements eliminated. Another paper [5] presented a heuristic method for the reduction in the number of unproductive moves while loading the container ship. In the paper [3], the discussion relates to the problem of selecting the best location for the container within the container yard. The model developed in this paper—a mixed integer programming model—improves the effectiveness in the use of the container yard space, as well as the effectiveness in the relocation of the freight from the container yard to the berth, and it minimises the re-handling of the operation within the station (minimisation of unproductive movements). The model results in an improvement of output of the entire sea terminal. The elimination of unproductive moves was also presented in [12, 13]. This paper presented a dynamic programming model which minimises the number of handling tasks. The re-handling is also related to the storage space utilisation on the yard. The paper [17] presented questions of reliability and increase in efficiency in an inland terminal.

Optimisation, scheduling and effectiveness improvement are analysed not only in intermodal transport, but also through other transport systems and sectors. The concept of operation and maintenance is presented in the papers [23, 31, 34]. The papers [9, 22, 35] presented precise reliability analyses as well as models enabling optimisation of profits in the studied technical systems. Reliability models for optimisation of tasks in rail transport were presented in the papers by [25, 26], and in transport by air, in the papers [14–16, 27–29]. As far as the problem of failure in various technical systems is concerned, the papers by [30, 32, 33] presented modelling of ship movements in various failure situations. An essential element in the system, whereby handling operations are carried out, is also given as reliability of the information system as presented in the paper [24].

A considerable number of papers [12, 13, 18] are devoted to re-handling, storage, optimisation, effectiveness improvement and reliability in seaports. The authors focused their attention on the basic handling equipment, i.e. quay crane or yard crane. In contrast to the listed studies, this paper includes a productivity analysis of operations in an inland terminal, taking into account the indicator of costs and losses incurred by the handling point being investigated.

3 Study and Analysis of Handling Operations at the Container Terminal

The objective of the study is to analyse fuel costs associated with handling equipment during handling operations. The analysis takes into account operating times of individual machines. The study was divided into two stages. During the first stage, operating times of individual handling equipment items in the terminal were measured. The second stage consisted in processing the data representing fuel consumption as well as operating times of the engine during handling operations and idle periods (engine standing idle).

3.1 Analysis of Handling Operations at the Terminal

The study on the inland terminal was conducted between January and July 2015. All handling operations performed with reach stackers were taken into account. Taking into account the number of car (41 %) and rail operations (27 %), results from unloading and loading in relation to the means of transport. However, it can be said that there are definitely too many relocation operations (32 %). These movements generate losses associated with accelerated wear of parts in handling vehicles as well as an increased fuel consumption. For a precise comparison of handling operations in the given terminal, the Table 1 was compiled to provide statistical data.

As handling operations are described in Table 1, all results are rounded to whole numbers. The analysis includes all handling operations in individual weeks (from Week 3 to Week 27 in 2015). The highest number of rail operations occurred in Week 26, with an average number ranging from 12 to 28 operations. Other indicators show that in most weeks the number of operations varies significantly. The coefficient of variation amounted to less than 100 % only in two weeks (13 and 22), however, this value is still very high. Taking into account operations performed with regard to handling road vehicles, the highest number was encountered in Week 13 (812 handling operations), with the lowest number—238 operations—recorded in Week 4.

Table 1 Statistical data for handling operations

Week no.	Rail operations				Car operations				Relocations			
	$\sum_{i=1}^{m} x_i$	$\bar{x}$	σ	V (%)	$\sum_{i=1}^{m} x_i$	$\bar{x}$	σ	V (%)	$\sum_{i=1}^{m} x_i$	$\bar{x}$	σ	V (%)
3	245	16	18	113	419	28	21	75	250	17	14	83
4	238	13	22	167	238	30	25	83	238	16	13	85
5	246	12	21	171	542	25	25	100	271	14	14	101
6	427	27	35	133	599	37	24	65	250	16	9	57
7	370	16	26	164	800	35	37	107	529	23	19	83
8	295	20	33	168	518	35	22	63	256	17	13	74
9	349	19	27	138	607	34	28	84	327	18	17	95
10	367	23	30	130	406	25	17	68	385	24	11	47
11	371	20	27	139	381	20	20	98	320	17	15	91
12	389	28	28	102	542	39	22	56	472	34	17	49
13	415	26	24	94	812	32	32	102	669	42	18	43
14	470	28	32	116	505	30	23	79	461	27	17	64
15	222	19	22	120	362	30	18	60	213	18	13	76
16	358	19	24	127	604	32	28	88	418	22	15	68
17	357	21	23	112	608	36	26	74	494	29	13	46
18	355	22	24	107	431	27	23	85	429	27	16	58
19	336	19	23	121	592	33	25	75	529	29	24	83
20	391	23	27	119	672	40	28	70	547	32	22	68
21	376	21	25	122	586	33	26	79	520	29	20	71
22	375	25	25	98	530	35	21	58	642	43	24	57
23	372	25	27	107	457	32	23	73	473	32	18	58
24	337	20	24	120	551	32	26	80	654	38	26	66
25	353	14	21	150	644	27	27	102	511	21	17	78
26	492	20	26	133	491	20	23	115	414	17	14	86
27	302	18	25	142	372	22	18	84	189	11	11	97

Signs: i week number, m number of weeks

This amounts to approximately 30 road vehicle operations per week. Nonetheless, in this case, the standard deviation value suggests that the results should be similar to the average value. The coefficient of variation exceeding 100 % was attained only in 5 weeks. While analysing the results obtained for relocation operations, the highest number occurred in Week 13, with the average weekly number of operations being approximately 25. There is only one instance when the coefficient of variation has exceeded 100 %. In some cases, the values amounted to less than 50 %, which means that a similar number of handling operations within the yard were completed during the period subject to study.

4 Analysis of Container Terminal Productivity

There are two modes of operation to be distinguished for the handling equipment: productive operation in the course of handling operations, whereby all systems are activated; and unproductive operation, whereby the equipment is on and idle and no operations are being carried out (engine standing idle). The value of productive operation is almost equal to 80 %, while in the case of unproductive operation time, the value is 21 %.

4.1 Fuel Consumption Costs

The second stage of the study consisted in processing the data necessary to calculate indicators relating to fuel consumption G, and costs C of operation of the handling equipment. One of the assumptions relates to fuel price p = 4.29 PLN. The following section presents the method of calculations and the results obtained for two items of handling equipment.

(1) Equipment DRF RS1:

$$G = \frac{e_c}{t_c}. \tag{1}$$

$$G = \frac{14182}{965} = 14.7\,\frac{l}{mth},$$

$$C = G \times p, \tag{2}$$

$$C = 14.7 \times 4.29 = 63.10\,\text{PLN},$$

where,
e_c total quantity of consumed fuel [I],
t_c total operation time of the equipment [mth],
G fuel consumption,
p unit price per litre of fuel,
C cost of mth

Fuel consumption per time unit (mth) amounts to 14.7 l, while the fuel cost for that unit is PLN 63.10.

Taking into account the fuel costs, they vary considerably, depending on equipment operating time as well as the number of handling operations. The highest fuel costs in the container terminal were incurred in week 24, which amounted to PLN 6 057.6. In most cases, the fuel costs exceeded PLN 2 000. For unproductive

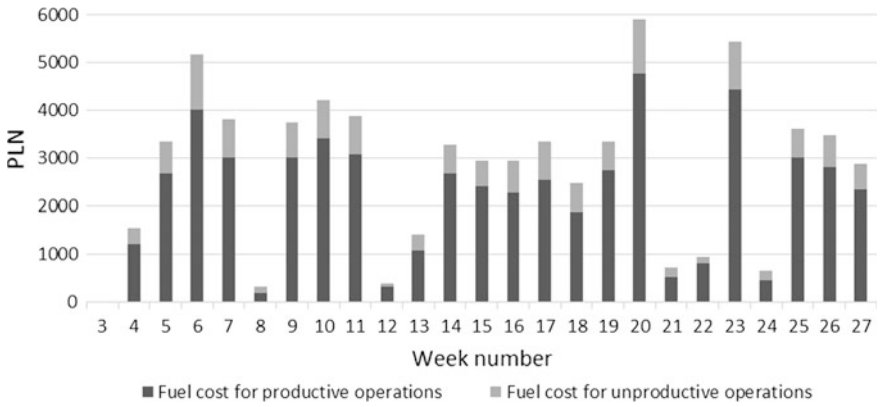


Fig. 2 Fuel consumption costs in individual weeks, consumed by the equipment DRF RS2

operations, the costs are not significant; however, if unproductive operation is limited, the terminal can reduce the costs and increase the profit.

(2) Equipment DRF RS2:

Based on formulae (1) and (2), it is possible to calculate indicators for the equipment:

$$G = \frac{16308}{1043} = 15.64 \frac{l}{mth},$$

$$p = 15.64 \times 4.29 = 67.10\,\text{PLN}$$

While comparing the results of calculations with the other equipment, the costs are slightly higher than for the first equipment. This situation results from minimally more frequent operation of the machine and slightly more handling operations completed.

The chart in Fig. 2 concludes that the highest costs associated with equipment operation were generated in Week 20, i.e. PLN 5 904. 8. For the studied period, the costs exceeded PLN 3 000 in 13 weeks.

5 Suggested Method

The solution [41] to the problem is based on three steps, which can be recorded as an ordered three of $\langle S, V, D \rangle$, where: S is task performance status, V is task performance value, and D is task performance order.

(1) Task performance status.

In the discussed approach, after reports are entered in the order fulfilment system, a check is first performed to see whether a task from the list can be performed or whether another task with another status value must be performed first. For the purposes of the model, it has been decided that tasks with a lower status value have a lower rank. It has been assumed that the value of the parameter can fall within the following range: $S = \{0, 1, 2, \ldots, n\}$.

As regards the list of containers that must be unloaded from the carriage to the container yard, the performance of this part of the method is the verification of the expected storage time of arriving containers and containers already stored. The point is that containers unloaded from the carriage may not block those which will be used for subsequent transport sooner. Therefore, the statuses of containers on the carriages and on the yard are compared as well as statuses of containers for unloading. If the i-th container is intended for earlier release than the j-th container, then $S_i > S_j$.

If containers are to be released at the same time, their status values can be the same.

(2) Task performance value:

Another parameter of the presented approach is the task performance value V. In its general meaning, the "value" can pertain to the observed value selected by the decision-maker. In practice, with the operation time schedule, attention is usually paid to the time of a single task performance, its cost and distance, usually in search of the minimum value of the quantities listed.

In the discussed case of unloading operations at an intermodal container terminal, the order fulfilment time was adopted as the time of order fulfilment, i.e. moving container from the carriage to the yard.

Attention is paid to the fact that values of performing the same task are different for a classical gantry crane, different for a reach stacker; they also differ if the container is placed on the first layer as compared to being placed higher.

(3) Task performance order:

Establishing the value of task performance does not answer the question related to the order in which they should be performed. As a result, the discussed approach to the performance of a cycle of orders is complemented by the value of the sequence of task performance marked as D where $D = \{0, 1, 2, \ldots, n\}$.

The time (or another "value") needed for conversion (or preparation) required to perform the next task must be determined. An analysis of the required "preparation" times will make it possible to define the order of task performance when the adopted value will be the lowest (or the highest in the case of looking for solutions with the maximum value). The time needed to get from the container placement location to the next containers to be unloaded will be defined. In this way, a tree of task performance variants can be prepared, where the node is the trip of the container from the carriage to the storage place, the branches, on the other hand, defines

values of the trip time to the places, from which subsequent loading units are taken. The choice of path with the lowest sum of values will facilitate the procurement of the sequence of nodes travelled, i.e. the sequence of task performance.

6 Analysis of Method Usefulness

In order to verify the usefulness and applicability of the method, as many as 4 variants are suggested (see Table 2), so as to compare fuel consumption costs before and after the method is employed.

The variants determine possible fuel cost savings once the method has been applied. Variant V is an output variant for the actual object prior to method application. The chart presenting fuel cost savings for the first equipment is presented in Fig. 3.

Table 2 Possible variants of fuel cost reduction

Variant no.	Variant I (%)	Variant II (%)	Variant III (%)	Variant IV (%)	Variant V (%)
Productive operation	−20	−30	−20	−30	100
Unproductive operation	−20	−20	−30	−30	100

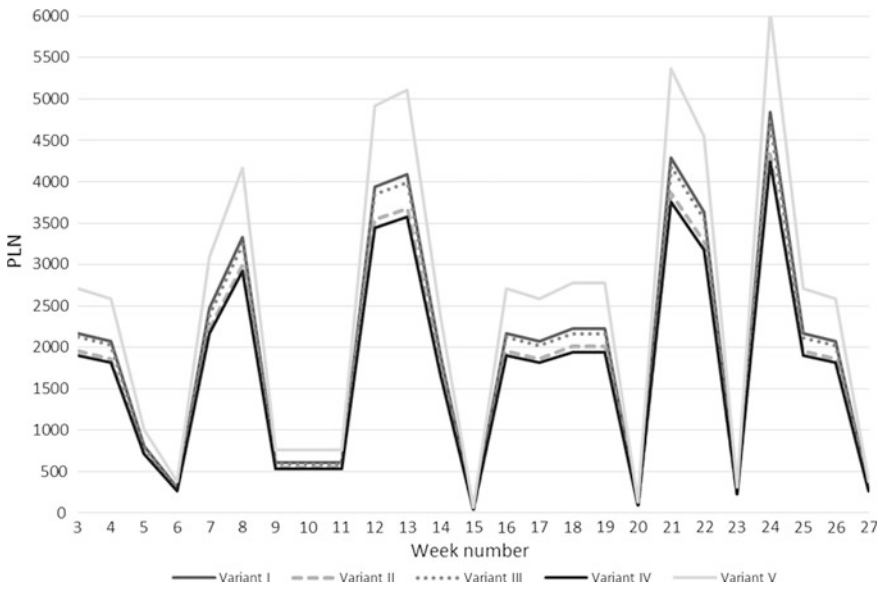


Fig. 3 Chart presenting fuel cost savings after application of various variants to DRF RS1

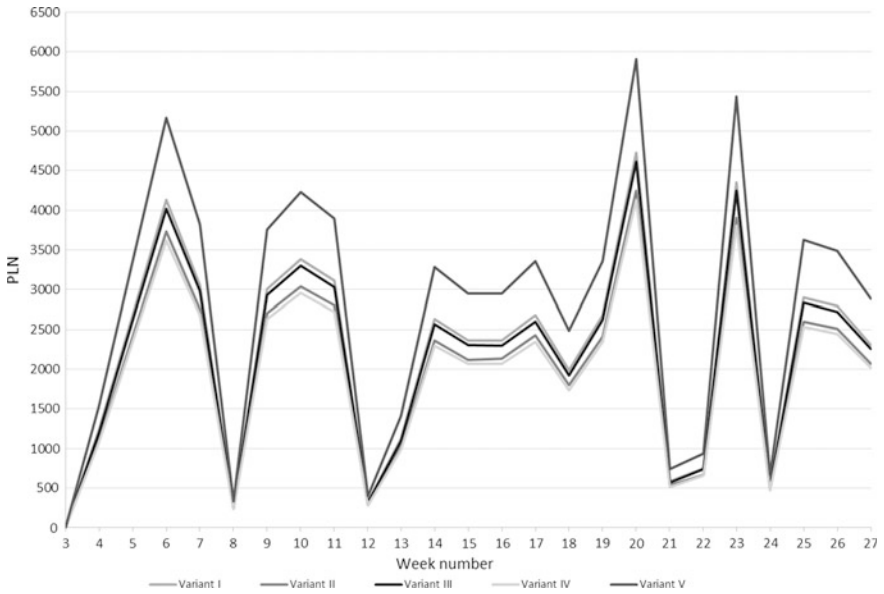


Fig. 4 Chart presenting fuel cost savings after application of various variants to DRF RS2

For the first equipment, Variant IV provides the best results. In this case the savings amount to PLN 1 500, whereas Variant I results in the lowest savings.

By analysing the chart in Fig. 4, similar to the case of the first equipment, Variant IV provides the highest savings, whereas Variant I results in the lowest values. Here the savings amount to PLN 1 700.

7 Summary

Ground terminals are an important element of the intermodal transport chain. From the point of view of the intermodal transport operators the main problem is an economic efficiency and the reliability of the performed operations in terminals. Unproductive move limitation is a difficult task that has not yet been solved. The solution discussed in this article focuses on simple heuristics closed to three steeps: assigning majority of operations, its values, and preparing proper order of them. As a result of the implementation of SVD schema can be obtained reduction of fuel consumption of up to 40 %, at the same time increasing the operational potential of cargo handling machinery.

References

1. Ambrosino, D., Sciomachen, A., Tanfani, E.: Stowing a containership: the master bay plan problem. Transp. Res. Part A **38**, 81–99 (2004)
2. Chen, T.: Yard operations in the container terminal—a study in the 'unproductive moves'. Marit. Policy Manag. **26**(1), 27–38 (1999)
3. Chen, L., Lu, Z.: The storage location assignment problem for outbound containers in a maritime terminal. Int. J. Prod. Econ. **135**, 73–80 (2012)
4. Crainic, T.G., Kim, K.H.: Intermodal transportation. Transportation **14**, 467–537 (2006)
5. Ding, D., Chou, M.C.: Stowage planning for container ships: a heuristic algorithm to reduce the number of shifts. Eur. J. Oper. Res. (2015)
6. Giel, R., Młyńczak, M., Plewa, M.: Logistic support model for the sorting process of selectively collected municipal waste. In: Theory and Engineering of Complex Systems and Dependability, pp. 369–380. Springer International Publishing (2015)
7. He, J., Huang, Y., Yan, W., Wang, S.: Integrated internal truck, yard crane and quay crane scheduling in a container terminal considering energy consumption. Expert Syst. Appl. **42**(5), 2464–2487 (2015)
8. He, J., Huang, Y., Yan, W.: Yard crane scheduling in a container terminal for the trade-off between efficiency and energy consumption. Adv. Eng. Inform. **29**, 59–75 (2015)
9. Hinz, M., Temminghoff, P., Bracke, S.: APTA approach: analysis of accelerated prototype test data based on small data volumes within a car door system case study. In: RAMS: The 12th Probabilistic Safety Assessment and Management, PSAM 12 (2014)
10. Jodejko-Pietruczuk, A., Nowakowski, T., Werbiska-Wojciechowska, S.: Issues of multi-stated logistic support system performing in a system of systems. In: Proceedings of CLC 2013: Carpathian Logistics Congress, TANGER Ltd., pp. 659–664 (2014)
11. Kaveshgar, N., Huynh, N.: Integrated quay crane and yard truck scheduling for unloading inbound containers. Int. J. Prod. Econ. **159**, 168–177 (2015)
12. Kim, K.H.: Evaluation of the number of rehandles in container yards. Comput. Ind. Eng. **32**(4), 701–711 (1997)
13. Kim, K.H., Park, Y.M., Ryu, K.R.: Deriving decision rules to locate export containers in container yards. Eur. J. Oper. Res. **124**(1), 89–101 (2000)
14. Kierzkowski, A., Kisiel, T.: The simulation model of logistic support for functioning ground handling agent, taking into account the probabilistic time of aircrafts arrival. In: Proceedings of CLC 2013: Carpathian Logistics Congress, TANGER Ltd., pp. 463–469 (2014)
15. Kierzkowski, A., Kisiel, T.: Functional readiness of the check-in desk system at an airport. In: Theory and Engineering of Complex Systems and Dependability. Springer International Publishing, pp. 223–233 (2015)
16. Kierzkowski, A., Kisiel, T.: Functional readiness of the security control system at an airport with single-report streams. In: Theory and Engineering of Complex Systems and Dependability, pp. 211–221. Springer International Publishing (2015)
17. Kierzkowski, A., Zajac, M.: Analysis of the reliability discrepancy in container transshipment. In: 11th International Probabilistic Safety Assessment and Management Conference and the Annual European Safety and Reliability Conference, pp. 826–831 (2012)
18. Kozan, E., Preston, P.: Genetic algorithms to schedule container transfers at multimodal terminals. Int. Trans. Oper. Res. **6**(3), 311–329 (1999)
19. Kisiel, T., Valis, D., Zak, L.: Application of regression function—two areas for technical system operation assessment. In: Proceedings of CLC 2013: Carpathian Logistics Congress, TANGER Ltd., pp. 500–505 (2014)
20. Lau, H.Y., Zhao, Y.: Integrated scheduling of handling equipment at automated container terminals. Int. J. Prod. Econ. **112**(2), 665–682 (2008)
21. Liang, C., Gu, T., Lu, B., Ding, Y.: Genetic mechanism-based coupling algorithm for solving coordinated scheduling problems of yard systems in container terminals. Comput. Indus. Eng. (2015)

22. Lücker, A., Hinz, M., Knübel, G., Bracke, S.: Reliability analysis of organic fibres based on small data volumes. In: RAMS 2015—61th Reliability and Maintainability Symposium, Palm Harbor, Florida, U.S.A. (2015)
23. Nowakowski, T., Tubis, A., Werbińska-Wojciechowska, S.: Maintenance decision making process–a case study of passenger transportation company. In: Theory and Engineering of Complex Systems and Dependability, pp. 305–318. Springer International Publishing (2015)
24. Polak, R., Laskowski, D.: Reliability of routing protocols. J. KONBiN **35**(1), 51–62 (2015)
25. Restel, F.J.: The Markov reliability and safety model of the railway transportation system. In: Safety and Reliability: Methodology and Applications—Proceedings of the European Safety and Reliability Conference, ESREL 2014 (2015)
26. Restel, F.J.: Defining states in reliability and safety modelling. In: Theory and Engineering of Complex Systems and Dependability, pp. 413–423. Springer International Publishing (2015)
27. Siergiejczyk, M., Krzykowska, K.: The analysis of implementation needs for automatic dependent. Marine Navigation and Safety of Sea Transportation: Navigational Problems, vol. 241 (2013)
28. Siergiejczyk, M., Krzykowska, K., Rosiński, A.: Reliability assessment of cooperation and replacement of surveillance systems in air traffic. In Proceedings of the Ninth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, pp. 403–411. Springer International Publishing (2014)
29. Stańczyk, P., Stelmach, A.: Selected aspects of modeling the movement of aircraft in the vicinity of the airport with regard to emergency situations. In: Theory and Engineering of Complex Systems and Dependability, pp. 465–475. Springer International Publishing (2015)
30. Stańczyk, P., Stelmach, A.: The use of on-board flight recorder in the modeling process of aircraft landing operations. In: Safety and Reliability: Methodology and Applications—Proceedings of the European Safety and Reliability Conference, ESREL 2014, pp. 2029–2033 (2015)
31. Tubis, A., Werbińska-Wojciechowska, S.: Inventory management of operational materials in road passenger transportation company—case study. In: CLC 2013, Carpathian Logistics Congress—Congress Proceedings, pp. 65–70 (2014)
32. Vališ, D., Žák, L., Pokora, O.: Contribution to system failure occurrence prediction and to system remaining useful life estimation based on oil field data. Proc. Inst. Mech. Eng. Part O: J. Risk Reliab **229**(1), 36–45 (2015)
33. Valis, D., Pietrucha-Urbanik, K.: Contribution to diffusion processes application in the area of critical infrastructure security assessment. In: Applied Mechanics and Materials, pp. 539–548 (2013)
34. Vintr, Z., Valis, D.: Vehicle maintenance process optimization using life cycle costs data and reliability-centered maintenance (2006)
35. Vintr, Z., Valis, D.: Modeling and analysis of the reliability of systems with one-shot items. In: Reliability and Maintainability Symposium, 2007, RAMS'07, IEEE, pp. 380–385 (2007)
36. Woźniak, W., Jakubowski, J.: The choice of the cost calculation concept for the mass production during the implementation of the non-standard orders. In: 26th IBIMA Conference, Madrit 2015, pp. 2364–2371 (2015)
37. Woźniak, W., Wojnarowski, T.: A method for the rapid selection of profitable transport offers within the Freight Exchange Market. In: 25th IBIMA Conference, Amsterdam 2015, pp. 2073–2085 (2015)
38. Woźniak, W., Stryjski, R., Mielniczuk, J., Wojnarowski, T.: Concept for the Application of Genetic Algorithms in the Management of Transport Offers in Relation to Homogenous Cargo Transport. In: 26th IBIMA Conference, Madrit 2015, pp. 2329–2339 (2015)
39. Zajac, M., Swieboda, J.: An Unloading Work Model at an Intermodal Terminal, In Theory and Engineering of Complex Systems and Dependability, pp. 573–582. Springer International Publishing (2015)

40. Zajac, M., Swieboda, J.: The method of error elimination in the process of container handling. In: 2015 International Conference on Military Technologies (ICMT), IEEE. pp. 1–6 (2015)
41. Zając, M, Żołędziowska, A.: Heuristic support in container trans-shipment decision. In: Carpathian Logistics Congress 2015, in progress
42. Zając, P.: The idea of the model of evaluation of logistics warehouse systems with taking their energy consumption under consideration. Archiv. Civil Mech. Eng. **11**(2), 479–492 (2011)

Scheduling in Grid Based on VO Stakeholders Preferences and Criteria

Victor Toporkov, Dmitry Yemelyanov, Alexander Bobchenkov and Alexey Tselishchev

Abstract A preference-based approach is proposed for Grid computing with regard to preferences given by various groups of virtual organization (VO) stakeholders (such as users, resource owners and administrators) to improve overall quality of service and resource load efficiency. A specific cyclic job batch scheduling scheme is examined which performs job flow scheduling balancing between the VO stakeholders' conflicting preferences and policies. Two different job scheduling evaluation functions are proposed to implement fair resource sharing mechanisms.

Keywords Distributed computing · Grid · Stakeholders · Preferences · Scheduling · Virtual organization · Economic models · Job batch

1 Introduction

In distributed computing with a lot of different participants and contradicting requirements the most efficient approaches are based on economic principles [1–6]. Two established trends may be outlined among diverse approaches to distributed computing. The first one is based on the available resources utilization and application level scheduling [1, 7, 8]. As a rule, this approach does not imply any global

V. Toporkov (✉) · D. Yemelyanov · A. Bobchenkov
National Research University "MPEI", ul. Krasnokazarmennaya, 14, Moscow 111250, Russia
e-mail: ToporkovVV@mpei.ru

D. Yemelyanov
e-mail: YemelyanovDM@mpei.ru

A. Bobchenkov
e-mail: BobchenkovAV@mpei.ru

A. Tselishchev
European Organization for Nuclear Research (CERN), 23, 1211 Geneva, Switzerland
e-mail: Alexey.Tselishchev@cern.ch

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*, Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_44

resource sharing or allocation policy. Another trend is related to the formation of user's virtual organizations (VO) and job flow scheduling [9, 10]. In this case, an external scheduler, e.g. a metascheduler or a meta-broker [9], is an intermediate chain between the users and local resource management and job batch processing systems.

VOs, on one hand, naturally restrict the scalability of resource management systems. On the other hand, uniform rules of resource sharing and consumption, in particular based on economic models, make it possible to improve the job-flow level scheduling and resource distribution efficiency.

In most cases, VO stakeholders pursue contradictory goals working on Grid. VO policies usually contain mechanisms of interaction between VO stakeholders, rules of resource scheduling and distribution and define user shares either statically or dynamically [11, 12]. Besides, VO policy may offer optimized scheduling to satisfy both users' and VO common preferences, which can be formulated as follows: to optimize users' criteria or a utility function for selected jobs [3, 13], to keep resource overall load balance [14], to have job run in strict order or maintain job priorities [15, 16], to optimize overall scheduling performance by some custom criteria [17, 18], etc.

Users' preferences and VO common preferences (owners' and administrators' combined) may conflict with each other. Users are likely to be interested in the fastest possible running time for their jobs with least possible costs whereas VO preferences are usually directed to balancing of available resources load or node owners' profit boosting. In fact, an economical model of resource distribution per se reduces tendencies to cooperate [19]. Thus VO policies in general should respect all members to function properly and the most important aspect of rules suggested by VO is their fairness. A number of works understand fairness as it is defined in the theory of cooperative games, such as fair quotas [12, 20], fair user jobs prioritization [5, 16], and non-monetary distribution [21].

In many studies VO stakeholders' preferences are usually ensured only partially: either owners are competing for jobs optimizing only users' criteria [2, 3], or the main purpose is the efficient resources utilization not considering users' preferences [5], sometimes multi-agent economic models are established [1, 8] which are not allowing to optimize the whole job flow processing.

The cyclic scheduling scheme (CSS) [22, 23] has fair resource share in a sense that every VO stakeholder has mechanisms to influence scheduling results providing own preferences. A flexible approach that takes into consideration VO users' preferences and finds a balance between VO common preferences and users' preferences in some cases in the framework of the CSS is discussed further. The rest of the paper is organized as follows. Section 2 presents a general cyclic scheduling concept. The proposed VO preference based scheduling technique is presented in Sect. 3. Section 4 contains settings for the experiment, and Sect. 5 contains the simulation results for the proposed scheduling approach. Finally, Sect. 6 summarizes the paper.

2 Cyclic Scheduling Scheme and the Fair Resource Sharing Concept

Scheduling of a job flow using CSS is performed in time cycles known as scheduling intervals, by job batches [24]. The actual scheduling procedure consists of two main steps. The first step involves a search for alternative scenarios of each job execution or simply alternatives. An alternative for a single job represents a determined computational resource subset, having start and finish times inside the scheduling interval boundaries. During the second CSS scheduling step the dynamic programming methods [23] are used to choose an optimal alternatives' combination with respect to the given VO criteria. This combination represents the final schedule based on current data on resources load and possible alternative executions.

Alternatives have time (T) and cost (C) properties. In particular, alternatives' time properties include execution CPU time (overall running time), execution start and finish times, total execution runtime. Thus, a common optimization problem may be stated as either minimization or maximization problem of one of the properties, having other fixed or limited, or involve Pareto-optimal strategy search involving both kinds of properties [7, 9, 14]. For example, total job batch execution time minimization with a restriction on total execution cost ($T \rightarrow \min$, $\lim C$).

Alternative executions for each job may be obtained for example as auction-based offers from resource owners [1, 2], as offers from local scheduling systems [3] or by some directed search procedures in a distributed environment [1, 25].

The concept of a fair resource sharing in VO suggests that all stakeholders of a VO should be able to influence scheduling results in accordance with their own preferences. In order to implement such a mechanism, the resource request alongside with a user job has a custom parameter: a user scheduling criterion. An example may be a minimization of overall running time, a minimization of overall running cost etc. [25]. This parameter describes user's preferences for that specific job execution and expresses a type of an additional optimization to perform when searching for alternatives.

The first step of CSS employs a sequential procedure to collect alternative variants for each job of the batch according to the specified optimization criteria. If a batch is considered as a queue, jobs placed at the top take advantage during the search for alternatives and end up with more possible and suitable alternative executions on average (while those jobs that reside at the bottom risk to get unscheduled). The second step of CSS uses alternatives' sets formed according to users' preferences and, thus, users have means to affect the scheduling results.

However, the second step of CSS is based on administrators' criteria and does not normally take users' preferences into account. Trying to meet best average load criteria for example, the system may schedule a single job to run according to the worst alternative from a perspective of a user that has originally submitted that job and user's criteria.

3 Alternative Optimization Based on Users' Preferences

The approach described above has a distinct imbalance. VO administrators in general have much stronger impact on final scheduling results as their criteria (also known as VO common preferences or policy) are employed at the final step of CSS.

In order to recover the balance between VO administrators and users a new type of property is introduced for an alternative. Let us denote that as U (user utility) and treat is as we do with T (time) and C (cost).

So then the second step optimization problem could be in form of: ($C \rightarrow \max$, $\lim U$), ($U \rightarrow \min$, $\lim T$) and so on. Utility for each alternative stands for some value corresponding to a user criterion specified to the resource request for the job. We define U so that the more some alternative corresponds to user's preferences the smaller is the value of U.

Thus, the formal statement of the second step optimization problem for n jobs of the batch could be written as:

$$f(\bar{s}) = \sum\nolimits_{i=1}^{n} f_i(s_j) \rightarrow \text{extr}, u_i(s_j) \leq u_i \leq u^*, \quad (1)$$

where $f_i(s_j)$ is efficiency [24] of an alternative s_j for job i based on VO preferences, $u_i(s_j)$ is a utility for this alternative from the perspective of user, u_i is a partial user utility sum value (for example for jobs $i, i+1, \ldots, n$ or $i, i-1, \ldots, 1$), $u*$ is a general limit on user utility for the whole batch, and $\bar{s} = s_1, \ldots, s_j, \ldots, s_n$.

Average utility $U_a = u^*/n$ which is correlated to the restriction u^* in (1) and can be used to simplify the further analysis.

We consider the two main approaches to represent a user utility function. Assuming that the first alternative found on the first step of CSS is in most cases the best for all users, let the custom utility function be an alternative order $U = 0, 1, 2, \ldots$, etc. We will call it an *order* utility function.

Another approach that has proven to be more flexible is based on the relationship to user-defined optimization criteria. The first alternative found for a job provides the best possible value $Z_{\min}$ of the user-defined criterion which corresponds to the left interval boundary ($U = 0\,\%$), the value of the last alternative found for the job—as the "worst" of all possible ones, corresponds to the right interval boundary $Z_{\max}$ ($U = 100\,\%$). In the general case for each alternative with the value Z of the optimization criterion, U is set depending on its position in $[Z_{\min}; Z_{\max}]$ using the following formula: $U = \frac{Z - Z_{\min}}{Z_{\max} - Z_{\min}} * 100\,\%$.

Thus, each alternative gets its utility in relation to the "best" and the "worst" optimization criterion values user could count on according to the priority of tasks in the batch. Examples of user utility functions for a job with four alternatives and a cost minimization criterion are presented in Table 1.

Using this approach one can describe the optimization task for the second step in CSS as follows: minimize the total job batch execution time while on average

Table 1 User utility examples for a job with execution cost minimization

Job execution alternatives	Execution cost	*Order* utility	*Relative* utility (%)
First alternative	5	0	0
Second alternative	7	1	20
Third alternative	11	3	60
Fourth alternative	15	4	100

ensuring the usage of alternatives with 0–20 % deviation from the best possible scheduling result ($T \rightarrow \min$, $\lim U_a = 20\,\%$).

4 Experiment Settings

An experiment was prepared as follows using a custom distributed environment simulator [26] comprising both application and job-flow scheduling levels.

Virtual organization properties:

- The resource pool includes 100 heterogeneous computational nodes with a relative performance indicator distributed evenly on [2, 10] interval.
- A specific cost of a node is an exponential function of its performance value (base cost) with an added variable margin distributed normally as ± 0.6 of a base cost.
- The scheduling interval length is 600 time quanta.
- The initial resource load with owner jobs is distributed hyper-geometrically resulting in 5–10 % time quanta excluded in total.

Job batch properties:

- Jobs number in a batch is 40.
- Nodes quantity needed for a job is an integer number distributed evenly on [2, 6].
- Node reservation time is an integer number distributed evenly on [100; 500].
- Job budget varies in the way that some of jobs can pay as much as 160 % of base cost whereas some may require a discount.
- Every request contains a specification of a custom criterion which is one of the following: *job execution runtime*, *finish time* and *overall execution cost*.

During an experiment a VO and a job batch is generated. Then CSS is applied to solve the optimization problems with a total utility limited. The results can be examined to analyze processes and the efficiency of the fair resource sharing model.

The important feature of the present study is how users' preferences comply with a VO common policy and optimization. It is evident that if all VO stakeholders (including both users and administrators) are interested in overall running time minimization then the best strategy is to select the first alternatives found for each job. A more complex optimization from (1) is required otherwise.

Thus, an experiment is conducted for studies of the following combinations of VO member's preferences:

- The *mixed* combination: only a half of jobs comply with VO preferences.
- The *conflict* combination: all jobs have custom scheduling criteria that fully contradict VO preferences.

5 Experimental Results

Two series of experiments were carried out for combinations listed above.

The *conflict* combination had the overall job batch execution cost maximized (owners' criterion) with alternatives utility limited ($C \rightarrow \max$, $\lim U$). At the same time for all VO users a criterion is set to minimize each job's execution cost. Thus, VO owners' preferences are clearly opposing users' ones.

The *mixed* combination had the overall cost maximized as well, but only half of jobs voted for cost minimization, the other half had time optimizations (finish time and running time) preferred.

Let us review scheduling results while using relative alternative scoring parameter in regards of user preferences satisfaction. As it can be observed from the diagram, with the utility constraint U_a increasing the total cost C increases as well and reaches its maximum when there is no actual constraint: maximum cost is reached with $U_a = 100\,\%$. In other words, when there is no constraint on the utility ($U_a = 100\,\%$), resulting job cost has the maximum possible value, and in case when the utility restriction is the most stringent ($U_a = 0\,\%$), the resulting cost is the minimum possible. So, by setting the restriction on U_a it is possible to establish some fair scheduling VO policy balanced between common (VO's) and local (users') goals. The result achieved with a $U_a \leq 100\,\%$ constraint corresponds to the cycle scheduling scheme described in [24].

A horizontal dashed line in Fig. 1 marks the average between the maximum (VO administrators' preferences) and minimum (VO users' preferences) cost values achieved in the experiment representing the compromise value. This mutual trade-off average job execution cost value in a conflicting combination is achieved when limiting U_a to $U_a = 25\,\%$, i.e. when the alternatives, that, on average, are not more than 25 % worse, than the best value for user criterion, are scheduled for execution.

Figure 2 shows the average task cost execution for mixed and conflicted combinations together. Cost diagram for a mixed configuration is flatter, the minimum value is greater, and the higher values of the optimization criterion (C) are achieved in comparison to a case of a fully conflict preferences combination given the same value of the user constraint. These features allow VO administrators to have more optimization possibilities owing to the tasks with correlating criteria. The same behavior can be seen when using *order* function as a user utility for possible alternative executions.

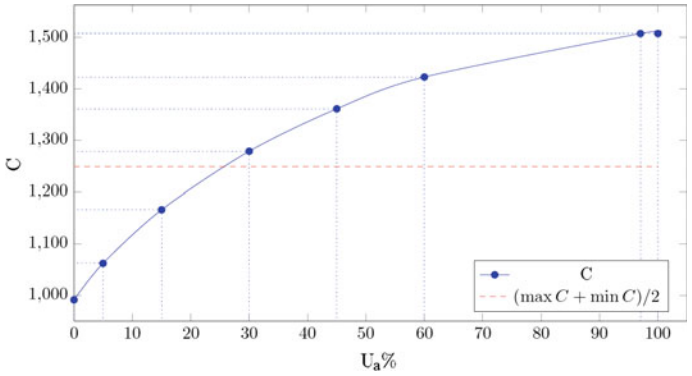


Fig. 1 Average job execution cost in a total job batch cost maximization problem with a *conflict* preferences combination and a *relative* utility function

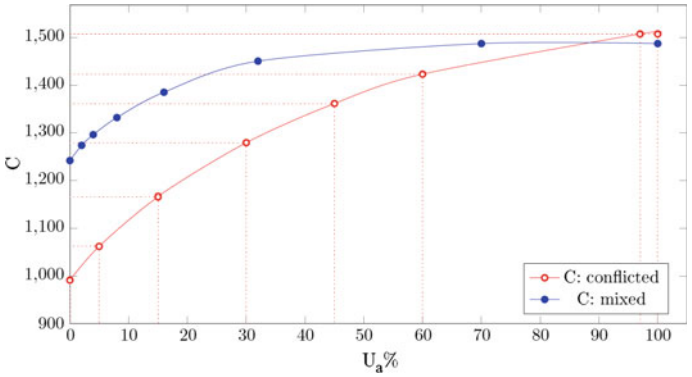


Fig. 2 Average job execution cost comparison in a total job batch cost maximization problem with both *conflict* and *mixed* preferences combinations and a *relative* utility function

Cost diagram for both conflict and mixed preferences combinations is presented in Fig. 3. The graphs changes monotonically from minimum possible cost value (with constraint $U_a = 0$ to a maximum cost value (achieved when $U_a = 5.3$). As average number of alternatives found for a single job during the experiments was 6.3, the constraint $U_a = 5.3$ means the VO policy was able to schedule any alternative including the last one with minimum user criteria value (i.e. without any real constraints).

The average cost value when using order function in a conflict preferences combination reached with only $U_a = 1$ restriction when on average the second best alternatives are selected for each job. These results can be compared to a $U_a = 25\,\%$ restriction needed for a compromise scheduling solution when using a relative utility function. Consequently Figs. 1, 2 and 3 show that VO administrators have

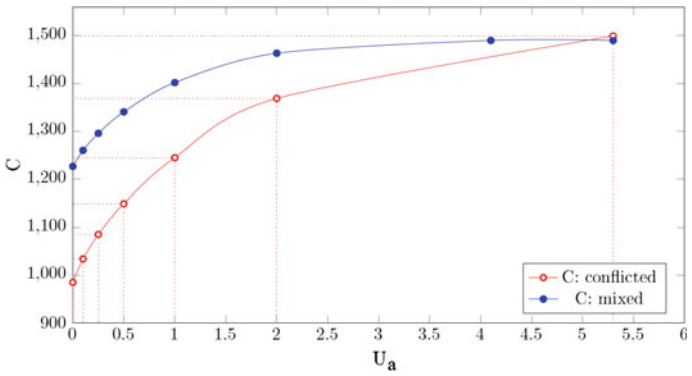


Fig. 3 Average job execution cost comparison in a total job batch cost maximization problem with both *conflict* and *mixed* preferences combinations and an *order* utility function

considerably wide range of options for optimizing the job flow scheduling with relatively little trade-offs for the VO users.

This advantage especially shows up in a case of mixed preferences. Let us consider Fig. 4 to explain this effect. Presented on this figure is the average resulting value of user relative utilities chosen during the scheduling for jobs with different user optimization criteria. In addition, the dashed line represents the actual average value of the user utility for the whole job batch. It can be observed that the relatively smaller average user utility value in a mixed case is provided by the jobs with criteria correlated to the VO optimization criteria. Thus, jobs tending to minimize runtime and finish time have utility value below the average while the jobs with a cost minimization criterion have utility value significantly above the average. It can be explained as jobs executed in a less time use on average more expensive resources and their criteria are in some accordance with common VO criteria (cost maximization). Jobs tending to minimize the execution cost are conflict with the VO preferences. Therefore, the best optimization strategy is to choose

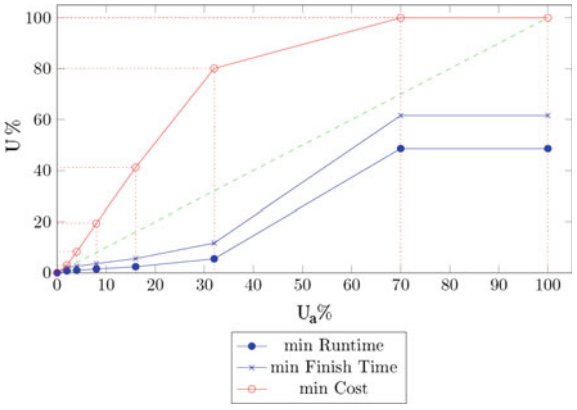


Fig. 4 Resulting utility values for jobs with different optimization criteria in a total job execution cost maximization problem

the first best alternatives for jobs having correlating criteria and the end-of-list alternatives for jobs having conflicting criteria.

This way, the advantage of the VO administrators in a mixed case has been reached mostly on account of ignoring the preferences of users with conflicting criteria.

6 Conclusion and Future Works

In the present paper, a problem of finding a balance between VO stakeholders' preferences to provide fair resource sharing and distribution is studied. In the framework of the cyclic scheduling scheme an approach is proposed which involves user utility combined with time and cost criteria for overall scheduling efficiency estimation. In our work two different utility functions were considered and studied. The order function is based on alternatives, and the priority relative function is based on the relationship to user-defined optimization criteria.

An experimental study included simulation of a job flow scheduling in VO. Different combinations of VO stakeholders' preferences were studied: for example, when some users are in compliance with VO preferences and others being not. A problem of finding a compromise solution which ensures fair resource sharing for scheduling problem is reviewed separately. The experimental results prove the mentioned scheduling scheme suggest tools to establish efficient cooperation between different VO stakeholders even if their goals and preferences are contradictory and to find a balance between VO preferences and those of its users and resource owners.

Further research will be related to additional mechanisms development in order to find a scheduling solution balanced between all VO stakeholders.

Acknowledgments This work was partially supported by the Council on Grants of the President of the Russian Federation for State Support of Young Scientists and Leading Scientific Schools (grants SS-362.2014.9 and YPhD-4148.2015.9), RFBR (grants 15-07-02259 and 15-07-03401), and by the Ministry on Education and Science of the Russian Federation, task no. 2014/123 (project no. 2268).

References

1. Buyya, R., Abramson, D., Giddy, J.: Economic models for resource management and scheduling in grid computing. J. Concurrency Comput. **14**(5), 1507–1542 (2002)
2. Dalheimer, M., Pfreundt, F., Merz, P.: Agent-based grid scheduling with calana. In: Parallel Processing and Applied Mathematics, 6th International Conference, PPAM 2005, pp. 741–750 (2006)
3. Ernemann, C., Hamscher, V., Yahyapour, R.: Economic Scheduling in Grid Computing. In: Feitelson, D., Rudolph, L., Schwiegelshohn, U. (eds.) JSSPP, vol. 18, pp. 128–152. Springer, Heidelberg (2002)

4. Farahabady, M.H., Lee, Y.C., Zomaya, A.Y.: Pareto-optimal cloud bursting. IEEE Trans. Parallel Distrib. Syst. **25**, 2670–2682 (2014)
5. Garg, S., Yeo, C., Anandasivam, A., Buyya, R.: Environment-conscious scheduling of HPC applications on distributed cloud-oriented data centers. J. Parallel Distrib. Comput. **71**(6), 732–749 (2011)
6. Garg, S.K., Konugurthi, P., Buyya, R.: A linear programming-driven genetic algorithm for meta-scheduling on utility grids. J. Par. Emergent Distr. Syst. **26**, 493–517 (2011)
7. Berman, F., Wolski, R., Casanova, H.: Adaptive computing on the grid using AppLeS. IEEE Trans. Parallel Distrib. Syst. **14**(4), 369–382 (2003)
8. Thain, T., Livny, M.: Distributed computing in practice: the condor experience. J. Concurrency Comput. Pract. Exp. **17**(2–4), 323–356 (2005)
9. Kurowski, K., Nabrzyski, J., Oleksiak, A., Weglarz, J.: Multicriteria aspects of grid resource management. In: Nabrzyski, J., Schopf, J.M., Weglarz, J. (eds.) Grid Resource Management. State of the Art and Future Trends, pp. 271–293. Kluwer Acad. Publ. (2003)
10. Rodero, I., Villegas, D., Bobroff, N., Liu, Y., Fong, L., Sadjadi, S.M.: Enabling Interoperability among Grid Meta-schedulers. J. Grid Comput. **11**(2), 311–336 (2013)
11. Gulati, A., Ahmad, I., Waldspurger, C.: PARDA: proportional allocation of resources for distributed storage access. In: FAST '09 Proceedings of the 7th Conference on File and storage technologies, pp. 85–98, California, USA (2009)
12. Carroll, T., Grosu, D.: Divisible load scheduling: an approach using coalitional games. In: Proceedings of the Sixth International Symposium on Parallel and Distributed Computing, ISPDC 07, p. 36 (2007)
13. Rzadca, K., Trystram, D., Wierzbicki, A.: Fair game-theoretic resource management in dedicated grids. In: IEEE International Symposium on Cluster Computing and the Grid (CCGRID 2007), pp. 343–350, Rio De Janeiro, Brazil. IEEE Computer Society (2007)
14. Inoie, A., Kameda, H., Touati, C.: Pareto set, fairness, and nash equilibrium: a case study on load balancing. In: Proceedings of the 11th International Symposium on Dynamic Games and Applications, pp. 386–393, Arizona, USA (2004)
15. Jackson, D., Snell, Q., Clement, M.: Core Algorithms of the Maui Scheduler. In: Revised Papers from the 7th International Workshop on Job Scheduling Strategies for Parallel Processing, JSSPP'01, pp. 87–102, London, UK. Springer (2001)
16. Mutz, A., Wolski, R., Brevik, J.. Eliciting honest value information in a batch-queue environment. In: 8th IEEE/ACM International Conference on Grid Computing, pp. 291–297, New York, USA, (2007)
17. Blanco, H., Guirado, F., Lrida, J.L., Albornoz, V.M.: MIP Model Scheduling for Multi-clusters. In: Euro-Par 2012, pp. 196–206. Springer, Heidelberg (2012)
18. Takefusa, A., Nakada, H., Kudoh, T., Tanaka, Y.: An advance reservation-based co-allocation algorithm for distributed computers and network bandwidth on QoS-guaranteed grids. In: Schwiegelshohn, U., Frachtenberg, E. (eds.) JSSPP 2010, vol. 6253, pp. 16–34. Springer, Heidelberg (2010)
19. Vohs, K., Mead, N., Goode, M.: The psychological consequences of money. Science **314** (5802), 1154–1156 (2006)
20. Kim, K., Buyya, R.: Fair resource sharing in hierarchical virtual organizations for global grids. In: Proceedings of the 8th IEEE/ACM International Conference on Grid Computing, pp. 50–57. IEEE Computer Society, Austin, USA (2007)
21. Skowron, P., Rzadca, K.: Non-monetary fair scheduling cooperative game theory approach. In: Proceeding of SPAA '13 Proceedings of the Twenty-Fifth aNnual ACM Symposium on Parallelism in Algorithms and Architectures, pp. 288–297. ACM, New York, NY, USA (2013)
22. Toporkov, V., Toporkova, A., Tselishchev, A., Yemelyanov, D., Potekhin, P.: Preference-based fair resource sharing and scheduling optimization in grid VOs. Proc. Comput. Sci. **29**, 831–843 (2014)

23. Toporkov, V., Toporkova, A., Tselishchev, A., Yemelyanov, D., Potekhin, P.: Metascheduling and heuristic co-allocation strategies in distributed computing. J. Comput. Inf. **34**(1), 45–76 (2015)
24. Toporkov, V., Tselishchev, A., Yemelyanov, D., Potekhin, P.: Metascheduling strategies in distributed computing with non-dedicated resources. In: Zamojski, W., Sugier, J. (eds.) Dependability Problems of Complex Information Systems, Advances in Intelligent Systems and Computing (AISC), vol. 307, pp. 129–148. Springer International Publishing Switzerland (2015)
25. Toporkov, V., Toporkova, A., Tselishchev, A., Yemelyanov, D.: Slot selection algorithms in distributed computing. J. Supercomput. **69**(1), 53–60 (2014)
26. Toporkov, V., Tselishchev, A., Yemelyanov, D., Bobchenkov, A.: Composite scheduling strategies in distributed computing with non-dedicated resources. Proc. Comput. Sci. **9**, 176–185 (2012)

Vulnerability of Passenger Transportation System—The Main Information Provided by Key Stakeholders. Case Study

Agnieszka Tubis and Sylwia Werbińska-Wojciechowska

Abstract In the presented paper, authors focus on the issues connected with passenger transportation system vulnerability assessment. Following this, in the article the authors investigate the problem of information needs that make possible a full risk analysis performance for municipal transport system. Thus, there is presented a short literature review connected with assessment methods used in the analysed research area. There are also defined the main definitions. This gives the possibility to define the main information system that meets of vulnerability performance. There is also given an analysis aimed at investigation of potential use of vulnerability analysis results by public transport stakeholders. Article ends with some conclusions and directions for future research.

Keywords Transportation system · Vulnerability · Stakeholders · Information

1 Introduction

Transportation systems are subject to degradations of its services which may result in degradation of system performance [3, 23]. This negative events are called disruptions and may be unplanned and unexpected (e.g. vehicles' technical failures, accidents, natural disasters), planned (e.g. capacity reductions due to planned construction work or repairs, planned maintenance operations), or known some time in advance (e.g. crew strikes). The classification of transportation system disruptions is given in e.g. [28, 32].

Such disruptions increase transportation system vulnerability, what may cause the negative consequences for all group of stakeholders [14]. Non-efficient transport

A. Tubis (✉) · S. Werbińska-Wojciechowska
Wroclaw University of Technology, 27 Wybrzeze Wyspianskiego Str., Wroclaw, Poland
e-mail: agnieszka.tubis@pwr.edu.pl

S. Werbińska-Wojciechowska
e-mail: sylwia.werbinska@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*, Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_45

imposes significant costs on society in terms of environmental, safety and health impacts [17, 25, 31]. For example it causes difficulties to rationally use resources, reduces labour division, decrease people's quality of life (e.g. congestions), or distorts business activity [14, 15, 25].

Nowadays, the level of society's dependence from achieving reliable transport service increases. Thus, such issues as vulnerability assessment or risk management are gaining in importance (see e.g. [2, 13, 14, 20, 33]). Taking into account the sustainable transport idea, where a transport system should be characterized by accessibility, quality and safety of offered services, and environmentally friendly, properly conducted risk analysis and vulnerability analysis are the base in the process of potential disruption occurrence controlling and risk management performance (see e.g. [21, 24]).

Moreover, stakeholders' involvement in the supply of public transport services is necessary in order to meet the main stakeholder—passenger needs [16, 25]. Following this, supplying the passenger with the best service is connected with creating the effective relationships between different stakeholders (municipality, public transport company, state road maintenance service, parts suppliers, etc.). The effective customer service is possible only when the necessary information flows between the main stakeholders are performed. These information is to be used in planning, managing and operating public transport processes. Taking into account, that public transport organizations operate and develop under dynamic circumstances, the groups of necessary information/data, as well as the level of obtained data, may change in time.

Following this, authors focus on the problem of information needs that make possible a full risk analysis performance for municipal transport system. The identification of information needs is based on vulnerability analysis requirements, stakeholders' expectations, and current information base developed by chosen passenger Transport Company in Poland. Thus, the remainder of this paper is organized as follows: Sect. 2 introduces the short literature review connected with assessment methods used in the analysed research area. There are also defined the main definitions and key stakeholders of public transport system. Section 3 presents the main information system that meets information needs of vulnerability analysis performance. Section 4 is aimed at investigation of potential use of vulnerability analysis' results by public transport stakeholders. Section 5 concludes the paper.

2 Vulnerability of Transportation Systems—Transportation, Stakeholders, Assessment Issues

The proper definition of vulnerability analysis' information needs requires the short introduction with the main definitions connected with investigated research area.

For any transportation system, *its main function is to realize the movement of persons and goods from A place to place B by safety and efficient way with*

minimum negative impact on the environment [9]. Additionally, to ensure the functionality of the transport system is defined as *the ability of the system to meet the requirements of the mobility of passengers/cargo and uninterrupted movement of people and goods in time and space, taking into account e.g. social and economic objectives of the society*, including e.g. [9, 11, 30]:

- means of transport availability,
- requirements conducted by law regulations,
- the principle of social justice,
- environmental factors (ecology),
- performed transport work,
- safety and security.

In addition, the identification of the transport system requires knowledge of basic elements, such as (Fig. 1) [9, 11]:

- infrastructure,
- system elements (means of transport)—the knowledge about the type its quantity, reliability characteristics,
- humane resources,
- information flow,
- performed maintenance and operational tasks, and rules for their organization,
- elements influencing the decision-making process in the system.

Based on this, it is possible to define the transportation system of Poland as *a system of demand and supply markets interconnected by material and information flows in which the processes involved in the movement of people/cargo are carried out in an economically efficient manner while maintaining an appropriate level of service* [10]. In this sense, the system covers the whole country geographically. Moreover, this definition is also in line with the objectives of sustainable development strategy [8].

For the purpose of this article, authors focus only *on the public transportation system that operate in chosen city in Poland (Lower Silesia region). The analysed system offers passenger transport service (by city buses and trams) which is available for use by the general public in the city agglomeration.* Following this,

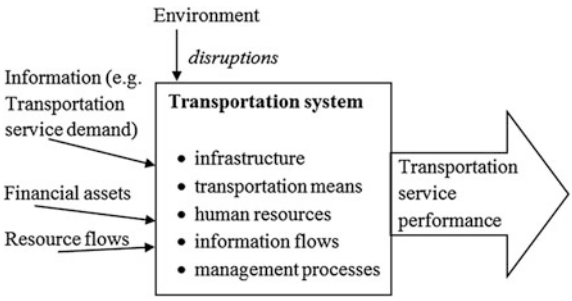


Fig. 1 Transportation system and its elements. *Source* Own contribution based on [11, 27]

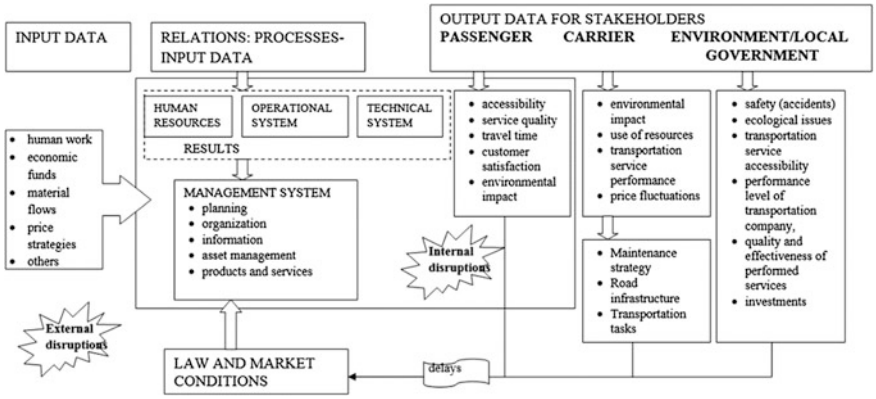


Fig. 2 Transportation system model with taking into account the key stakeholders. *Source* Own contribution based on [5, 27]

there are a number of actors currently involved in the planning and organising of public transport in the chosen city. For example, based on work [19], there can be defined at least 7 key stakeholders, when the buses/trams are treated as tools and as end products in the supply chain. These stakeholders are passengers, operator, federal and state government, bodywork manufacturer, industry body and chassis manufacturer. Developing transportation system model for given city agglomeration (Fig. 2) there were defined three groups of stakeholders that play a role in terms of public transport planning and decision making- passengers, a municipal carrier and local government.

The performance of public transport system may be interrupted by internal and external disruptions that influence the travel behaviour of key stakeholder—passengers. To effectively prevent and minimize the effects of occurred disruptions, there should be periodically conducted risk and vulnerability analyses.

The concept of transport system vulnerability is analysed in multidimensional way in the known literature. Thus, there is no one commonly recognized definition that would be efficient in every research investigations [18]. One of the first work that defines the transportation vulnerability is [2]. Author in this work defines transportation vulnerability in terms of road network serviceability. This definition is broadly used in the literature and valid for most of the transportation systems performance [18].

Another approach to the definition of transportation vulnerability refers to the *measure of probability of an unwanted event (disruption) occurrence and the consequences of a disruption appearance/critical elements not performance* (see e.g. [7, 12, 14, 16]). Another definition is proposed by Ouyang et al. in work [20], where authors defines vulnerability *as decrease in the performance level due to an unwanted/troublesome event occurrence*. A brief summary of the vulnerability literature is proposed e.g. in works [18, 22].

There can be found a lot of works in the current literature that are aimed at transportation system vulnerability assessment (especially transportation network vulnerability assessment). The developed approach proposes some methods as well as measurement indexes, see e.g. works [1, 6, 24, 26, 29].

To sum up, recently published works on transportation system vulnerability assessment focus mainly on the investigation of transport network performance during its nodes disruption occurrence. There is no research aimed at problem of transportation systems vulnerability assessment treated as prediction element being implemented in the process controlling system, whose task is to ensure the correctness of passenger service processes performance. Furthermore, the proper implementation of the vulnerability analysis requires data gathering and processing as well as cooperation between key stakeholders. This issue is investigated in the next sections.

3 The Main Information System that Meet Information Needs of Vulnerability Analysis Performance

In the previously conducted research work, the authors have developed a model of vulnerability analysis as a tool for a process controlling tailored to the needs of the public transport company. The procedure involves three stages of implementation:

- identification of internal and external disruptions to the public transport system performance,
- scenario planning aimed at preventing and reducing the effects of potential hazards occurrence,
- audits focused on improving the system response to identified threats.

Comprehensive analytical procedure is presented in [28].

In order to prepare a comprehensive vulnerability analysis there is necessary a cooperation of many organizations who act as stakeholders in the system of public transport. At the same time, there can be distinguished three main groups of stakeholders (Fig. 3):

(1) The company providing passenger transport services in the field of public transport,
(2) The organizer of public passenger transport in the agglomeration, which is responsible for giving service permits, licenses, etc. On the basis of the Public transport from 16 December 2010, the organizer of municipal passenger transport is the Municipality of given agglomeration,
(3) passengers that using municipal transport services.

Only the first two organizations are to be responsible for preparing vulnerability analyses, while passenger are primarily the beneficiaries of their results in the form

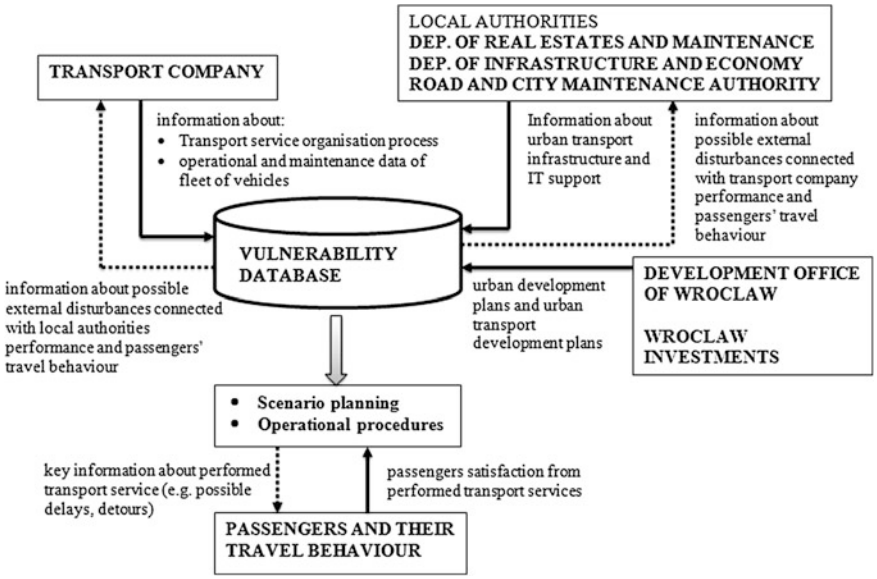


Fig. 3 The main information flows from the public transport key stakeholders and database developed for vulnerability analyses performance

of procedures, guidelines and provided information. Both stakeholders are responsible for preparation of vulnerability analyses tailored to their decision-making processes and level of responsibility for public transport performance. These analyses mostly relate to the same issues (e.g. risk of transport performance quality decrease), but due to the identified different types of risks, they are complementary.

Research conducted by the authors are focused mainly on preparing the vulnerability assessment method adapted to the needs of the transport company. However, the limitation mainly to the data being at disposal of the transport company is insufficient from the point of view of the adopted scope of analysed areas. It should be noted, that the transport company is responsible only for transportation service organization (and therefore for the quality of provided services) and for used fleet of vehicles. Following this, in the Fig. 3, there are presented the main information flows from public transport key stakeholders that supply the data base used for vulnerability analysis performance. According to the scheme, the transportation carrier is to be supplied the developed information system with data concerning mainly:

- vehicles operational and maintenance data (including primarily quantity, capacity, condition, age),
- available human resources (including in particular the operational staff—drivers, mechanics, dispatchers),

- service quality parameters (including in particular the frequency of service, punctuality, safety, comfort),
- efficiency, reliability and complexity of IT systems (including primarily support systems for on-board computers in vehicles, financial and accounting systems, repair and storage systems, measurement systems),
- reliability of logistics systems that support warehouse management and material flow management within the enterprise,
- procedures for crisis management that are obligated in the company,
- IT systems to support processes of cars monitoring,
- system of communication with passengers and the company's internal communication system.

Other elements of the municipal transport system highlighted in the analysis (stops, the remaining urban transport infrastructure, and IT support) remain in the management of other defined group of stakeholders. Thus, in the example of Wroclaw city, the main tasks connected with municipal transport organization performs Mayor of Wroclaw. Therefore, we can distinguish the following units of the Municipal Office and Municipalities budgetary units jointly responsible for the planning and organization of the transport system:

- Department of Real Estates and Maintenance → Department of Transportation, including: Department of Public Transport Management and Team for Cooperation with the Traffic Management Centre and Public Transport,
- Department of Infrastructure and Economy → Team for Transport Development and the Department of Urban Engineering
- Road and City Maintenance Authority.

These Local Office units should prepare vulnerability analysis for planning their own activities, and then in the framework of improving urban transport security system, make available the required data to the carrier. Priority should be given here for information on:

- operation and maintenance of road infrastructure,
- operation and maintenance of rail infrastructure,
- planning timetables,
- performance and management of passenger information system,
- functioning of the Intelligent Transport System and obtained measurements;
- ticket distribution system,
- results of audits in quality of transport services,
- IT systems to support monitoring of the city.

The information exchange should of course be bilateral. The carrier also must provide the necessary data to the Municipal Office and supplying its information system. In this way, it would increase the efficiency and accuracy of performed analysis, which positively affect the quality of the planning and transportation system reliability.

The information system developed for vulnerability analysis should also be supplied by data from such Local Offices as the Development Office of Wroclaw and Wroclaw Investments. Data transferred from these units will focus on the urban development plans and the development of public transport, which on one hand may affect the identification of controlled hazards (e.g. the planned shutdown of selected sections of routes) and generate the occurrence of uncontrolled events (e.g. increased risk of road incidents).

4 Vulnerability Analysis and Their Potential Use by Public Transport Stakeholders

Vulnerability analysis is focused on increasing the reliability of the public transport system and improvement of its resilience to emerging hazards. Therefore, identification of potential disruptions occurrence is only the first step of the procedure, which must be continued in the form of scenario planning and preparation of hazard procedures in the event of an emergency.

Currently, both groups of stakeholders (carrier more narrowly, the organizer of transport in a wider range) identify possible hazards/disruptions to the public transport system in the frame of crisis management, but this is done only for their own needs. According to the authors, there are currently close collaboration between distinguished groups of stakeholders in the framework of preventive actions performance. Cooperation is usually taken only when one of hazard events occur, when it is only possible to take corrective actions to limit the damages.

Meanwhile, cooperation in this field seems to be necessary and should not only pertain to information about current local events, technical failures and natural disasters. Crisis Management Centre of the city of Wroclaw in their duties is keeping records of random events, technical failures and natural disasters, which significantly disrupted the functioning of the city of Wroclaw and documenting their progress [4]. These data should constitute the basis for determining the probability of these disruptions occurrence in the future, which contributes significantly to vulnerability analysis performance developed by municipal transport company. Nowadays, there is a lack of assessment analyses that would support decision-making processes of managers in the organization.

City Office is also responsible for preparing appropriate emergency response procedures, which should form the basis scenario prepared plans aimed at eliminating the effects of events not controlled by the carrier. At the same time, prepared in the frame of controlling scenarios for transport company may form the basis for the preparation of proposed solutions submitted by the Crisis Management Centre to local authorities in case of emergency occurrence. To make this possible, both groups of stakeholders need to cooperate in the course of predictive and preventive actions performance.

The results of the conducted analysis and prepared on its base performance scenarios should also take into account the information needs of the third group of stakeholders of public transport system, namely its users. Research conducted by students of Wroclaw University of Technology in November–December 2014 and November–December 2015 clearly indicated that passenger information systems do not meet expectations in case of emergency. At the same time, the current communication infrastructure allows for proper and up-dated information for passengers about occurred disturbances and activities performed to limit their consequences. Thus, one of the key success factor—the system of communication/information sharing fails.

5 Summary

The results presented in this paper are a continuation of research conducted by the author in the area of development of the concept of vulnerability adapted to the needs of a transport company and being a tool in the process controlling system. The effects of this study clearly demonstrate the need for strong cooperation between the different groups of stakeholders in the area of a comprehensive vulnerability analysis development. Every organization gathers and analyses specific data whose acquisition is both time consuming and capital intensive. Thus, in order to ensure the effectiveness of forthcoming analysis and its actuality, it is necessary to achieve a cooperation between transport company and the organizer of public transport. This cooperation should mainly involve the exchange of data and scenario planning process, but also should include providing of information about the control of the correctness of taken assumptions.

The planned further research in this area are to be focused on a comprehensive vulnerability analysis performance for the analysed passenger transport company and use of obtained results to improve their business.

References

1. Balijepalli, Ch., Oppong, O.: Measuring vulnerability of road network considering the extent of serviceability of critical road links in urban areas. J. Transp. Geogr. **39**, 145–155 (2014)
2. Berdica, K.: An introduction to road vulnerability: what has been done, is done and should be done. Transp. Policy **9**, 117–127 (2002)
3. Cats, O., Jenelius, E.: Beyond a complete failure: the impact of partial capacity reductions on public transport network vulnerability. In: The 6th International Symposium on Transportation Network Reliability (INSTR), Japan, pp. 1–10 (2015)
4. Crisis Management Centre. http://bip.um.wroc.pl/artykul/100/3226/centrum-zarzadzania-kryzysowego. Accessed 27 Jan 2016
5. Dajani, J.S., Gilbert, G.: Measuring the performance of transit systems. Transp. Plann. Tech. **4**, 97–103 (1978)

6. El-Rashidy, R.A., Grant-Muller, S.M.: An assessment method for highway network vulnerability. J. Transp. Geogr. **34**, 34–43 (2014)
7. Erath, A., Birdsall, J., Axhausen, K.W., Hajdin, R.: Vulnerability assessment methodology for Swiss road network. Transp. Res. Rec. J. Transp. Res. Board **2137**, 118–126 (2009)
8. European Commission. http://ec.europa.eu/environment/eussd/. Accessed 27 Jan 2016
9. Fricker, J.D., Whitford, R.K.: Fundamentals of Transportation Engineering. A Multimodal Systems Approach. Inc., Upper Saddle River, New Jersey (2004)
10. INVAPO WP5 5.4.6.: Technical, economical and environmental studies of Vistula river revitalization and navigability restoration on Warsaw-Gdansk tour. Gdynia Maritime University, Gdynia (2014)
11. Jacyna, M.: Modelling and assessment of transportation systems (in Polish). Publishing House of Warsaw University of Technology, Warsaw (2009)
12. Jenelius, E.: Network structure and travel patterns: explaining the geographical disparities of road network vulnerability. J. Transp. Geogr. **17**(3), 234–244 (2009)
13. Jodejko-Pietruczuk, A., Plewa, M.: Reliability based model of the cost effective product reusing policy. In: Safety and Reliability: Methodology and Applications—Proceedings of the European Safety and Reliability Conference, ESREL 2014, pp. 1243–1248, Taylor & Leiden (2014)
14. Johansson, J., Hassel, H.: An approach for modelling interdependent infrastructures in the context of vulnerability analysis. Reliab. Eng. Syst. Saf. **95**, 1335–1344 (2010)
15. Kierzkowski, A., Kisiel, T.: The simulation model of logistic support for functioning ground handling agent, taking into account the probabilistic time of aircrafts arrival. In: CLC 2013: Carpathian Logistics Congress—Congress Proceedings, pp. 463–469, Tanger LTd. (2013)
16. Kierzkowski, A., Kisiel, T.: Functional readiness of the security control system at an airport with single-report streams. In: Theory and Engineering of Complex Systems and Dependability: Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, June 29–July 3 2015, Brunów, Poland, Springer (2015)
17. Kisiel, T., Valis, D., Zak, L.: Application of regression function—two areas for technical system operation assessment. In: CLC 2013: Carpathian Logistics Congress—Congress Proceedings, Tanger LTd. pp. 500–505 (2013)
18. Mattsson, L.-G., Jenelius, E.: Vulnerability and resilience of transport systems—a discussion of recent research. Transp. Res. Part A **81**, 16–34 (2015)
19. Napper, R.: Route bus transport—stakeholders, vehicles and new design directions. In: Proceedings of the 29th Conference of Australian Institutes of Transport Research, Adelaide (2007)
20. Ouyang, M., Pan, Z., Hong, L., He, Y.: Vulnerability analysis of complementary transportation systems with applications to railway and airline systems in China. Reliab. Eng. Syst. Saf. **142**, 248–257 (2015)
21. Proper, J.W.: How to manage resilience in public transport organizations (2011). www.cvs-congres.nl/cvspdfdocs/cvs11_031.pdf
22. Reggiani, A., Nijkamp, R., Lanzi, D.: Transport resilience and vulnerability: the role of connectivity. Transp. Res. Part A **81**, 4–15 (2015)
23. Restel F.J.: Defining States in Reliability and Safety Modelling. In: Theory and Engineering of Complex Systems and Dependability: Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, June 29–July 3, 2015, pp. 413–423, Brunów, Poland, Springer (2015)
24. Rodrigues-Nunez, E., Garcia-Palomares, J.C.: Measuring the vulnerability of public transport networks. J. Transp. Geogr. **35**, 50–63 (2014)
25. Susnienie, D., Jurkauskas, A.: Stakeholder approach in the management of public transport companies. Transport **23**(3), 214–220 (2008)
26. Taylor, M.A.P., D'Este, G.M.: Transport network vulnerability: a method for diagnosis of critical locations in transport infrastructure systems. In: Murray A.T., Grubesic T.H. (eds.) Critical Infrastructure. Reliability and Vulnerability, Springer (2007)

27. Tsolakis, D., Thoresen, T.: A framework for demonstrating that road performance meets community expectations. Road Transp. Res. **7**(3), 79–85 (1998)
28. Tubis, A., Werbińska-Wojciechowska, S.: Issues on vulnerabilility of passenger transportation system (in Polish). In: Proceedings of Winter School of Reliability: Reliability of technical systems, 10th–16th January, 2016, Szczyrk, Poland, Publishing House of Warsaw University of Technology (2016)
29. Wang, Z., Chan, A.P., Yuan, J., Xia, B., Skitmore, M., Li, Q.: Recent advances in modelling the vulnerability of transportation networks. J. Infrastruct. Syst. **21**, 2 (2015)
30. Zając, M.: Principles of work load in intermodal transhipment point. In: CLC 2013: Carpathian Logistics Congress—Congress Proceedings, pp. 685–690, Tanger LTd. (2013)
31. Zajac, M., Swieboda, J.: An Unloading Work Model at an Intermodal Terminal. In: Theory and Engineering of Complex Systems and Dependability: Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, June 29–July 3 2015, pp. 573–582, Brunów, Poland, Springer (2015)
32. Zhu, S., Levinson, D. M.: Disruptions to transportation networks: a review. In: Levinson, D. M., Liu, H.X., Bell, M. (eds.) Network Reliability in Practice. Selected Papers from the Fourth International Symposium on Transportation Network Reliability, pp. 5–20. Springer, New York (2012)
33. Zurek, J., Smalko, Z., Zieja, M.: Methods applied to identify causes of air events. In: Bris, R., Soares, C.G., Martorell, S. (eds.) Reliability, Risk and Safety: Theory and Applications, pp. 1817–1822. Taylor & Francis (2010)

Asynchronous System for Clustering and Classifications of Texts in Polish

Tomasz Walkowiak

Abstract The paper presents an online system for clustering and classification of texts in the Polish language. It allows running complex workflows of language and machine learning tools. A high throughput and low latency was achieved by an asynchronous style of programming and a usage of message oriented middleware—RabbitMQ. Authors discuss the architecture assumptions, the language processing modelling notation for a workflow definition and the system architecture. Moreover, a sample Single Page Application is presented that clusters uploaded corpora and shows results online.

Keywords Natural language processing · Polish language · Web application · Clustering · Classification

1 Introduction

Computer-based text analysis is nowadays becoming more and more popular. It includes such tasks as information retrieval, data analysis, classification, genre recognition and sentiment analysis [9]. It became usable due to three reasons: amount of data, availability of language tools and computational resources. First, we have large amounts of textual data in digital format. Starting from official publications (books, journals, newspapers and magazines), through on-line publications (web pages, blogs, posts, tweets), to personal communications (e-mails, SMSs). Secondly, language and machine learning tools are under development for many years. For almost all popular languages, tools like taggers [10], name entity recognizers [6] and parsers [14] are available. The most mature, in sense of

T. Walkowiak (✉)
Faculty of Electronics, Wroclaw University of Science and Technology,
Wybrzeze Wyspianskiego 27, 50-320 Wroclaw, Poland
e-mail: tomasz.walkowiak@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_46

accuracy, speed and code stability, are tools for English (for example NLTK[1]). Thirdly, since processing of large texts requires a huge computational power, an availability of computer resources (university data centers, private or public clouds) become an important factor for a widespread of computer-based text analysis.

However, building its own computer-based text analysis workflow is a problematic process, especially for researchers without an experience in computer engineering. Large number of tools, especially for non-English languages, like Polish, are very hard to be installed and integrated [11]. It is mostly due to a fact that they were developed as a research code to verify proposed by authors' algorithms. Moreover, language tools are being developed in different technologies (like C++, Python, Java, R and OCaml). They are very often unstable and hard to be scaled. These problems are overcome by making linguistic tools available in Internet [13] and developing web based applications, which allows to process text data online. It allows researcher to use already installed and tested tools. In case of Polish, such functionality has Multiservice [7] and multilingual WebLicht [5] that includes tools for Polish language.

However, these tools were not designed for processing large corpora and they are not very efficient. They add a large time overhead due to data transmission and exhaustive data formats. Moreover, they are only language tool oriented, lacking machine learning tools. Additionally they allow constructing simple pipelines of tools, not more sophisticated workflows as is required for clustering or classification tasks.

That is why authors design and built a web accessible system for clustering [14] and classifications of texts in Polish. One of the system requirements was to provide high throughput and low latency. The system was built based on the experience gained during development of the test engine [9, 11, 12]. Section 2 describes architecture assumptions. Presented there workflows for clustering and classification allowed to define the modelling notation presented in Sect. 4. The noticed problems in the test engine implementation (discussed in Sect. 3.1) resulted in a redefinition of original architecture [11] as presented in Sect. 4 and a usage of asynchronous approach (discussed in Sects. 3.2 and 3.3) achieved by a usage of message oriented middleware [2]—RabbitMQ.[2]

2 Architecture Assumptions

The aim was to develop a system that will allow defining and running a workflow of language and clustering tools. Let us analyze an example workflow for clustering as presented in Fig. 1. As an input, we have a set of texts that we want to process by a set of activities (marked as boxes in Fig. 1). At first, the documents are converted

[1] http://www.nltk.org/.

[2] https://www.rabbitmq.com.

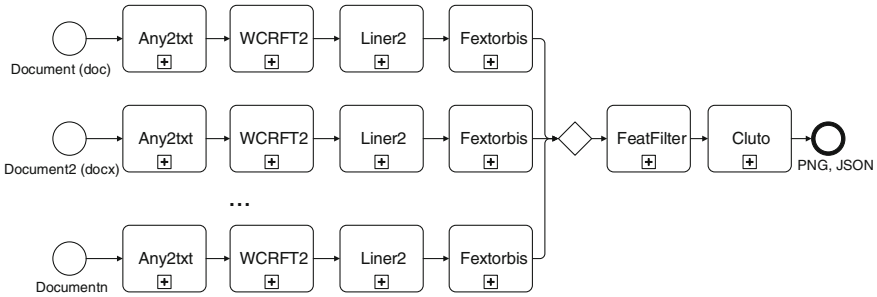


Fig. 1 Text clustering workflow

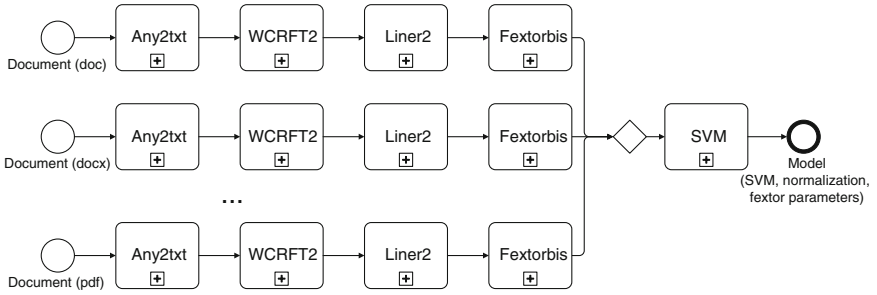


Fig. 2 Learning workflow for text classification

to a uniform text format (Any2txt box). Next, each text is analyzed by a part-of-speech tagger (WCRFT2 [10]) and then it is piped to a name entity recognizer (Liner2 [6]). After it, the feature extraction module [4] comes into stage (box named Fextorbis). A set of morphological and grammatical features as proposed in [9] were used. Next, the data received from the feature extraction for each input file has to be unified, filtered and transformed by the activity named FeatFilter. It uses a range of methods from the system for distributional semantics SuperMatrix [3] and statistical R package.[3] Finally, a clustering tool called CLUTO[4] is run to cluster texts.

As it could be noticed part of processing of each document (pipeline: Any2txt, WCRFT2, Liner2, Fextorbis) could be run in parallel. Next, marked by rhombus in Fig. 1, all parallel pipelines must complete. They are merged and next processing pipeline is called, consisting of FeatFilter and Cluto.

For classification, the other workflow must be defined, since the supervised classification consists of two stages: learning (workflow presented in Fig. 2) and

[3] https://www.r-project.org.

[4] http://glaros.dtc.umn.edu/gkhome/cluto/cluto/overview.

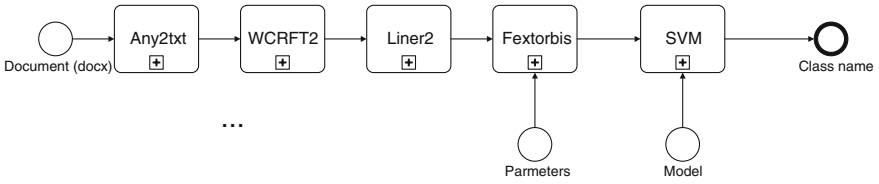


Fig. 3 Text classification workflow

classification (workflow presented in Fig. 3). It uses Support Vector Machine implementation from scikit-learn[5] Python package.

The defined processing activities (Any2txt, WCRFT2, Liner2, Fextorbis, FeatFilter, SVM and Cluto) are software tools developed in different languages (Java, C++, Python and R) and available in a form of source code or executive binaries (CLUTO). Moreover, some of them (WCRFT2, Liner2 and SVM in classification) have large models. Therefore, the time of loading a model is much longer than processing a single text file. Therefore, tools have to work as services with loaded data (models) in memory. The defined activities have a common structure. They have one input file (or directory), set of configuration parameters (for example defining the used model) and produces one file (or directory) as output.

Summarizing, the proposed system has to have as an input a corpus and a description of the workflow. Next, the workflow has to be run, calling in parallel required language or machine learning tools. Therefore, there is a need to connect to one another system components (language and machine learning tools). It requires some middleware that will enable communication and management of a text analysis process.

3 Asynchronous Communication

3.1 Analysis of the Test Engine Drawbacks

The middleware of our previous system [11] was built based on a propriety queue system realized as a set of REST [1] services implemented in Python framework CherryPy[6] and deployed in Apache server. It was accessed via HTTP protocol in a pooling manner. A pooling approach requires that each system component (client or server) has to connect to a system middleware several times per seconds. It caused a set of efficiency problems. The number of open connections is equal to number of workers and number of texts in analyzed corpora. Therefore, it could easily exceed

[5] http://scikit-learn.org/.

[6] http://www.cherrypy.org/.

limit of number of open connections for Apache server, which is 1000. The problem become noticeable even faster (for smaller corpora) due to a fact that used implementations of HTTP clients (for example Python urlib2 library or Java HttpConnect with default property "keepAlive" set on) are not closing TCP/IP immediately. As a result, the number of open TCP/IP connections on web enlarge limits. Second efficiency problem was caused by a usage of Python for the middleware. The standard implementation of Python prevents two threads in the same process from running code at the same time. It results with a maximum usage of only one processor by the middleware. It additionally limited the throughput of the system.

The next, important element of text processing system is the workflow execution engine. It was realized in a synchronous manner using one thread for each processing pipeline. Therefore, the number of threads was equal to a number of files in corpus. That caused problems for corpora with tens of thousands of texts.

3.2 Asynchronous Architecture

The processing of the workflow could be seen as a set of requests to language and machine learning tools. The most common approach of handling a set of requests is synchronous. It works in standard blocking input/output way. Each incoming request is assigned to a separate thread. The request thread is blocked until the response is not returned to the client. After that, thread goes back to the thread pool and is able to handle another request. Such architecture in terms of handling incoming requests has advantages, but has also some cons especially, when the number of incoming requests grows up. In case of time-consuming operations on the request thread, it is unable to assign it to another connection while the job is not finished [8].

Such problems were noticed in the test engine as discussed in the previous chapter. That is why we decided to use different architecture—asynchronous one. It has a single thread running an event loop and waiting for incoming events. Events are emitted by such external sources such as TCP/IP connections. The asynchronous approach gets rid of context switching overhead and thread stack for each processed request. Request thread with event loop handles event sequentially. This thread must be fast and cannot be blocked for a long time [8].

3.3 Message Oriented Middleware

The required middleware technology should allow sending and receiving messages in asynchronous way between different language and machine learning tools distributed in the local network. If required the messages should be queued. Such functionality could be achieved by a message oriented middleware [2]. It is based

on a message broker that provides common infrastructure for interactions between distributed components of the system.

Many message oriented middleware solutions are available, like for example: IBM WebSphere MQ, NServiceBus, Tarantool, RabbitMQ, Apache ActiveMQ or Qpid. Among them RabbitMQ was selected as one of the mostly widespread message oriented middleware. It is an open source message broker written in Erlang. RabbitMQ supports many messaging protocols, among them the most important is the AMQP[7] standard. It has clients for a large number of platforms (C++, Java, Python) as required by technologies used by language and machine learning tools. Moreover, RabbitMQ has a good performance and scalability.

4 System Architecture

4.1 Architecture Overview

The described functionalities and the selection of message-oriented middleware result in the system architecture presented in Fig. 4. The core of the system consists of the RabbitMQ that queues tasks and NLPEngine that executes workflows.

The data processing is done by language and machine learning tools working as workers. Each worker collects a task from one of queues managed by RabbitMQ. The task consist of parameters specific for a worker (in JSON format) and name of input and output file (or directory). The worker loads data from the Samba network file system, processes them and returns the result to the Samba file system, informing the NLPEngine when the task was finished. The network file system was used to store input and output files since they can be very large. The size of text file during processing could be enlarged by 30 times [11].

Almost all communication between system components goes through RabbitMQ. An exception is the communication between external applications and the LPMN engine that goes through a database. It is used to inform the web application (by NLPREST2) of the workflow processing progress. Moreover, the database is used for logging purposes.

The NLPREST2 grants the access to the system from the Internet. It consists of a set of REST [1] services that allows to load files to the system, start the workflow execution, monitor the processing, and download results. Such approach allows an easy integration with almost any kind of application (mobile ones, web base, desktop or server one), built nearly in any programming language. RabbitMQ sends the received workflow to LPMN engine that sends tasks to required workers.

[7] https://www.amqp.org/.

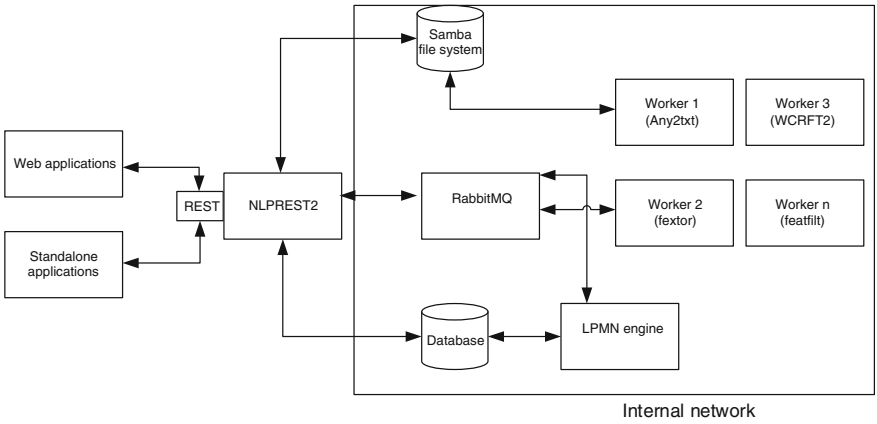


Fig. 4 The engine architecture

```
<tool>:="any2txt" | "cluto"| "featfilt"| …
<worker >:=<tool>|<tool>"("<JSON>")"|"out("<STRING>")"
<pipeline>:= <worker> |  <worker> "|" <pipeline>
<n_input>:="urlzip"|"file"|"files"|"filzezip"|"dspacezip"
<input>:=<n_input>"("<JSON>")"| <n_input>"("<STRING>")
<source>:=<input>|<input><pipeline>"|""dir"
<workflow>:=<source>|<pipeline>
```

Fig. 5 Language Processing Modelling Notation gramma (where not presented rukes: <JSON> defines JSON statement, <URL> proper urls)

4.2 Language Processing Modelling Notation

The graphical representation of language tool workflow (Figs. 1, 2, and 3)—Language Processing Modelling Notation (LPMN) [12], was inspired by the BPMN.[8] The notation allows defining the functionality of complex language tools by combining simple ones. The aim of LPMN is automated processing of the workflow. It is notated in a formal language. The gramma of which is presented in Fig. 5.

The language consists of names of workers (rule <tool> in Fig x), source definition (rule <source>), named outputs ("out") and aggregation statement ("dir", rhombs in Figs. 1, 2 and 3). The exemplar LPMN statement is presented in Fig. 6.

[8] http://www.bpmn.org/.

```
urlzip(http://ws.clarin-pl.eu/public/teksty/ksiazki.zip)|
any2txt|wcrft2|fextor({"features":"base"})|dir|
featfilt({"similarity":"jaccard","filter":"min_df-1 min_tf-1"})|
cluto({"no_clusters":3,"analysis_type":"plottree"})
```

Fig. 6 Example of the LPMN statement

4.3 LPMN Engine

The LPMN engine is the system component responsible for running workflows and informing external applications of processing status. It takes LPMN statements from the RabbitMQ engine, checks LPMN syntax, sends requests to workers by RabbitMQ, waits for a task finish and runs next tasks.

The LPMN engine is implemented in the asynchronous architecture (as discussed in Sect. 3) and communicates with workers in the remote procedure call[9] style. The LPMN engine plays a role of a client and the worker a server. The engine runs only one thread for each LPMN workflow. The processing thread sends requests to workers (first in the pipeline) for all files defined in the LPMN workflow sources. Next, it checks each callback queues one by one with a short timeout (5 ms). If any task is finished, next worker (as defined in the workflow) is called. If all requests are responded and there no more tasks to be executed the LPMN finishes processing and stores ids of resulting output files in the database.

4.4 Web Application

To show a functionality of the engine a web application for clustering workflow defined in Fig. 1 was developed. It is available online at http://ws.clarin-pl.eu/demo2/webstylo.shtml. It is a Single Page Application that uses NLPREST2 web service to run and monitor workflows on a server side.

The application allows loading texts from different sources (local files, Clarin-PL repository[10] or any URL), choosing the features for the description of documents, setting the options for clustering (filtering method, feature, and similarity metric) and running the workflow. The progress of processing is shown to a user. Finally, results (Fig. 7) are presented in graphical forms in a circle and in an interactive tree.

[9] https://www.rabbitmq.com/tutorials/tutorial-six-java.html.

[10] https://clarin-pl.eu/dspace.

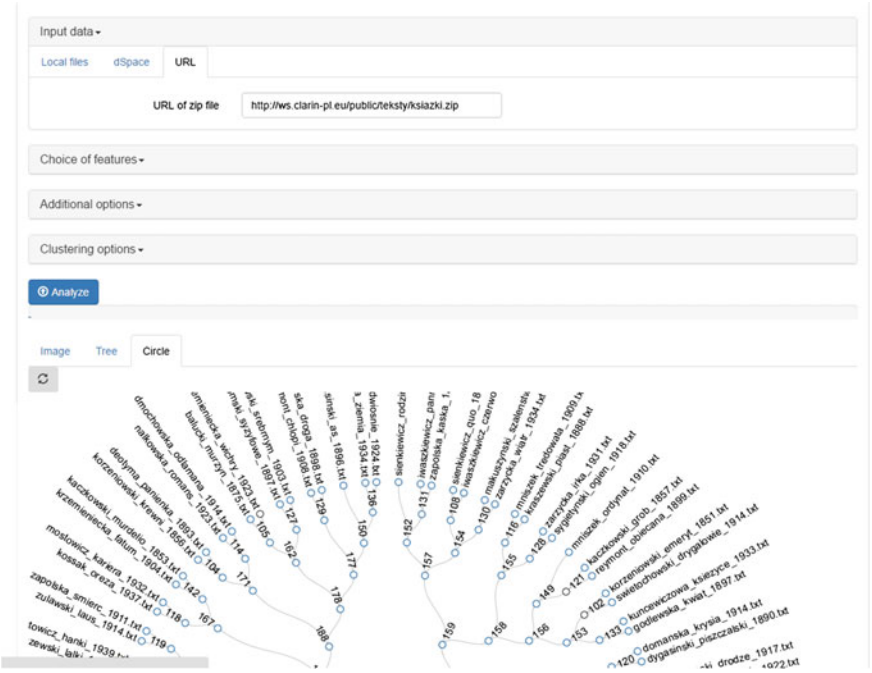


Fig. 7 Clustering texts in Polish—web interface

5 Summary

The paper presented the asynchronous system for clustering and classification of texts in the Polish language. The system allows defining and running flexible workflows of language (like tagger, named entity recognizer, feature extractor) and machine learning tools (like Cluto, SVM). It is a base for developing Web applications, which process texts online.

The efficiency test showed that the system gives an overhead of less than 10 ms per each language and machine learning tool call. The system was tested on corpus with 35 000 files with the LPMN workflow consisted of seven workers. There were no efficiency problems. The LPMN engine was using less than 1 % of CPU, whereas the RabbitMQ middleware more than 200 %. It shows that the described system is able to process very large corpora without any additional time overheads. Of course, the time of processing of the whole corpus depends on the number of workers run for each activity. This number is limited by an amount of available computational resources.

In the next step, we plan to implement the system in a public cloud (like Amazon) and a research computer grid (like PL-Grid). It should allow to process huge corpora on demand using available computational resources.

Work financed as part of the investment in the CLARIN-PL research infrastructure funded by the Polish Ministry of Science and Higher Education

References

1. Allamaraju, S.: RESTful Web Services Cookbook: Solutions for Improving Scalability and Simplicity. O'Reilly Media and Yahoo (2010)
2. Banavar, G., Chandra, T., Strom, R., Sturman, D.: A case for message oriented middle-ware. In: Proceedings of the 13th International Symposium on Distributed Computing, pp. 1–18 (1999)
3. Broda, B., Piasecki, M.: Parallel, massive processing in SuperMatrix—a general tool for distributional semantic analysis of corpora. Int. J. Data Min. Model. Manag. **5**(1), 1–19 (2013)
4. Broda, B., Kędzia, P., Marcińczuk, M., Radziszewski, A., Ramocki, R., Wardyński, A.: Fextor: a feature extraction framework for natural language processing: a case study in word sense disambiguation, relation recognition and anaphora resolution. Stud. Comput. Intel. **458**, 41–62 (2013)
5. Hinrichs, M., Zastrow, T., Hinrichs, E.: WebLicht: Web-based LRT services in a distributed science infrastructure. In: Proceedings of the International Conference on Language Resources and Evaluation, pp. 489–493. European Language Resources Association (2010)
6. Marcinczuk, M., Kocon, J., Janicki, M.: Liner2—a customizable framework for proper names recognition for Polish. Stud. Comput. Intel. **467**, 231–253 (2013)
7. Ogrodniczuk, M., Lenart, M.: A Multi-purpose Online Toolset for NLP Applications. LNCS, vol. 7934, pp. 392–395. Springer, Berlin (2013)
8. Palczynski, M., Walkowiak, T.: Synchronous vs asynchronous processing in high throughput web applications. In: 15th International Conference Reliability and Statistics in Transportation and Communication, pp. 195–202. Transport and Telecommunication Institute, Riga (2015)
9. Piasecki, M., Walkowiak, T., Eder, M.: WebSty—an Open Web-based System for Exploring Stylometric Structures in Document Collections. In: Digital Humanities, accepted abstract, Krakow (2016)
10. Radziszewski, A.: A tiered CRF tagger for Polish, intelligent tools for building a scientific information platform. Stud. Comput. Intel. **467**, 215–230 (2013)
11. Walkowiak, T.: Web based engine for processing and clustering of Polish texts. In: Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, pp. 515–522. Springer (2015)
12. Walkowiak, T., Piasecki M.: Web-based natural language processing workflows for the research infrastructure in humanities. In: 5th Conference of the Japanese Association for Digital Humanities, JADH 2015, pp. 61–63 (2015)
13. Wittenburg, P., et al.: Resource and service centres as the backbone for a sustainable service infrastructure. In: Proceedings of the International Conference on Language Resources and Evaluation, pp. 60–63. European Language Resources Association (2010)
14. Wróblewska, A.: Polish Dependency Bank. Linguist. Issues Lang. Technol. **7**(1) (2012)

Simulation-Based Dependability Analysis of Systems in Multiple Time-Horizons

Tomasz Walkowiak and Dariusz Caban

Abstract When designing and operating any technical system, it is essential to take into account the possible faults that may occur during its operation. Dependability analysis lets us determine the level of redundancy that ensures continuity of service at an economically justified level of assurance. The analysis tends to underemphasize the extremely improbable, simultaneous failures of all or almost all system components. Yet, these situations should not be ignored as their consequences are often disastrous. The use of straightforward simulation in this case is very difficult—there is a huge disparity between the timescales of occurrence of the various types of events (connected with system functioning, fault occurrence and catastrophic breakdowns). The paper presents some experience with applying a unified simulation approach to deal with these multiple time-horizons. The usefulness of the proposed approach is demonstrated in two test studies: a discrete transport system and a web based information system.

Keywords Dependability · Risk analysis · State-transition model · Event-based simulation

1 Introduction

A. Avizienis, J.C. Laprie and B. Randell proposed that dependability be defined as the capability of systems to deliver service that can justifiably be trusted [1]. The definition relates dependability with the functionality of systems, i.e. with their ability to provide the functionality in presence of faults. This is closely related to systems reliability, i.e. the probability of a device/system performing its purpose

T. Walkowiak (✉) · D. Caban
Wrocław University of Science and Technology, Wybrzeże Wyspiańskiego 27,
50-320 Wrocław, Poland
e-mail: tomasz.walkowiak@pwr.edu.pl

D. Caban
e-mail: dariusz.caban@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_47

adequately for the period of time intended under encountered operating conditions [2]. The cited definitions are similar, but with significant differences. These differences in the approach address the problems encountered when trying to define the reliability as applying to computer systems, software, or complex information/management systems. In all these cases the occurrence of faults is strongly related to the currently performed operations. Thus, it requires an approach combining analysis of faults with system operation.

The system dependability encompasses various related issues: availability (readiness for correct service), reliability (continuity of correct service), safety (absence of catastrophic consequences on the users and the environment), security (capability to provide service only to authorized users), confidentiality (absence of unauthorized disclosure of information), integrity (absence of illegal system modifications) and maintainability (ability to undergo repairs and modifications) [1]. All these concepts integrate system operation with various types and effects of faults occurrence. There is a very significant difference in the time scales of the analyzed events: normal operational changes, occurrence of single faults and occurrence of very unlikely events (usually combinations of faults that may have catastrophic consequences).

In the paper, we propose a unified approach to dependability analysis of complex systems that combines system modelling in terms of states and transitions with Monte Carlo simulation [8]. We underscore the problems with assessing hazards connected with very unlikely faults or combinations of faults, requiring special analysis techniques to deal with the disparate time horizons. The approach aims to reuse the same simulation tools both for computing availability/reliability and for risk assessment.

2 State-Transition System Model

A model of complex technical systems needs to encompassing resource allocation, system reconfiguration, multiple modes of operation, etc. All these phenomena can be described in terms of a state-transition model (S-T system model). Some diverse examples of systems analysis, conforming to this system view, are presented in Part 4.

The considered systems realize multiple tasks that arrive on their input. These tasks form the system workload that varies in time. The workload changes can be characterized by some stochastic process.

The system consists of a set of resources used to perform the system input tasks. The execution of each task involves a subset of system resources. Thus, multiple tasks are executed in parallel involving exclusive subsets of the resources. System configuration determines the resources allocated to the execution to the various tasks in the system. Some resources may be idle—not allocated to any task. On the other hand, some tasks may have unsatisfied resource requirements (empty subsets or subsets that do not include all the required functionalities), waiting for execution (queued).

The allocation of resources to tasks is not predetermined. The same task may be realized using different system resources, possibly requiring different time to complete. Once a task completes, all the allocated resources are released to other tasks. System configuration (allocation of resources to tasks) changes when tasks are completed or new tasks are input. It may also change due to internal system events, causing system reconfiguration. All these events (input, output and internal) are further called the **system operation events**. The decisions concerning the allocation of resources to tasks are driven by the system operation events. In the considered model, we abstract from the procedures used to determine the configurations, concentrating on their outcomes.

System operation events occur with different intensities (different mean times between their occurrences). These time differences are small as compared to the times between fault occurrences. Thus, the operational events define the shortest time horizon for system modelling.

The state-transition analysis of a system is based on building a state-transition model of the system: defining the system states and then determining the transitions between the states. In the considered model, the system state includes its configuration (and sometimes additional information taken into consideration by the management system). The transitions reflect the changes in state initiated by the operation events.

In the considered systems, it is assumed that the operational state is defined by the system configuration (a vector of subsets of resources currently allocated to each task). The set of all accessible operational states is denoted as F.

System state must also describe its dependability status: if all the resources are operational or some are down, if their state is observable (taken into consideration by the management system), etc. The changes of the dependability state are caused by **dependability events**: various resource faults and their renewal. There are hardware faults, permanent and transient. Since systems usually dependent on information processing, so there are also malfunctions caused by software faults and human mistakes. Intentional intrusions and vulnerability exploitation are also considered as dependability events. The average time between the occurrences of dependability events is incomparably longer than the time between operational ones. This constitutes another time horizon of the system.

The dependability state of the system is a vector of the dependability states of its resources, whether they are operational, faulty, under repair or isolated. The set of all dependability states of the system is denoted as D. The state space of the system is derived as a subset of the Cartesian product of operational states and dependability ones (after eliminating impossible configurations of resources in a specific dependability state):

$$\mathbf{S} \subset F \times D. \tag{1}$$

The transitions between the system states are initiated by operational and dependability events. When an operational event occurs the configuration is modified. When a resource fails, the corresponding dependability state is modified

according to the mode of the fault. Other coordinates may also be affected due to propagation of the fault and management decisions. Each state transition is described by its from-state and the to-state. The set of all transitions is a subset of the Cartesian product of the system state space:

$$\mathbf{G} = \mathbf{S} \times \mathbf{S}. \tag{2}$$

The occurrence of events, operational and dependability ones, is described by stochastic distributions. Each transition has a corresponding distribution attributed to it. The distribution binds the time the system stays in the transition from-state with the probability of changing to the to-state $P_g(\tau)$. Thus, we have to consider that the state model consists of three components: states set, transitions set and distributions vector.

In most cases, the complex technical systems conform to the ergodic property, i.e. they have stationary distributions of system states probabilities that are attained in due performance times. The probability of every system state P_S does not change in time once the stationary states distribution is reached.

3 System Analysis

3.1 Monte-Carlo Simulation

The S-T model of the system can be analyzed using a number of different methods and tools, especially using graph analysis and linear algebra. There are multiple tools available on market, if one is satisfied with the assumption that all transitions can be described with exponential time distributions (constant intensities). There are two problems to this approach: the huge size of the state space and the disparate time horizons (in effect, very unbalanced transition matrices). In effect, the analytical approach is limited to fairly simple systems.

One of the most versatile solutions is to use Monte-Carlo simulation [8]. It is based on the application of an event-driven simulator, where random operational and dependability events are generated using appropriate pseudo-random generators, to gather statistical data on S-T system performance.

The analysis is performed using a simulator, custom designed for this purpose. It is based on the publicly available SSF simulation engine that provides all the required simulation primitives and frameworks, as well as a convenient modeling language DML [11] for inputting all the system model parameters. By repeating the simulator runs multiple times using the same model parameters, we obtain several independent realizations of the same process (the results differ, since the system model is not deterministic). These are used to build the probabilistic distributions of the results, mainly the average measures.

A significant problem with this approach arises from the differences in time horizons of the various events. The operational events are very frequent in

comparison to fault occurrences. In effect, to obtain a meaningful number of faults initiated transitions, required for statistical reasoning, the simulator is required to process huge numbers of operational events. Thus, Monte Carlo simulation requires much greater computational resources than simply operational simulation. In some cases this overhead is prohibitive and process decomposition must be used.

3.2 Process Decomposition

Process decomposition is a technique of analyzing discrete state transition processes. It is based on combining subsets of system states and separately analyzing the transitions within the subsets and separately those changing the current subset. In case of some processes (e.g. the discrete Markov processes), there are sound mathematical foundations to process decomposition—when is it permissible and what are the likely errors incurred. Unfortunately, in the general case of a state-transition process there are no such foundations, though the approach can still be used as a heuristic technique.

Process decomposition assumes that it is possible to partition the states of the process in such a way that:

- the transitions between the states within each subset are much more probable (occur much more often) than the transitions between the subsets;
- the subprocesses connected with transitions between the states of each subset are ergodic.

The natural decomposition of state space, in case of the considered systems with disparate time horizons, is to group the states with the same dependability state. Let's denote the subset of states with the same dependability state $d_i \in D$ as $\mathbf{S}_{d_i}$.

The S-T process is decomposed into a number of smaller processes corresponding to the above discussed state subsets. For each of them, the stationary distribution of states is denoted as $P_s^{d_i}$ where $s \in \mathbf{S}_{d_i}$. It can be determined using the simulator.

The S-T process that that combines the subprocesses is determined by assuming a state space of D. Transition distributions are determined using the following equation, where d_i and d_j denote the from-state and to-state:

$$P_{d_i,d_j}(\tau) = \sum_{s' \in S_{d_j}} \sum_{s \in S_{d_i}} P_s^{d_i} \cdot P_{s,s'}(\tau). \tag{3}$$

The simulator is modified to incorporate state decomposition as follows. First, the stationary distribution is determined for fully operational system. Then, the transition distributions from it are determined using (3) and the next dependability state is simulated. The corresponding stationary distribution is then determined and the steps are repeated. This is continued until the required statistics are accumulated.

Process decomposition reduces the number of simulation iterations. The greater the disparity of time horizons, the greater reduction in simulation time is possible. Incidentally, the inaccuracies introduced by this technique are also reduced when the time horizons are significantly different.

3.3 Analysis of Unlikely Faults

The technical systems are usually designed to cope with the normal load of work, under usual conditions of operation. This also implies an economically justifiable level of redundancy that ensures uninterrupted operation in case of the expected incidents or normal fluctuations in the workload. It does not address the risks connected with unlikely combinations of events. There is very little likelihood that the simulator will ever analyse a situation, where a number of improbable (and assumed independent) events occurs at the same time.

In most situations, unlikely events can be safely ignored: it is improbable that they ever occur in the system lifecycle. Even slight possibility of a fault is not acceptable if its consequences are catastrophic. This introduces yet another time horizon of events, a time horizon that cannot be handled by process decomposition since the events do not have any meaningful distribution of time between their occurrences.

The analysis of unlikely catastrophic situations is addressed by risk assessment. Risk is defined as a combination of fault likelihood and consequences of its occurrence. Risk analysis is used to ascertain the consequences of an event or decision, when these cannot be a priori determined. The term "risk" is used in decision-making, when the results of the decisions cannot be predicted [13]. The analysis ensures that the risks are identified, likelihood of their occurrence is established and the consequences are evaluated.

There are various alternative approaches to risk assessment [7]. Most popular is the qualitative approach, based on expert opinions [10]. It describes the risk in generic terms of 'high', 'medium' and 'low'. Quantitative analysis uses numeric values to signify similar notions [9]. The risk is proportional to the product of likelihood and consequences:

$$R = LC. \tag{4}$$

Likelihood is very closely related to probability of events occurrence. In this approach, the exponential scale of likelihood is assumed, i.e. the likelihood is proportional to the rank of probability. Let's assume that an unlikely event occurs with a constant intensity λ and its effects prevail for time τ. Then, the probability that at a specific time instant the system is affected by the event is approximately equal to the product of them, and the likelihood is given by equation:

$$L = \alpha + \beta \log(\lambda\tau), \quad \text{if } \lambda\tau \ll 1. \tag{5}$$

The constants α and β are used to ensure the required range of likelihood values.

Likewise, the consequences of the fault are proportional to their duration. Thus, the consequences are assessed by the formula:

$$C = \varphi\tau + \varepsilon. \tag{6}$$

The constants φ and ε are used to scale the consequences adequately (to conform to the 'high', 'low' notions).

The analysis based on Monte Carlo simulation is insufficient to analyse the effects of unlikely faults. The simulator was modified to enable this analysis by introducing manual injection of predetermined unlikely faults (at deterministic time instances). The simulator is then used to predict their effects on system operation. The system is analysed to ascertain its ability to resume normal operation after a catastrophic fault. This is measured by the period of time that elapses before the system is capable of normal operation. The time can be significantly longer than the actual duration of fault. Thus, the simulator provides a prediction of the time τ, used in Eqs. (5) and (6).

4 Case Studies

4.1 Discrete Transportation Systems

The discrete transportation system (DTS) is a model of the transportation system, in which goods are transported by a fleet of vehicles of limited capacity [5, 16, 18]. The flow of goods is discrete, as it corresponds to the movement of the vehicles between the various destinations (or nodes in the travel routes). The system model consists of locations from which goods are collected and to which they are carried and of vehicles travelling between the nodes.

The vehicles are manned by drivers. There is a single central node and a number of local ones. The central node is the only destination of assignments generated at the local nodes. The central node generates goods assignments destined to all the local nodes.

The input tasks are generated independently of each other. Poisson process is used to model it. Each local node has an attribute which determines its characteristic rate of input task generation. The central node is described by an array of assignments rates, one for each local node. There is a fixed time in which each task must be completed. Depending on the nature of the DTS system, this time is fixed by local regulations or is part of the service agreement between the assignees and the transport service provider.

The system makes use of the following resources used to perform tasks: the set of vehicles, the set of vehicle operators (drivers), assigned to vehicles when they

transport goods between the nodes. Whenever a vehicle is assigned to a task, a driver must also be associated to it. Any unallocated driver can be associated with any vehicle. Only one driver at a time is associated with a vehicle. The parameters describing the driver model include the daily working hours limit, maximum uninterrupted driving period and minimum break duration. The system operational events include: input task generation, starting the vehicle route and approaching the destination node. To model different traffic conditions the vehicle speed (determining the route latency) is described by a random value [12].

The DTS resources (drivers and vehicles) used to perform tasks could fail. The vehicles are assumed to break down occasionally, in accordance to their reliability parameters (failure rates). On being repaired (after a random repair time), the vehicles continue the work they were realizing before breakdown. No transloading of the containers is allowed. Drivers get ill or temporarily unavailable. After a prescribed leave of absence they come back to work. Driver illness is modelled as a stochastic process with three categories of illness [18]: short sickness (1 or 2 days), typical illness and protracted disability (longer than 10 days).

Since each task in DTS has a guaranteed time of delivery, we propose to define the performance of DTS as an average ratio of on-time deliveries [3]. Its value depends on the system parameter values. The system analysis method described in part 3 was used to analyze exemplar DTS model (with system model parameters as defined in [5]).

The average ratio of on-time deliveries for different numbers of vehicles and drivers for one dependability state (calculated in the shortest time horizon) is presented in Fig. 1a. Next, the simulation were repeated for different reliability states using the technique described in Sect. 3.2. Results (for the second time horizon of the system) are presented in Fig. 1b. As it could be noticed, the slopes of the average ratio surface are shifted towards greater numbers of vehicles and drivers.

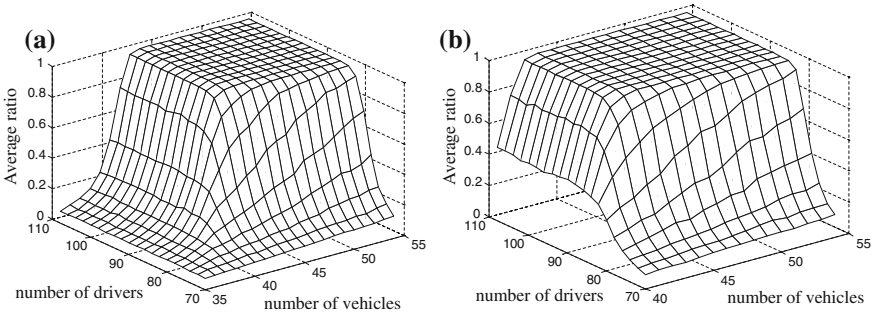


Fig. 1 The average ratio of on-time deliveries for different numbers of vehicles and drivers (**a**) in short time horizon (for only system operation events) and (**b**) in longer time horizon (including dependability events)

Following unlikely catastrophic situation could be identified in case of DTS [3]:

- unforeseeable unavailability of a significant proportion or all drivers (strike, epidemic illness),
- unforeseeable reduction in the number of available or all vehicles,
- rapid increase in the service demand.

Such situation could be analysed following an approach proposed in Sect. 3.3.

4.2 Web Based Systems

We also consider a class of information systems that are based on web interactions, both at the system—human user (client) interface and between the various distributed systems components [4, 6, 15].

We could distinguish different types of input tasks. Each type is described by its chorography [17] that defines a sequence of subtask realized by the service components. A service component is a piece of software that is entirely deployed on a host. Each subtask requires some computational resources to be finished. Moreover, service components are working in different queuing models (as discussed in [14]). A limited amount of tasks could be run in parallel, other subtasks are waiting in a FIFO like queue at each of service component. Each host is described by its processing power. A time required to realize the task is equal to the time required for communication between hosts on which components are located and the time required to realize each subtasks That is equal to a sum of time spend in the queue and the time of processing each subtask. These times depends on the number of processed subtasks, computational resources required to realize subtasks at each of the service components and a hosts' processing power. The described model of the tasks realization defines a short time horizon (from milliseconds to minutes). Using Monte-Carlo approach and an custom designed simulator it is possible to calculate the task response time for a given system configuration and input load characteristic [14].

There are numerous sources of faults in Web based information systems. There are different sources of faults in web based systems. These includes hardware malfunctions (like faults of computer equipment, network devices, power down or failures of internet connection), software bugs, exploitation of software vulnerabilities, malware proliferation, drainage type attacks on system and its infrastructure (such as DDOS) [4].

The faults can either affect a host or only a service running on it. We distinguish the following classes of faults that affect the host: crash—the host cannot process services that are located on it and performance fault. Whenever a fault manifest in a Web based system, whether it is a hardware failure or a security attack, the continuity service could be maintained by moving the service component from the

affected hosts to those that are still operational. It decrease the system performance. The performance loss could be analyzed by mentioned Monte-Carlo simulator.

The analysis of unlikely faults (for example very unprovable combination of several host faults [6]) could be analyzed using a method presented in Sect. 3.

5 Conclusions

A unified approach to dependability analysis of complex systems is presented. It combines system modelling in terms of states and transitions with Monte Carlo simulation and manual injection of faults. It includes methods for dealing with multiple time-horizons of various types occurring in the system, i.e. connected with system functioning, fault occurrence and unlikely faults. Unlike other approaches, this allows one to use the same system models and simulation tools both for performance analysis, dependability/reliability and critical (catastrophic) hazards assessment.

The usefulness of the approach is demonstrated in two diverse test studies: a discrete transport system and a web based information system. In both situations useful, non-trivial results are obtained.

The presented work was supported with research funds of Wroclaw University of Science and Technology.

References

1. Avizienis, A., Laprie, J., Randell B.: Fundamental concepts of dependability. In: Proceedings 3rd IEEE Information Survivability Workshop, Boston, Massachusetts, pp. 7–12 (2000)
2. Barlow, R.E.: Engineering reliability. ASA-SIAM Series on Statistics and Applied Probability (1998)
3. Caban, D., Walkowiak, T.: Risk analysis applied to discrete transportation systems. In: Proceedings of the European Safety and Reliability Conference Advances in Safety, Reliability and Risk Management, ESREL 2011, pp. 2731–2738. CRC Press/Balkema (2011)
4. Caban, D., Walkowiak, T.: Preserving continuity of services exposed to security incidents. In: Proceedings of the Sixth International Conference on Emerging Security Information, Systems and Technologies, SECURWARE, pp. 72–78 (2012)
5. Caban, D., Walkowiak, T.: Reliability analysis of discrete transportation systems using critical states. In: New Results in Dependability and Computer Systems: Proceedings of the 8th International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, pp. 83–92, Springer (2013)
6. Caban, D., Walkowiak, T.: Risk assessment of web based services. In: Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, pp. 97–106, Springer (2015)
7. Cox, L.A.: What's Wrong with Risk Matrices? Risk Analysis, vol. 28, no. 2 (2008)
8. Fishman, G.: Monte Carlo: Concepts, Algorithms, and Applications. Springer, New York (1996)

9. Flyvbjerg, B.: From nobel prize to project management: getting risks right. Proj. Manage. J. **37** (3), 5–15 (2006)
10. Kaplan, S., Garrick, B.J.: On the quantitative definition of risk. Risk Anal. **1**(1), 11–27 (1981)
11. Liu, J.: Parallel Real-time immersive modeling environment (PRIME), scalable simulation framework (SSF), User's manual. Colorado School of Mines Department of Mathematical and Computer Sciences (2006)
12. Surmacz, T., Mazurkiewicz, J., Walkowiak, T.: Universal discrete transport system analysis based on road monitoring data. In: Reliability and Statistics in Transportation and Communication, (RelStat' 12), pp. 370–378, Transport and Telecommunication Institute (2012)
13. Talbot, J., Jakeman, M.: Security Risk Management Body of Knowledge. Wiley Interscience, New Yok (2009)
14. Walkowiak, T.: Web server performance and availability model for simulation. J. Pol. Saf. Reliab. Assoc. **3**(2), 189–196 (2012)
15. Walkowiak, T.: Simulation based availability assessment of services provided by web applications with realistic repair time. Eksploatacja i Niezawodność - Maintenance and Reliability **16**(2), 341–346 (2014)
16. Walkowiak, T., Mazurkiewicz, J.: Functional availability analysis of discrete transport system simulated by SSF tool. Int. J. Crit. Comput. Based Syst. **1**(1–3), 255–266 (2010)
17. Walkowiak, T., Mazurkiewicz, J.: Human resource influence on dependability of discrete transportation systems. In: Dependable Computer Systems, AISC, vol. 97, pp. 271–283. Springer, Heidelberg (2011)
18. Walkowiak, T, Michalska, K.: Functional based reliability analysis of Web based information systems. In: Dependable Computer Systems, pp. 257–269. Springer, Berlin (2011)

Compression Codec Change Mechanisms During a VoIP Call

Radosław Wielemborek, Tomasz Sobieraj and Dariusz Laskowski

Abstract This paper explores the opportunities of using not standardized compression codecs while transmitting the voice data over the existing network infrastructure of VoIP (Voice Over Internet Protocol) providers based on SIP (Session Initiation Protocol) in the Internet. Telecommunication networks offer a large set of services aimed at satisfying the ever increasing needs of users. One of them is a VoIP service implemented through network IP/TCP or IP/UDP stack. One of the most important determinants of this process is the need to use data compression. Codec implementation allows you to manage the data compression degree by changing the width of bandwidth usage and voice quality modeling. These features have priority when using VoIP on mobile devices on pre-paid cards such as smartphones where constraints are the size of the packet, the area signal strength in your area and the computing power of the device. Therefore, it is reasonable to develop new technology that provides flexibility in this regard. First of all, implementation of new codecs and ability to put them into VoIP calls according to SIP signaling standards influence on lowering the cost of deployment and then final costs of calls from subscribers located anywhere around the world with access to the Internet even at low bandwidth. The second advantage is the lower network load affecting the reduction of the costs of its maintenance by the owners of the network or service providers.

Keywords Network · VoIP · Audio compression · Audio codec

R. Wielemborek (✉) · D. Laskowski
Faculty of Electronics, Institute of Telecommunications,
Military University of Technology, Gen. S. Kaliskiego 2 Street,
00-908 Warsaw, Poland
e-mail: rwielemborek7@gmail.com

D. Laskowski
e-mail: dariusz.laskowski@wat.edu.pl

T. Sobieraj
Faculty of Mathematics and Information Science, Warsaw University of Technology,
Koszykowa 75, 00-662 Warsaw, Poland
e-mail: t.sobieraj@mini.pw.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_48

1 Introduction

VoIP service is known to mankind since the end of the 20th century. Initially, its purpose was to reduce the high cost of telephone calls by means of data transmission networks. However, over time it began to supplant the classic services offered by mobile phones (i.e. smartphone, pocket, etc.). At the present time, anyone can use it anywhere, anytime thanks to the popularization of access to the Internet by a number of hotspots, cheaper desktop Internet and data packets added "free of charge" to telephone services. It has a lot of advantages: low costs for registering into operator gate, benefits resulting from the use of IP (for example, restricting data transmission when subscribers "silent") and high mobility. Furthermore, compared with traditional telephony, VoIP introduces the independence of phone operators, free calls within the operator's network, full mobility, low cost infrastructure and possibility to send image or data at no additional cost.

With VoIP is inseparably linked the selection of appropriate codecs for data transmission. The message before send must be compressed to reduce the bandwidth needed to make a connection due to the limitations. It is the most important during VoIP calls using smartphones, where access to the internet data packet is limited and accountable, and signal coverage is not the same strong. Often in poor urban areas it is available only 2G systems. By changing the compression ratio VoIP codec it becomes possible seamless conversation even in those areas where the signal-to-noise ratio is low. Therefore, the choice of the codec is a compromise between high quality of the transmitted voice, the size of data transferred and the device computing power needed to compression. These parameters are very important when the VoIP calls are used in small mobile devices on pre-paid cards like smartphones. The most popular codecs standardized by ITU-T are G.711 in different varieties and G.726 [1]. Each of them has different operating parameters. G.711 provides the best quality of speech however, at the expense of the relatively wide bandwidth resulting from sampling frequency 8 kHz with a resolution of 8 bits per sample. While the codec G. 726 uses encoding at 16, 24 and 36 kbps at a much lower quality. Custom codecs for the purpose of this publication have been named A and B. They are described in the "concept of the research." However, not all VoIP gateways operators provide the ability to establish voice calls using any codec despite the availability the same codecs setting in VoIP terminals. In addition not standardized packet stream may be deleted by intermediate devices while establishing VoIP call using own codec. No less important is the security of transmitted information during a VoIP call. Just like during normal conversations in analog telephony, Internet telephony, there is a risk of call interception and eavesdropping. Custom compression codec significantly increases the level of security (confidentiality) transmitted data due to the fact that its only users know the parameters. The transmission of data in VoIP can be protected using encrypted VPN tunnels (Virtual Private Network), their evolution DMVPN (Dynamic Multipoint Virtual Private Network) and additional components such as virtual Tor (The Onion Router) network [2]. An additional security mechanism, tested in this

paper is codec change to not standardized format during a VoIP call so that the person eavesdropping was not able to understand the content transmitted on the basis of the encoded information. There are two mechanisms presented in Sect. 2. In Sect. 3 it will be served research environment. The survey will be described in Sect. 4, along with its detailed analysis. While in Sect. 5 will be a summary.

2 Payload Change Mechanism During a VoIP Call

The global VoIP provider gateways and intermediate devices may interfere with the content of data transferred in UDP (User Datagram Protocol)/RTP (Real-time Transport Protocol) packets. When in the PT (Payload Type) appears unknown value indicating the A or B codec there is the chance that entire data stream may be rejected and this will result in a lack of connection. One of the ways of research will be an attempt to change the standardized by the ITU-T codec on not standardized codec during the established VoIP call. This change can take place in two ways [3].

The first one uses the REINVITE VoIP message, signaling the remote site connection desire to change the format of data transmitted in the UDP/RTP stream. This method is recommended in the standards describing the VoIP signaling (RFC3261). It allows you to inform the remote site and all intermediaries in transport of data (transit) network devices, which can interfere in the transmitted UDP/RTP packets as VoIP servers, proxy servers, routers, NAT (Network Address Translation), IDS (Intrusion Detection System), IPS (Intrusion Prevention System), NAC (network access control) and NAP (Network Access Protection), firewall the purposive change the format of transmitted data.

In the second way of change the transmitted data type takes place only in a UDP/RTP layer below the application layer (VoIP signaling). Terminal equipment perform codec switch by changing the codec format transmitted in already established UDP/RTP data packets stream without any VoIP signaling messages. Signaling of codec change is performed by assigning new value to the PT field in RTP headers of data packets in the transmitted UDP/RTP stream. In this scenario, it is assumed that the devices which are parties of connection are monitoring the PT field value of RTP header of received packets and supports the case in which this value may change during a call by changing its receiver and transmitter to handle the new data format. This method allows skipping or passing transparently through some network devices acting as a transit for established VoIP session like VoIP servers, routers, NAT, some firewalls but not for checking the packet content devices such as IDS, IPS, NAC, NAP and proxies.

According to the VoIP standards like SIP and H.323 endpoints while switching codec using one of the above methods may operate only on data formats agreed during INVITE—OK—ACK call setup messaging. This introduces the risk that if the first scenario mode switching operation, using REINVITE message does not work due to the removal of these messages about non-standard data formats for transit devices the information will be also removed in the phase of a call setup

messages INVITE—OK. According to VoIP standards endpoint devices can't change the data type in the UDP/RTP stream to not agreed format during the exchange of INVITE—OK—ACK setup messaging. Such a change in the type of data in the UDP/RTP stream can be interpreted by devices such as IDS, IPS, NAC, firewalls, as an attempt to attack and the data stream will be blocked. That is the reason why not standardized codecs will be presented during call initialization.

3 Environmental Research Concept

The test system was located in the capital of Poland—Warsaw. Used VoIP gateways were localized in different countries around the world. For the selection criteria served to server activity and own server infrastructure possession by the operators.

Devices, used for the implementation of the research were software and equipment from renowned manufacturers i.e. Drytek Access Point, Cisco routers and firewalls, LANForge generates stateful network traffic and monitors packets for throughput and correctness, Hewlett Packard servers, 3Com switches, Wireshark and JPerf software.

Many VoIP providers do not have their own servers, by renting them from other VoIP providers and leasing pool accounts. It was not making sense to use them for tests because the results were being the same. The figure below (Fig. 1) illustratively shows the locations of selected gateways.

The test system was consisted of two identical domains connected to the public network—Internet by two public, static IPv4 addresses. Private network addressing

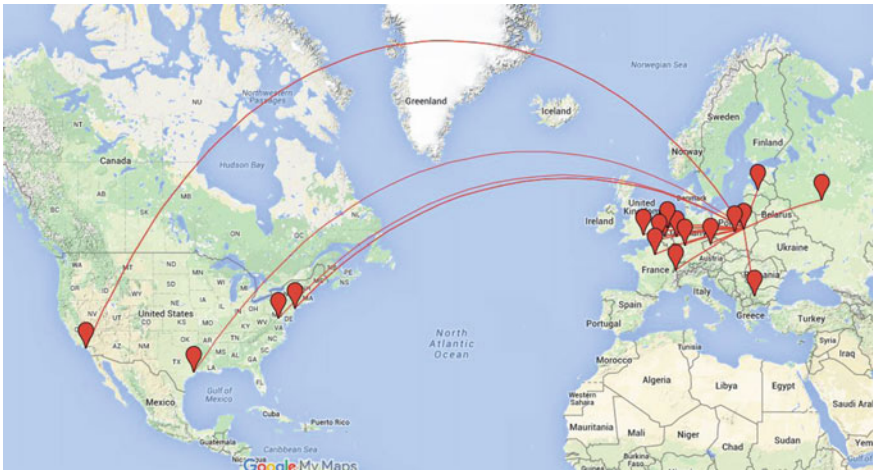


Fig. 1 Examples of routes validating VoIP service with custom codecs

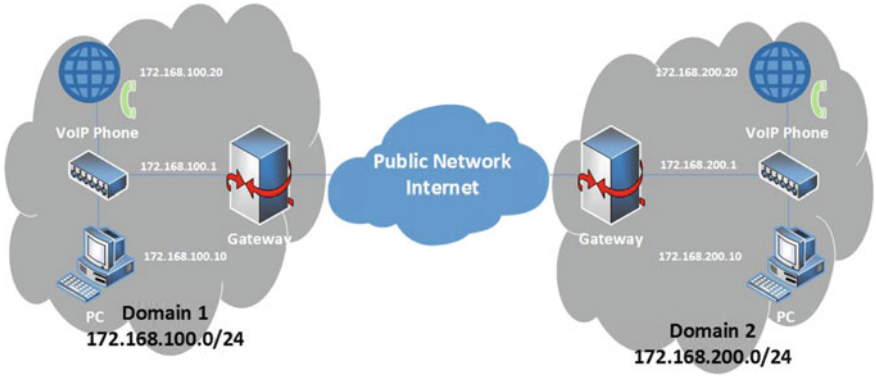


Fig. 2 Testbed for custom codecs

(172.168.x.0/24) differed only in the third octet (100 or 200). Each of the domains consisted of a VoIP terminal, work stations which function was monitoring network traffic and also acting as a log server at IPv4 address 172.168.x.10 and the default gateway providing NAT, DHCP and DNS in these networks which address was 172.168.x.1. Equipment used by the operators like VoIP gateways and other IP network devices act as 'black boxes' (their configuration, manufacturer, model, software and version are not known). This description has been given on the picture (Fig. 2).

For the research purpose there are used standard codecs like G.711 and G.726-16 k and custom payload types described as A and B codecs. The first of these codecs will offer high data compression influencing on bandwidth reduction at intelligible voice quality and low overhead information. The second one is to provide identical features but for video calls. For the research purpose in both domains there are two pseudorandom generators initiated by the same seed. This solution allows generating a specific payload, which enables the assessment of established calls parameters and distortions between source and destination (in VoIP operator network infrastructure and mediate devices) by comparing the data received with sent.

4 Research

These tests are divided into two stages—the first stage consists on selecting public VoIP providers whose servers are active and have their own server infrastructure. Then, it was made an attempt to create subscriber accounts in all of the chosen operators. If the operator make available several VoIP servers and each account can be set up independently there were undertaken such an attempt. After subscriber accounts were created for the indicated servers, an attempt to terminals registration and the trial VoIP connection between the terminals was made. All data were

registered during tests for further analysis of cases, when attempting to register or call establishing was unsuccessful.

The first test concerned the establishing of VoIP calls using preferred initiator codec G.711 A-law defined by the ITU-T and IANA. Then the connection attempts were made using G.726 audio codecs, and custom payload types A and B (described in Sect. 3) using the dynamic range (Dynamic Payload Type) value for the PT field of the RTP header. The aim of the tests carried out in the first stage was to select active public VoIP operators with their own server infrastructure, then check how it is possible to establish connections that use non-standard data formats, using the infrastructure provider.

The second phase of tests were performed only on the VoIP server group selected in the first stage for which the configuration of the subscriber's accounts were successful, VoIP terminals registered on servers as indicated VoIP users, established calls using standard (G.711 A-law) and custom (A and B) codec formats. Further research consisted in call establishing and then changes the codec during its lifetime using both methods described above of codec changing during the VoIP calls by using VoIP REINVITE message and changing the format of the data in the UDP/RTP stream and updating the PT field of the RTP header of transmitted packets. A description of each step of the test procedure is as follows:

VoIP terminals were configured for every VoIP server in order to proper register for the indicated subscriber account and use this server and credentials during call setup messaging.

Test 1: Connection between VoIP terminals is establishing using the G.711 A-law codec, then it is used VoIP signaling REINVITE command to subsequently change codec to the G.726-16 k and then to custom A and B formats.

Test 2: Connection between VoIP terminals is establishing using G.711 A-law codec. Next by changing the data format in the UDP/RTP packets stream and updating the PT field value of RTP header successively codec is changed to G.726-16 k and then to custom A and B formats.

Test 3: Connection between VoIP terminals is established using a G.726 16 k codec, then codec is changed to custom A format using a REINVITE VoIP signaling message.

Test 4: Starting from the connection status obtained in Test 3 return to the basic codec (G.726 16 k) is performed using REINVITE VoIP signaling message.

Test 5: Connection between VoIP terminals is established using a G.726 16 k codec. Next codec is changed to custom A format using the mechanism of changing the data format in the UDP/RTP stream and updating the PT field value of the RTP header.

Test 6: Staring from the connection status obtained in Test 5 return to the basic codec is performed by changing the data format to audio codec G.726 16 k, using the mechanism of changing the data format in the UDP/RTP stream and updating the PT field to appropriate value in RTP header.

Table 1 Test results no. 1

Number of VoIP operator's servers	
Chosen for the research	71
Successful registration	46
Successful established VoIP	26
Unsuccessful account set	8
Unsuccessful registration	17
Unsuccessful call establishment	16
One-sided connection	4
Successful calls using ITU-T codecs (G.711, G.726-16 k)	
Successful REINVITE message used	19
Successful PT UDP/RTP packet change	19
Successful calls using not standardized codecs (A, B)	
Successful REINVITE message used	18
Successful PT UDP/RTP packet change	20
Successful returns to ITU-T codec (G.711)	
Successful REINVITE message used	15
Successful PT UDP/RTP packet change	16

Results of this study are presented in the following tables (Tables 1 and 2) (Fig. 3).

Research results presented above show that the only 1/3 VoIP providers allow for connection with the use of the SIP protocol. Other VoIP gateways do not allow for account setup (8), registration (17), call establishment (16) or voice was one-way (4). Most of the VoIP gateways (20/26) is sufficiently flexible that does not investigate the structure of network traffic and VoIP signaling and does not reject packets with unknown field value shows the type of data compression codec. Method of changing the codec (by change on-the-fly payload) works for one time more than just a REINVITE. This means while changing codecs during VoIP call you can use any data compression codec depending on the prevailing network conditions.

For 26 working servers the payload change using both methods success 19 times on ITU-T codecs, and on the custom data formats (A and B codecs), 18 times using the REINVITE message and 20 times using a switching codec by changing PT field value within UDP/RTP packets. An additional mechanism to return to the previously used codec worked 15 times after using REINVITE switching and 16 times for the use of changing PT field value within UDP/RTP packets. For some gateways calls were established in different codec than declared in setup messaging G.711 or attempt to switch codec during a call resulted in loss of audibility at one or both parties of connection which indicates an incorrect handling of codec switching mechanism by the VoIP operator's server. During the research tested codecs and switching mechanisms behave in a fully consistent and predicable manner. This

Table 2 Test results no. 2

L. p.	Test 1 reinvite				Test 2 PT change				Test 3	Test 4	Test 5	Test 6
	First connection codec	G.726 16k	A	B	First connection codec	G.726 16k	A	B				
1	G.711-Alaw-64K	YES	YES	YES	G.711-Alaw-64K	YES	YES	YES	YES	YES	YES	YES
2		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES
3		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES
4		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES
5		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES
6		YES	YES	YES		YES	YES	YES	YES	YES	YES	NO
7	GSM-06.10	YES**	NO - iLBC	YES*	GSM-06.10	YES**	YES	YES**	NO	NO	NO	NO
8	G.711-Alaw-64K	YES*	YES*	YES*	G.711-Alaw-64K	YES*	YES*	YES*	NO	NO	NO	NO
9	G.729A	NO	NO	NO	G.729A	NO	YES	YES	NO	NO	YES	YES
10	G.711-Alaw-64K	YES	YES	YES	G.711-Alaw-64K	YES	YES	YES	YES	YES	YES	YES
11		YES*	YES	YES*		YES*	YES	YES*	NO	NO	NO	NO
12		YES*	YES**	YES*		YES*	YES**	YES*	NO	NO	NO	NO
13		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES
14		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES
15		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES
16		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES
17		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES
18		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES
19		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES
20		YES	YES	YES		YES	YES	YES	YES	YES	YES	YES

*Without voice

**One way voice

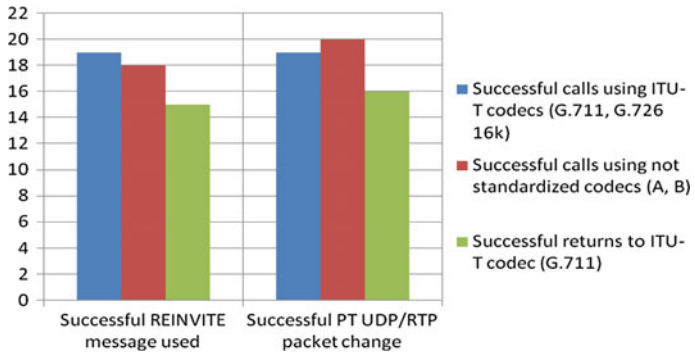


Fig. 3 Test results

demonstrates the high reliability of tested mechanisms. After conducting these tests, it can be assumed that the implementation of new codecs in the field of VoIP technology will not entail the need for changes to the signaling standards, which determine the future work direction. This research and the tested mechanisms will be served into better use of available bandwidth during VoIP calls using low bit rate connections. It will be used by subscribers of VoIP using the Internet on mobile devices running on the pre-paid card where signal coverage is varied, amount of data is limited and the amount of data transferred is important.

5 Conclusions

A study carried out on the pages of this paper was aimed to check the possibility of using custom codecs to establish VoIP call. For this purpose it was used two mechanisms for changing the codec during a call.

The first one use REINVITE VoIP message, it signals remote connection party desire to change the format of transmitted data in the UDP/RTP stream. The second one changes the value of the PT (Payload Type) field in the RTP headers of packets transmitted within UDP/RTP stream. Packet transmission tests with not standardized codecs with compression were conducted using the VoIP operators servers located around the world.

The results show that establishing VoIP calls using not standardized codec is possible and that the used mechanisms fulfill their functions. That can be achieved by time-continuous identification of network environment functionality. Therefore, using the available knowledge and programming tools, one of the numerous, possible to be specified, mathematical dependencies characterized by transparency was presented. It is obvious that its components can be specified for other topologies taking into account the adjustment of weights for the next network architecture tested [4].

References

1. ITU-T standards: G.711. Pulse code modulation (PCM) of voice frequencies, G.726. 40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation (ADPCM), H.323. Packet-based multimedia communications systems
2. RFC3261: SIP: Session Initiation Protocol
3. RFC4566: SDP: Session Description Protocol
4. Lubkowski, P., Laskowski, D.: Test of the Multimedia Services Implementation in Information and Communication Networks, Advances in Intelligent Systems and Computing, vol. 286, pp 325–332. Springer, Switzerland (2014). ISSN 2194-5357

Dependability Model of an Area Monitoring System with Mobile Sensors

Wojciech Zamojski

Abstract An area monitoring system with mobile sensors is investigated. The system is constructed on the base of a team of mobile sensors which are patrolling a fixed area and gathered data are delivered to a central processing station. The system is considered as a services network (a system with SOA architecture). The measure of system efficiency is proposed.

1 Introduction

An area monitoring system with mobile sensors is investigated in this paper. The system is constructed on the base of a team of mobile sensors which are patrolling a fixed area and gathered data are delivered to a central processing station for further handling. Communication between the patrols and the data processing center is realized by a network of transmitter stations (processing nodes providing dedicated services) and communication links. The main task of the system is to update a situation map covering the fixed area. The data are collected by the mobile wireless sensors working on the ad hoc scheme [3]. The communication network of the system has stable structure of nodes and links and works as a wide computer network.

The system's resources (patrols, nets and links) may get their operations disrupted because they are operating in not always friendly environment which can often be a source of failures, faults (bugs) and attacks but the main task of the system (updates of the map) have to be continued. Environmental conditions, changing locations of mobile hosts and structural changes in a wireless network can affect dependability parameters of the system [1, 2, 3, 5, 6].

W. Zamojski (✉)
Faculty of Informatics, Department of Computer Science, Wroclaw School of Information Technology Horyzont, Wejherowska St. 28, 54-239 Wrocław, Poland
e-mail: wojciech.zamojski@pwr.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_49

The system can be considered as a services network [5] or as a system with SOA architecture [3]. Realized tasks engage (dynamic allocation) particular system resources and for any given task there can be several different hardware, software or functional configurations associated with different performance levels (throughput) and different time schedules [6].

The considered and modelled system is similar to many military solutions [4].

2 Area Supervision System

An Area Supervision system is based on a team of mobile patrols, the network of relays (transmitters—access points—Base Stations) and a central processing station. The mobile sensors are monitoring the area and providing results of observation to any available access point. The AS system is observing the area F and forms the current map of the monitored area in the central R.

The mobile wireless network under consideration is presented in Fig. 1. Mobile patrols (diamonds in the figure) are moving within the operation area F and communicate with other users directly or via the Base Stations (an antenna with a dotted oval represents its range) network. The range of mobile sensors is limited (r).

A part of the area F would not be fully covered by the range of access points. Sometimes the mobile patrol is forced to move closer to an appropriate input Base Station in order to send updating message. In consequence the map is created on more or less delayed data. Minimum lag corresponds to the time of direct transmission of the report through the Base Station network. When the patrol has no direct access to the BS network then the patrol has to approach the closest operating input station and the time of this travel is added to the monitoring task time. Of course, the time of the patrol relocation is much longer than time of report transmission across the communication network.

Update of the maps depends on efficient work of patrols (mobile sensors) and on performance and availability of the communication network.

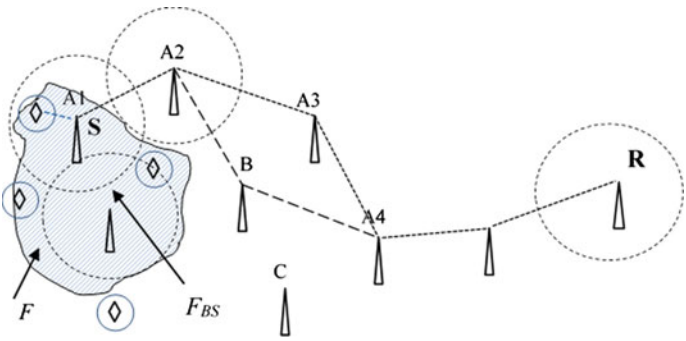


Fig. 1 Example of a mobile wireless network working as Area Supervision system

Mobile sensors can be eliminated (damaged) from the AS system and then renewed. During the renewal time updating processes would be realized via the other patrols, which have to increase their terrestrial monitoring area. Nodes and links of the Base Station network can be eliminated (damaged) too and then they can be restored (renewed) for efficient work after some time. During the station's renewal time the reports from mobile patrols are transferred to other efficiently operating network resources; this means that a new transmission path in the network can be created (reconfigured).

3 Model

The considered Area Supervision system includes:

- wireless mobile patrols—sources of information updating the situation in the monitored area F,
- a communication network based on a fixed grid of base stations within which suitable communication paths are created,
- a headquarter providing updated area maps.

A number of needed patrols (M) may be set arbitrarily or based on the analysis which takes into account terrain and surface quality, possible speed of patrol vehicles, range of transmitters, etc. For example, for the lowland area with unchanging level of moving difficulty the number of required sensors can be estimated as $M \approx F/r$, where r corresponds to a patrol transmitter range.

Base stations provide an interface between the patrols and the communication network. Mobile patrols are "on the move" and their transmitters have a limited range r. In many cases, only a portion of the monitored area (F_{BS} area in the Fig. 1) is in the direct range of the base stations; $F_{BS} \subseteq F$. Patrols outside the direct access area are forced to do appropriate reallocation to the area F_{BS}, which is associated with additional delay time of report sending—τ_D.

If a mobile patrol (S—source) needs to communicate with a receiver (for example with the headquarter R or other patrol) it has to choose an adequate access point (A, B, C on the Fig. 1) and next to create a path between it and the receiver—the paths are dynamically changing during system exploitation.

3.1 The Task of Map Updating

The map updating task of the area J_{Update} is implemented on the basis of the following subtasks:

1. subtask of preparing the current report of the monitoring field—J_{FM}. It is understood as a sum of a monitoring area process and a process of providing current data (the report) to an efficient input station BS_{in}
2. subtask of communication J_{Com}—sending the report from the input station BS_{in} to the station R
3. subtask of generation (calculation) of the current version of the map J_{Gen}.

Time of the map updating task may be evaluated as

$$\tau_{Update} = \tau_{FM} + \tau_{Com} + \tau_{Gen} \tag{1}$$

where

τ_{FM} realization time of the monitoring subtask
τ_{Com} realization time of the communication subtask
τ_{Gen} time of the map generation subtask.

3.1.1 Monitoring Subtask

The monitoring subtask is realized in one of two possible ways;

- a patrol is located inside the F_{BS} area and its report is sent directly to the input station BS_{in}. The duration of the subtasks J_{FM1} can be estimated as $J_{FM1} \approx \tau_A$, where τ_A means time of the direct access to the BS network,
- a patrol is working outside of the F_{BS}, i.e. in the region $F \cap \overline{F_{BS}}$. After preparation of the report the patrol must bring it to the area F_{BS}, i.e. to pass to the appropriate input station. The duration of the report delivering subtask can be estimated as $J_{FM2} \approx \tau_A + \tau_D$, where τ_D means the journey time to the direct access area. Of course, the access time τ_A is disproportionately small compared with the time τ_D and very often $J_{FM2} \approx \tau_D$.

3.1.2 Communication Subtask

The report from the BS_{in} station is transmitted in a communication path established according to computer network routing rules. The time of the communication subtask is evaluated as $\tau_{Com} = T_{Com}^{S-R} = n_N \tau_{Tr} + n_L \tau_{CL}$ where n_N and n_L are numbers of nodes and communication links in the path, τ_{Tr}—time used for tracing procedures in the node and τ_{CL}—transit time between two nodes [7].

The communication path can be configured to bypass overloaded or faulty nodes and communication links, for example according to the rule of neighbourhood stage one [7].

3.1.3 Generation Map Subtask

The subtask of the map generation is beyond interest of the following considerations.

4 Efficiency of the Network

4.1 Efficiency of Area Monitoring

The area F is monitored by M patrols. It is assumed that part of the patrols have a direct access to the BS network and the rest of them have to move up to any station that provides access to the BS network. If a patrol is eliminated from its duties then other patrols should take responsibility of monitoring the whole area F, which means that average size of inspected area by any patrol is changing during exploitation of the AS system.

4.1.1 Reverse Model

So called *reverse model* is used to evaluate an influence of a number of active patrols on characteristics of the area monitoring system.

If the number of monitoring patrols decreases then the time of the inspection area increases according to

$$\tau_{JM}(m(t)) \cong \tau_{JM}(M)\frac{M}{m(t)} \tag{2}$$

where

M	fixed (nominal) number of patrols needed for monitoring the area F
$m(t)$	actual number of patrols used for inspection of the area
$\tau_{JM}(m(t))$	inspection time of area F by $m(t)$ active patrols
$\tau_{JM}(M)$	monitoring time of the area by the fixed number of patrols.

4.1.2 Coefficient of Area Monitoring

Efficiency coefficient describing possibility to monitor the area F by $m(t)$ mobile patrols is introduced as

$$e(t) = \frac{F_m(t)}{F_M} \tag{3}$$

where
F_M assumed area monitored by the nominal M patrols,
$F_{m(t)}$ the area monitored by $m(t)$ actually working patrols

As it was mentioned before two cases are studied;

- the area or a part of it is inspected by patrols that have direct access to the Base Station network, and in this case the efficiency coefficient is evaluated as

$$e_1(t) = \frac{F_m(t)}{F_M} = \frac{F_{DAm}(t)}{F_M} \tag{4}$$

- a part of the area F, rarely the whole area, is monitored by patrols without direct access to the network BS; the patrols have to change their localizations and move up towards the nearest input station, then

$$e_2(t) = \frac{F_M \cap \overline{F_{DAm}(t)}}{F_M} \tag{5}$$

Both coefficients depend on efficient work of the mobile patrols and possibility to access the communication network of the AS system.

4.2 Efficiency of Communication Subsystem

A path of a requested communication subtask is created on the base of network resources. The path consists of the input BS_{in} station, a set of intermediate nodes BS, the receiver node R and required communication links. It is assumed that functionalities of nodes are activated dynamically. In many cases functionalities of intermediate nodes are limited to setting up a communication path understood as pointing to the nearest node which is active and efficiently operating.

The Base Stations are being damaged or destroyed. If any input BS station is put out of order, then some patrols loses theirs access to the communication network and they have to look for other input stations; theirs access time to the network increases greatly and their access time becomes dependent on the potential moving speed of the vehicles. If a base station is eliminated (gets faulty or destroyed) inside the communication network then the communication path is changed (reconfigured) and additional time is added to the time needed for sending the report ($\tau_{Com} = T_{Com}^{S-R}$).

4.3 Efficiency Measure of the AS System

The effectiveness of the Area Supervision system is estimated by the average time of updating the area map which is limited here to subtasks of the monitoring (J_{FM}) and the communication (J_{Com}).

As a result of failures or destruction events the number of monitoring patrols is reduced with probability q_{fail}. When any patrol is out of its duties then its functions are taken over by remaining patrols $M - m_{fail}(t)$, where $m_{fail}(t)$ means the number of non-working patrols at time t. The active efficient patrols have to enlarge their monitoring area and sometimes they need more time to access any input base station. Increase in the monitoring subtask time is modeled by the reverse model

$$\tau_{Jm\,fail\,(t)} \cong \frac{M}{M - m_{fail}(t)} \tau_{JM} \tag{6}$$

where

$m_{fail}(t)$ number of nonworking patrols

$\tau_{J\,M}$ realization time of monitoring task by the nominally M provided patrols

$\tau_{J\,m\,fail\,(t)}$ realization time of monitoring task by the $M - m_{fail}(t)$ efficient operating patrols

For the patrols working inside the F_{DA} their monitoring time is evaluated as

$$\tau_{Jm\,DA} \cong \frac{F_{DA}}{F} \frac{M}{M - m_{fail}(t)} (\tau_{JM} + \tau_A) \tag{7}$$

and for the patrols active out of the area F_{DA}

$$\tau_{Jm\ (F-FDA)} \cong \frac{F - F_{DA}}{F} \frac{M}{M - m_{fail}(t)} (\tau_{JM} + \tau_A + \tau_D) \tag{8}$$

4.3.1 Mean Time of the Monitoring Task

The nominal number of patrols M inspecting the area F together with the F_{DA} area is fixed. The patrols are eliminated with probability q. It is assumed that the damages of patrols are independent events and the duties of damaged (eliminated) patrols are taken by other patrols under the reverse model. If it is additionally assumed that all network stations BS are working efficiently, then the average time of the monitoring task can be estimated as

$$\mathrm{T}_J = \sum_{m_{fail}=0}^{M-1} \left(\frac{M}{M - m_{fail}} \left(\tau_{JM} + \tau_A + \tau_D \left(1 - \frac{F_{DA}}{F} \right) \right) (1 - q)^{M - m_{fail}} q^{m_{fail}} \right) + \tau_{Com} \tag{9}$$

When patrol damage probabilities are sufficiently small (q < 0.01), then the average time T_J can be estimated as

$$T_J \cong \sum_{m_{fail}=0}^{M-1} \left(\frac{M}{M - m_{fail}} \left(\tau_{JM} + \tau_A + \tau_D \left(1 - \frac{F_{DA}}{F} \right) \right) \left(1 - (M - m_{fail}) q^{m_{fail}-1} \right) + \tau_{Com} \right) \tag{10}$$

5 Conclusions

The proposed model of efficiency evaluation of the Supervised Area system can be used for estimation of required nominal numbers of patrols under anticipated risks for both the patrol operation and for the communication network. Destruction processes of patrols and communication network are consequences of unintended random events (failures) and/or are due to the unfriendly actions of third parties. Renewal processes of patrols (introduction of new patrols) and communication network (rebuilding of nodes and links) are skipped in the proposed method of evaluation—they require further studies. Also, analysis of the effectiveness of the communication network should be deeper and should take into account the renewal of stations and communication links [2].

Particular attention needs to be devoted to computational methods which would enable at least an engineering evaluation of selected dependability (efficiency) parameters of the area monitoring system.

References

1. Ekpenyong, M.E., Isabona, J.: Probabilistic link reliability model for wireless communication networks. Int. J. Signal Syst. Control Eng. Appl. **2**, 22–29 (2009)
2. Haboub, R., Ouzzif, M.: Secure and reliable routing in mobile adhoc networks. Int. J. Comput. Sci. Eng. Surv. (IJCSES) **3**(1) (2012)
3. Wang, J.: Exploiting mobility prediction for dependable service composition in wireless mobile ad hoc networks. IEEE Trans. Services Comput. **4**, 44–55 (2011)
4. Yasar, S.: Algorithm design for reliability analysis in mobile communication networks. J. Aeronaut. Space Technol. **3**(1), 29–39 (2007)
5. Zamojski, W.: Dependability of services networks. In: 3d Summer Safety and Reliability Seminars SSARS 2009, July 19–25, 2009, Gdańsk-Sopot, pp. 387–396 (2009)
6. Zamojski, W., Mazurkiewicz, J.: From reliability to system dependability—theory and models. In: Summer Safety and Reliability Seminars, SSARS 2011, Gdańsk-Sopot, Poland, 03–09 July 2011, vol. 1, pp. 223–231

7. Zamojski, W., Sugier, J.: Functional-reliability model of a services system with path reconfiguration ability. In: Zamojski, W., Sugier, J. (eds.) Dependability Problems of Complex Information Systems. Series: Advances in Intelligent Systems and Computing, vol. 307, pp. 167–187. Springer (2014)

Supr: Adaptive Byzantine Fault-Tolerant Replication

Maciej Zbierski

Abstract In the last decade, numerous Byzantine fault-tolerant (BFT) replication protocols have been proposed in the literature. However, practically all of these solutions were designed and optimized only for certain, typically very limited set of environment conditions. Despite previous efforts, no existing BFT replication protocol can guarantee stable and reasonable performance in both correct and faulty environments. In this article we attempt to address this problem by introducing Supr, a novel method for effortlessly combining multiple replication protocols into adaptive BFT solutions, which accommodate to a much wider spectrum of environment conditions than the existing BFT systems. Unlike previous approaches, Supr uses a fine-grained mechanism to monitor the parameters of the execution environment, which enables detecting and counteracting arbitrary faults exhibited in the system. To demonstrate its potential, we use Supr to create a sample BFT solution combining three existing replication protocols, each optimized for different conditions. The performed experiments demonstrate that our approach not only significantly outperforms existing solutions in varying environment conditions, but also does not introduce an observable overhead in stable environments.

Keywords Byzantine fault tolerance · State machine replication · Adaptive BFT · Distributed systems · Dependability

1 Introduction

The last decade has observed an increased interest in creating and deploying world-scale, complex distributed systems. As on-line services and cloud environments become more and more widespread, the demand for guaranteeing their correctness and fault-tolerance has nowadays become high as never before. Many contemporary systems use redundancy to guarantee the correctness of the service

M. Zbierski (✉)
Institute of Computer Science, Warsaw University of Technology, Warsaw, Poland
e-mail: m.zbierski@ii.pw.edu.pl

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_50

even when some server machines crash. However, the recurring incidents reported by the major players in the cloud computing market, including Amazon [1] and Google [8], show that this fault model may not be enough in practical deployments. As a result, the overall interest has recently been shifting towards systems providing a correct service even despite Byzantine (arbitrary) faults. Such solutions can not only tolerate server crashes, but also counteract unpredictable faults, or even malicious attackers. In practice, such model is typically provided through Byzantine fault-tolerant (BFT) replication [3]. In such approach the actual service is deployed on multiple nodes, or replicas, and enhanced with a dedicated coordination protocol to guarantee consistency.

Following the publication of PBFT by Castro and Liskov [3], countless Byzantine fault-tolerant replication protocols have been published in the literature (see for instance [5, 10–12] and references therein). However, each of those solutions has been designed to operate only in certain, very limited set of environment conditions. For instance, PBFT [3] achieves reasonable performance only in local area networks, and HQ [5] operates best for large number of replicas. Furthermore, the majority of these protocols have been optimized for the common case, i.e. fault-free environments, and consequently neglect the performance achieved when some nodes become faulty. For instance, it has previously been shown that practically all contemporary BFT replication protocols are prone to a so called MAC attack [4]. Such vulnerability enables faulty clients to indefinitely disrupt the progress of the whole protocol without being detected, simply by appropriately malforming the authentication codes attached to its request. Even though some BFT replication protocols, such as for instance Aardvark [4], have been designed to guarantee acceptable performance despite in faulty environments, they can exhibit more than 10 times worse throughput than the reference solutions when all nodes are correct. Consequently, despite numerous attempts, none of the existing Byzantine fault-tolerant replication protocols provides acceptable performance in both correct and faulty environments.

In this chapter we attempt to fill this gap by proposing Supr, a novel method for constructing Byzantine fault-tolerant replication systems. Supr views a BFT solution as a composition of self-sufficient replication protocols, which can be independently implemented, tested and replaced. A BFT service constructed using Supr actively monitors the properties of the execution environment and processes the client requests using only the protocol most appropriate for the detected conditions. As a result, Supr can be used to create solutions optimized for much broader range of environment conditions than the existing approaches. In order to demonstrate how the method introduced in this article could be used in practice, we use Supr to create Dali, the first BFT replication protocol which provides high performance in correct environments and still guarantees progress during MAC attacks. The conducted experiments demonstrate that Dali not only significantly outperforms existing BFT replication protocols in varying environment conditions, but also does not introduce an observable overhead in stable environments.

2 Related Work

Several existing BFT replication protocols have distinguished both fast and recovery sub-protocols in order to improve performance in optimistic case, and guarantee liveness in unfavorable conditions respectively. Zyzzyva [11] applies speculative execution whenever all replicas are correct and switches to a slower variant committing the requests only when some claims from the clients have been received. HQ [5] deploys a lightweight quorum-based protocol in favorable conditions and switches to PBFT when contention is detected. CheapBFT [10] uses an optimistic protocol requiring less replicas, but falls back to MinBFT [12] when faults are detected. However, in all these solutions the sub-protocols are tightly coupled with each other, thus increasing the effort required for proper implementation, testing and maintenance. Contrarily, Supr is the first generic approach for creating BFT solutions where each sub-protocol is entirely independent, and can be implemented, proven and replaced irrespectively of the others.

The concept of constructing BFT solutions comprised of multiple protocol instances has initially been proposed in Abstract [9]. Apart from acting as a traditional BFT protocol, each Abstract instance can decide to abort a client request whenever it cannot provide progress. In such situation, the aborted request is relayed to a different instance according to a predefined order. However, unlike Supr, Abstract does not proactively change the instance processing client requests to improve the achieved performance based on the detected environment conditions.

Bahsoun et al. [2] have recently proposed Adapt, a derivative of Abstract, which does not require a predefined order in which the instances are aborted. Instead, Adapt monitors the performance of the execution environment, such as the obtained throughput or size of incoming requests, and uses these parameters to select the protocol instance optimal for the detected environment conditions. However, contrary to Supr, Adapt struggles to provide acceptable performance under arbitrary faults, as the execution environment monitoring is not directly coupled with protocol instances. Furthermore, both Abstract and Adapt are likely to involve higher implementation and maintenance costs than Supr due to possible hidden dependencies between the sub-protocol instances. Finally, although Adapt does not require a predefined policy, but uses machine learning to determine the most appropriate instance, this approach can also be applied in Supr without an excessive effort.

3 Supr Architecture

This section introduces and describes Supr (sub-protocols), a new approach for effortlessly defining and implementing complex Byzantine fault-tolerant protocols. Each BFT solution created using the proposed method is viewed as a composition of so called Supr instances. Every instance represents an independent state machine

replication protocol, which processes client requests and generates responses, much like existing BFT protocols. At any moment exactly one instance is active, which means it is designated to consume the client requests and process them according to the underlying protocol specific for that instance. The non-active instances on the other hand do not receive nor execute new client requests, although they still can communicate with their corresponding instances on other nodes.

In order to oversee execution, Supr deploys on each replica an additional supervisor module, which is shared between all protocol instances. The supervisor acts as a multiplexer, intercepting all incoming messages and forwarding them to the corresponding Supr instance. Apart from that, the supervisor collects the information about the properties of the execution environment, such as the observed throughput, average request size in previous batches, etc. These parameters are reported directly by both active and inactive protocol instances. Additionally, instances report to the supervisor any issues regarding the protocol execution. These can include, but are not limited to, faults suspected in the system, observations about any suspicious behavior or notices concerning insufficient progress. Finally, the supervisor maintains and manages a transition policy, which defines how the observations reported by the sub-protocols should affect the selection of the active instance. The rules included in the policy should typically represent both the design goals and the properties of every Supr instance, such as unfavorable conditions, additional assumptions, etc.

The supervisor constantly monitors the information obtained from the sub-protocols, and by confronting them with the policy, it might decide to change the active instance by initiating a transition. When that happens, the supervisor selects the new active instance based on the transition policy and enqueues any subsequent requests received from the clients, instead of relying them to Supr instances. Immediately after all the requests received by the previously active instance have been processed, the supervisor finishes the transition by relaying the enqueued client messages to the new active instance.

The second module deployed on each replica and shared between all Supr instances is the sequencer, responsible for assigning unique identifiers to incoming requests. Instead of using its own protocol-specific method, every instance is required to query the sequencer module whenever it wishes to assign a client request with a unique identifier. This approach guarantees that the requests are ordered between multiple instances, which facilitates transitions between the protocols. Additionally, the sequencer is responsible for maintaining an identifier cache, enabling detecting duplicate client requests. The communication pattern between Supr modules located on a single node, including n sub-protocol instances, is depicted in Fig. 1. The diagram illustrates the process of message multiplexing performed by the supervisor and communication between the active instance and the sequencer module, according to the description presented above.

The process of converting existing implementations of traditional BFT replication protocols into Supr instances involves modifying the protocol to communicate with the modules described above and in most cases is straightforward and easy to perform. Firstly, the code responsible for assigning unique identifiers to incoming

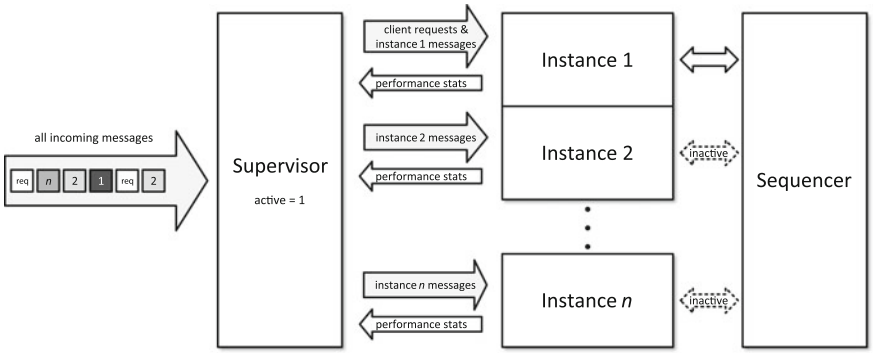


Fig. 1 Communication pattern between Supr modules located on a single replica

requests needs to be replaced with a query to the shared sequencer. Secondly, the implementation of the BFT protocol has to be enhanced to relay progress statistics to the supervisor wherever applicable, either periodically or on demand. The exact location where such functionality should be implemented depends on the properties of the instance and the choice of factors triggering transitions to other instances. Finally, the messages exchanged by each Supr instance should be enhanced with a field unique to that instance. This property is used by the supervisor to forward the received messages to their appropriate destinations, although in some cases this might not be necessary, for instance when the messages exchanged by the adapted protocol already differ somehow from the messages issued by the other instances.

4 Dali Protocol

In order to demonstrate how the approach introduced in this article can be applied in practice, this section uses Supr to create Dali, the first Byzantine fault-tolerant replication protocol which survives MAC attack, and at the same time does not degrade the performance achieved in correct environments. Dali consists of three Supr instances, each implementing an existing BFT replication protocol optimized for different environment conditions. The remainder of this section first provides a brief description of these protocols, and then introduces the corresponding transition policy used by the supervisor.

The first Dali instance uses PBFT [3], a protocol considered as a baseline solution in the field of Byzantine fault-tolerant replication. PBFT designates a single replica to perform the role of the primary, i.e. coordinate protocol execution by imposing a total ordering on client requests. If the remaining servers suspect that the current primary is faulty, they initiate a vote to elect among themselves the replica that will perform the role of the next primary. A similar approach is followed in the large majority of contemporary BFT replication solutions, including

the protocols used in the remaining Dali instances. PBFT uses a traditional three-round consensus to guarantee that the identifier assigned by the primary to a client request is correct. This means that PBFT needs to perform three all-to-all communication rounds between the replicas before each client request can be executed. To guarantee communication integrity, the exchanged messages are enhanced with message authentication codes (MAC).

The second Dali instance implements Zyzzyva [11], a protocol decreasing replication costs through speculation. Unlike PBFT, Zyzzyva optimistically executes client requests without prior consensus. When all replicas are correct, this requires only one communication round to be performed between the servers. However, if the clients are unable to receive a consistent response from *all* replicas before an assumed timeout, potentially because some nodes are faulty, they issue claims addressed to the replicas. These are subsequently used to establish a consistent state on all nodes and reach a consensus on a valid response to the corresponding client requests. Consequently, even though Zyzzyva can provide very high throughput and low latency in correct environments, the additional communication rounds and the delay imposed by the clients can significantly degrade the achieved performance whenever some nodes are faulty. Much like in PBFT, the exchanged messages are authenticated with MACs.

Finally, the third Dali instance deploys Aardvark [4], which, unlike both PBFT and Zyzzyva, guarantees progress during MAC attacks. Although Aardvark bases on the same three-round consensus as PBFT, it uses a combination of MACs and public-key signatures to authenticate the exchanged messages. As a result, the clients can no longer generate incorrect message authentication codes without being detected, since public-key signatures provide transferable authentication. However, at the same time this approach degrades the performance achieved by the protocol, since public-key signatures are more than an order of magnitude slower than message authentication codes [3].

As has been already observed in the literature, each of the protocols presented above performs best in different environment conditions. Zyzzyva provides the best throughput and latency in correct environments, but can be expected to perform worse than PBFT in unfavorable conditions, i.e. when some replicas or clients are faulty [6]. Furthermore, while only Aardvark is resistant to MAC attacks, it provides the worst performance in correct environments due to the additional cost of signature verification [4]. Consequently, when designing Dali, we have reflected these properties in its transition policy. As a result, Dali supervisor select Aardvark whenever a MAC attack is detected, chooses Zyzzyva if no faults are observed, and in remaining situations switches to PBFT.

Each Dali instance monitors the protocol execution and informs the supervisor about the observed anomalies. Dali transition policy uses two main factors to determine which instance should be activated by the supervisor. The first one is the frequency of Zyzzyva-specific claims received from the clients and their relation to the overall number of recently executed requests. When no client claims have been received for an assumed amount of time, all replicas can be considered correct, and an optimistic execution of Zyzzyva can be used to increase the overall performance.

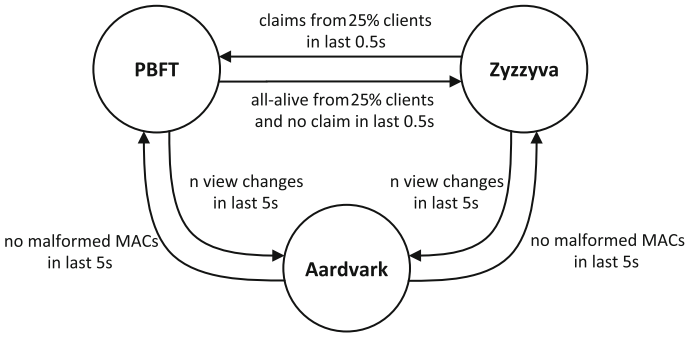


Fig. 2 A transition graph for Dali

The second decision factor is the frequency of primary changes triggered by an incorrectly authenticated request. This is because frequent and recurring primary changes typically indicate a MAC attack. The transition policy used in Dali is presented in Fig. 2. The quantitative parameters of the policy have been selected experimentally to minimize the amount of time required to react to changes in environment conditions, and at the same time prevent premature transitions to less resilient instances in situations where the faults are not exhibited only temporarily. Additionally, after each observed premature transition the time intervals used by the policy are additionally doubled to prevent such situations in the future.

5 Experiments and Evaluation

In order to verify our approach we have created stand-alone implementations of PBFT, Zyzzyva and Aardvark, and subsequently adapted them as Dali instances using the Supr approach introduced in this article. Additionally, each implemented protocol has been enhanced to take advantage of the cost-aware batching optimization introduced by the author [13]. The supervisor module has been setup to select active instances according to the transition policy described in the previous section. The test environment consisted of four servers, each equipped with a 3.4 GHz processor, 4 GB RAM and a Gigabit Ethernet controller, plus an additional machine hosting the client processes, all connected to the same local area network.

The performed experiments compare the throughput achieved by Dali and its reference protocols and analyze how Dali reacts to changes in environment conditions. Throughout the tests we consider throughput as the maximum number of client requests per second that could be processed by the service replicated using the respective protocol. The experiments do not directly compare Supr to the other existing adaptive BFT approaches, such as Abstract [9] or Adapt [2], as they would have to be additionally modified to implement a protocol with a functionality

similar to Dali, since none of them is capable of sufficiently tolerating faults of malicious origin.

The goal of the first experiment was to analyze the performance achieved by Dali running PBFT and Zyzzyva instances, and estimate the cost of performing transitions between them. The experiment started in a correct environment, where all replicas processed incoming requests according to respective protocols. At a certain moment, one of the replicas was manually suspended to simulate a node failure. After some time, the failed replica was restarted, and it subsequently resumed processing client requests. The throughput achieved by the considered protocols throughout the duration of the experiment is presented in Fig. 3. While Dali started the experiment in Zyzzyva instance, the supervisor initiated a transition to PBFT after the replica failure has been detected, i.e. immediately after enough client claims have been received. Similarly, after the failed replica was restarted and the supervisor was no longer receiving client claims, it changed the active instance back to Zyzzyva.

The throughput achieved by Dali has demonstrated to be practically indistinguishable from its reference protocols in both analyzed environment conditions, with the maximum difference below 5 %, and an average around 1 %. Additionally, Dali has achieved the highest average throughput of all protocols despite the change in environment conditions, with an exception of Zyzzyva for requests with large payloads. However, what cannot be observed in the figure, is that in such case Zyzzyva provided around 50 % higher request latency than Dali. While we have configured Dali policy to remain in PBFT in these conditions, in environments where the high throughput is the most important factor, the policy could be further modified in favor of Zyzzyva. The time required to perform a transition from Zyzzyva to PBFT has been dominated by the delay imposed by the clients and has demonstrated to be very similar to the one observed for Zyzzyva alone. Finally, although Dali requires more time to switch back to Zyzzyva in order to prevent

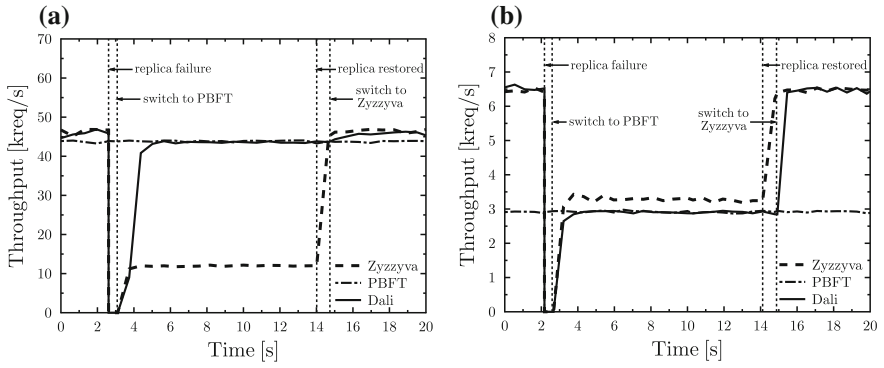


Fig. 3 Comparison of changes in throughput imposed by a single faulty replica. **a** Payload = 0 kB, **b** payload = 4 kB

premature transitions when the replica is still faulty, the switching itself does not exceed 1 s and still remains at an acceptable level.

The second experiment aimed at analyzing the behavior exhibited by Dali and the reference protocols during a MAC attack. The test started in a fault-free environment, and at a certain moment a single client started periodically issuing requests with malformed MACs [4]. After some time the faulty client has been turned off and the execution environment again became correct. Figure 4 presents the throughput achieved by the analyzed protocols throughout the test. While Dali started the experiment in Zyzzyva instance, as soon as its supervisor has detected the MAC attack, exhibited by four consecutive primary changes, it initiated a transition to Aardvark. Subsequently, the supervisor switched back to Zyzzyva after no malformed requests had been received for 5 consecutive seconds, as defined by the transition policy.

During the MAC attack both Zyzzyva and PBFT were unable to process any incoming requests. This is the result of recurring primary changes triggered by the replicas due to incorrect message authentication codes generated by the faulty client. Analogically, while Aardvark maintained progress during MAC attack, it exhibited even more than 10 times worse throughput than the remaining protocols whenever the environment was correct. On the other hand, Dali both provided high throughput in the fault-free environment and exhibited performance similar to Aardvark during the MAC attack. Although the time spent on performing transitions was slightly larger than in the previous experiment, it oscillated around 1 s and still be considered acceptable. Finally, as described earlier, the additional 5 s delay has been introduced by the policy after the end of the MAC attack to provide a trade-off between the achieved performance and the cost of a premature transition. However, this delay could be reduced to the level observed for the other transitions, provided the supervisor is additionally enhanced with appropriate client blacklisting mechanisms [4].

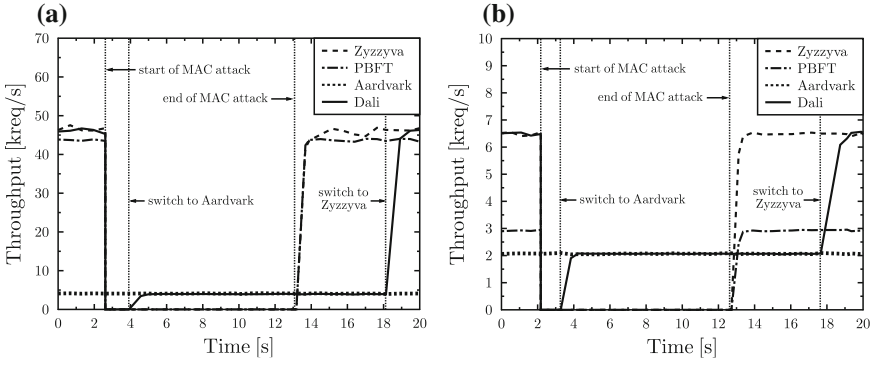


Fig. 4 Comparison of changes in throughput imposed by a faulty client issuing malformed MACs. **a** Payload = 0 kB, **b** payload = 4 kB

6 Conclusion

In this article we have presented Supr, a new method for creating adaptive Byzantine fault-tolerant replicated services. Unlike previous approaches, Supr determines the properties of the execution environment based on the notifications obtained directly from sub-protocol instances. This way the resulting solutions can react to more subtle environment changes and counteract less evident performance degradation caused by arbitrary faults. This makes Supr the first generic method for creating Byzantine fault-tolerant solutions which guarantee high performance in both correct and faulty environments. Furthermore, not only transforming existing protocols to Supr instances requires far less effort than in previous approaches, but also the resulting solution is easier to modify and reason about.

Additionally, to demonstrate that the proposed method can be used in practice, we have used Supr to create Dali, the first Byzantine fault-tolerant solution which provides performance competitive with contemporary BFT replication protocols in correct environments, and at the same time survives the MAC attack [4]. In the performed experiments Dali has demonstrated to easily outperform the other analyzed protocols under varying environment conditions. Additionally, while operating within a single sub-protocol instance, Dali has introduced no observable overhead over its reference protocols.

However, it is worth noting that Dali is by no means the only adaptive BFT solution that could be created using Supr. In the future, we plan to extend Dali with additional sub-protocols to make it adapt even better to different environment conditions. Additionally, we intend to utilize Supr to create a commercial-grade BFT replicated service and use it to enhance the scope of performed experiments, including the additional tests with contemporary fault injection techniques [7].

References

1. Amazon: Summary of the Amazon DynamoDB service disruption and related impacts in the US-East region. https://aws.amazon.com/message/5467D2/. Accessed September 2015
2. Bahsoun, J.P., Guerraoui, R., Shoker, A.: Making BFT protocols really adaptive. In: 2015 IEEE International Parallel and Distributed Processing Symposium (IPDPS), pp. 904–913. IEEE (2015)
3. Castro, M., Liskov, B.: Practical Byzantine fault tolerance. In: Proceedings of the Third Symposium on Operating Systems Design and Implementation, OSDI 1999, pp. 173–186. USENIX Association, Berkeley (1999)
4. Clement, A., Wong, E., Alvisi, L., Dahlin, M., Marchetti, M.: Making Byzantine fault tolerant systems tolerate Byzantine faults. In: Proceedings of the 6th USENIX Symposium on Networked Systems Design and Implementation, pp. 153–168, NSDI'09 (2009)
5. Cowling, J., Myers, D., Liskov, B., Rodrigues, R., Shrira, L.: HQ replication: a hybrid quorum protocol for Byzantine fault tolerance. In: Proceedings of the 7th Symposium on Operating Systems Design and Implementation, pp. 177–190, OSDI'06. USENIX Association, Berkeley, CA, USA (2006)

6. Duan, S., Peisert, S., Levitt, K.N.: hBFT: Speculative Byzantine fault tolerance with minimum cost. IEEE Trans. Dependable Sec. Comput. **12**(1), 58–70 (2015)
7. Gawkowski, P., Rutkowski, T., Sosnowski, J.: Improving fault handling software techniques. In: 2010 IEEE 16th International On-Line Testing Symposium, pp. 197–199. IEEE (2010)
8. Google: Today's outage for several Google services. https://googleblog.blogspot.com/2014/01/todays-outage-for-several-google.html. Accessed January 2014
9. Guerraoui, R., Knežević, N., Quéma, V., Vukolić, M.: The next 700 BFT protocols. In: Proceedings of the 5th European Conference on Computer Systems, EuroSys 2010, pp. 363–376 (2010)
10. Kapitza, R., Behl, J., Cachin, C., Distler, T., Kuhnle, S., Mohammadi, S.V., Schröder-Preikschat, W., Stengel, K.: CheapBFT: resource-efficient Byzantine fault tolerance. In: Proceedings of the EuroSys 2012 Conference (EuroSys 2012), pp. 295–308 (2012)
11. Kotla, R., Clement, A., Wong, E., Alvisi, L., Dahlin, M.: Zyzzyva: Speculative Byzantine fault tolerance. In: Symposium on Operating Systems Principles, SOSP (2007)
12. Veronese, G.S., Correia, M., Bessani, A.N., Lung, L.C., Verissimo, P.: Efficient Byzantine fault tolerance. IEEE Trans. Comput. **62**(1), 16–30 (2013)
13. Zbierski, M.: Cost-aware request batching for Byzantine fault-tolerant replication. In: Theory and Engineering of Complex Systems and Dependability. Proceedings of the Tenth International Conference on Dependability and Complex Systems DepCoS-RELCOMEX, June 29–July 3 2015, Brunów, Poland, pp. 583–592. Springer (2015)

Flood Risk Assessment from Flash Floods in Bodva River Basin, Slovakia

Martina Zeleňáková, Lenka Gaňová, Pavol Purcz, Ladislav Satrapa, Martin Horský and Vlasta Ondrejka Harbuľáková

Abstract The main objective of flood management as well as the entire management cycle is regulated by Directive of the European Parliament and of the Council 2007/60/EC on the assessment and management of flood risks. The paper deals with an application of preliminary flood risk assessment, particularly flood risk assessment from flash floods in the Bodva river basin, which is situated in the south of the Slovak Republic. The aim of the preliminary assessment of flood risk from flash floods is determining the hazard and the vulnerability of assessed area. The result of identification of a hazard is the determination of critical points in the basin and their contributing surfaces on the basis of the geometric and physiogeographic characteristics of the contributing surfaces. Vulnerability in the study area is determined on the basis of the type and density of the built-up area. The resulting flood risk is stated as a moderate risk. The scope and extremity of flood episodes point to the need to build a comprehensive system of flood protection measures in potential flood areas.

M. Zeleňáková · L. Gaňová · V.O. Harbuľáková (✉)
Department of Environmental Engineering, Faculty of Civil Engineering, Technical University of Košice, Vysokoškolská 4, 042 00 Kosice, Slovakia
e-mail: vlasta.harbulakova@tuke.sk

M. Zeleňáková
e-mail: martina.zelenakova@tuke.sk

L. Gaňová
e-mail: lenka.ganova@tuke.sk

P. Purcz
Department of Applied Mathematics, Faculty of Civil Engineering, Technical University of Kosice, Vysokoškolská 4, 042 00 Kosice, Slovakia
e-mail: pavol.purcz@tuke.sk

L. Satrapa · M. Horský
Department of Hydraulic Structures, Faculty of Civil Engineering, Czech Technical University in Praha, Thákurova 7, 166 29 Prague, Czech Republic
e-mail: satrapa@fsv.cvut.cz

M. Horský
e-mail: martin.horsky@fsv.cvut.cz

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*, Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_51

Keywords AISC · MS Word 2007 · Multi-authored volume · Formatting

1 Introduction

The increase in damage due to natural disasters is directly related to the number of people who live and work in hazardous areas and who continuously accumulate assets [1, 2]. Land-use planning authorities therefore have to manage effectively the establishment and development of settlements in flood-prone areas in order to prevent further increase in vulnerable assets [3, 4]. Flood risk analysis provides a rational basis for prioritizing resources and management actions. Risk analysis can take many forms, from informal methods of risk ranking and risk matrices to fully quantified analysis [5, 6]. It is important to keep in mind that a flood nowdays is expected to bring about a whole gamut of consequences. The costs of damages caused by extreme weather events (among which floods are a major category) have exhibited a rapid upward trend world-wide. The scope and extremity of flood episodes point to the need to build a comprehensive system of flood protection measures in potential flood areas.

Geographical information systems and multicriteria analysis (MCA) methods have been applied in several studies in flood risk assessment [7–12].

The paper is focused on preliminary flood risk assessment of flash floods in Bodva river basin, south of Slovakia. The task is to obtain knowledge on the spatial variability of flood risk from flash floods and in doing so supplement a preliminary flood risk assessment already conducted in 2011 for the purpose of proposing suitable flood mitigation measures for reducing the risk found. Thus a process for managing risk in locations endangered by flooding is secured. The added value of this work versus the preliminary flood risk assessment already conducted consists in the supplemental assessment of flood risk from flash floods and the fact that flood risk is not perceived only as a function of flood hazard but is also understood as a combination of flood hazard and vulnerability. The result of identification of a hazard is the determination and selection of critical points and their contributing surfaces, determined by the authors on the basis of selected basic geometrical and physiogeographical characteristics of the contributing surfaces.

Among the factors influencing flow and crucial in regard to the origin and size of flash floods, very intensive torrential rains play a key, primary role. The origin of flash floods is secondarily influenced by local or regional physiogeographical and hydrological conditions, and the hydrological conditions which can through torrential rains further accelerate or moderate the initiated process [13]. According to Acreman and Sinclair [14] the type and speed of hydrological processes occurring in the interior of a basin are determined by its natural environment and its characteristics. The mentioned authors assume that the dominant conditioning factors can also be represented by a small number of variables. This may be the surface and shape of the basin, the incline of its slopes, the length of the flow, the incline of the flow, the density of the river network, a dissection of the relief, precipitation, the

soil cover, the retention capacity of the basin, the landscape vegetation cover, its fragmentation and others. At the same time we could further specify each one, e.g. precipitation—by duration, intensity and frequency; soils—by character, type, depth, initial dampness (the actual amount of water in the soil), infiltration capacity, retention capacity, etc.

In the majority of scientific works [11, 12] etc., variables (parameters) are accepted for practical reasons which can be easily read (or measured) from existing maps and databases.

The aim of the paper is to apply a generally usable methodology, based on flood causal factors, realized in the GIS environment, the result of which will be the determination of areas with the occurrence of flood risk from flash floods.

2 Materials and Methods

In the following sub-chapters the need for assessment of flood risk from flash floods, a description of the methodology as well as an assessment of this flood risk are substantiated.

A foundation was the existing "Methodological instructions for identification of critical points", which was prepared in 2009 within the project "Assessment of floods in June and July 2009 on the territory of the Czech Republic" by the T.G. Masaryk Water Research Institute in Brno.

2.1 Identification of Risk

Knowledge on the material and spatial occurrence of flood situations and knowledge of the reasons that condition the increased frequency or regional differences in the occurrence of floods are an essential foundation for accepting effective and sustainable preventive flood mitigation measures.

2.1.1 Determining of Critical Points and Their Contributing Surfaces

A critical point is stipulated as a point of intersection of the borders of built-up territory of an urban area with the linear path of concentrated surface flow. For each critical point a contributing surface, so-called component basins, is generated the closing profile of which is formed by the relevant critical point. In practice this means that a surface flow of precipitation which falls on the territory of the contributing surface, flows to the profile of the critical point and continues further to the urban area, where it represents a danger for people and their property [15].

2.1.2 Selection and Characterization of Causal Factors

For this methodology causal factors are selected on the basis of those geophysical characteristics of basins which determine the character and course of the flash floods [16]: the surface of a basin, the slope of the basin, pedological relations, climatic relations—total precipitation (maximum daily total of precipitation in mm with probability of repeating once in 100 years; in Slovakia this is from 70 to 180 mm (70–180 l per m^2)), land use.

2.1.3 Defining of Contributing Surfaces

This step deals with the process of defining the contributing surfaces or the segments of a basin which are crucial from the viewpoint of the creation of a concentrated surface flow from flash floods and have adverse effects on built-up urban areas. On the basis of the previous literature as well as expert research, the following criteria have been selected as being crucial in relation to flash floods:

- relative value of the size (area) contributing surface (0.2–40 km^2),
- average slope of the contributing surface (≥5 %),
- share of arable land on the contributing surface (≥40 %),
- share of heavy soils on the contributing surface,
- relative value of the sum of 1-day precipitation with a period of repeating of 100 years on the contributing surface.

The criterion H which represents a combination of geometric and physiogeographical factors (hazard) was calculated according to the following Eq. (1):

$$\mathrm{H} = (w_1 \times P) + (w_2 \times A_{CS}) + (w_3 \times S) + (w_4 \times AL) + (w_5 \times HS) \qquad (1)$$

where:

P	relative value of the sum of one-day precipitation with a period of repeating of 100 years in mm with regard to maximum sum in the given area (–);
A_{CS}	relative value of the size of the contributing surface with regard to maximum considered size of the surface 40 km^2 (km^2);
S	average slope of the contributing surface (%);
AL	share of arable land on the contributing surface (%);
HS	share of heavy soils on the contributing surface (%);
w_1, w_2, w_3, w_4, w_5	weights (0.39; 0.20; 0.27; 0.09; 0.06)—importance of the causal factors was calculated by analytical hierarchy process (AHP) in Microsoft Excel [16]

Defining of contributing surfaces was done in Arc GIS 9—version ArcView 9.3. For all identified critical points are generated the relevant contributing surfaces and the parameters calculated which enter into the analysis of the resulting assessment:

- Size of the contributing surface—calculation using ArcGIS—*Calculate Geometry*.
- Average slope of the contributing surface—calculation of the slope using arctoolbox: Spatial Analyst Tools—Surface—Slope and subsequently calculation of the average slope—*Spatial Analyst Tools—Zonal—Zonal statistic*.
- Share of arable land on the contributing surface—calculate using Field Calculator.
- Share of the area of heavy soils on the contributing surface—calculate using *Field Calculator*.
- Relative value of the sum of 1-day precipitation with a period of repeating of 100 years in mm with respect to the maximum sum in the given area (max. 95 mm)—using *Field Calculator*.

A component of Spatial Analyst is a set of Hydrology instruments which contain a function enabling hydrological analyses of a basin to be carried out.

2.2 Risk Analysis

The riskiness of a locality is determined on the basis of a combination of the hazard of the contributing surfaces and the vulnerability of the territory beneath the critical point according to the rules of a 3 × 3 matrix (Table 1).

The primary aim of risk assessment of contributing surfaces is determining the riskiness of the locality using an ordinal scale (i.e. low, moderate, and high) and providing a foundation for the next stage, called risk management.

2.3 Results

For a practical illustration of the assessment of flood risk from flash floods the entire basin of the Bodva River, situated in the south of Slovakia, is selected. For practical application of the methodological process of selecting measures for flood protection with a focus on lowering the potential for adverse consequences of floods on human health, property and the environment the small town of Medzev in a partial basin of the Bodva was selected. The town of Medzev was in the scope of preliminary assessment of flood risk in Slovakia assessed as an area with an existing potential for significant flood risk.

Table 1 Matrix for calculation of the resulting flood risk from flash floods

V/H	A (≥17.7)	B (8.1–17.6)	C (≤8)
A (>2.3)	AA (high risk)	AB (high risk)	AC (moderate risk)
B (1.5–2.3)	BA (high risk)	AC (moderate risk)	BC (low risk)
C (<1.5)	CA (moderate risk)	CB (low risk)	CC (low risk)

2.4 Study Area

The river basin of the Bodva is demarcated by a contour dividing line which is to the north and east a dividing line toward the component Hornád basin. This dividing line leads along the ridge of the Volovský Mountains and from there turns to the southeast to Košice basin, where it runs along the highest places of the hilly parts of the Košice basin on the border with the Hungarian Republic. From the south the Slovak part of the Bodva basin is bordered by the state border with the Republic of Hungary. The western border of the basin is formed by a dividing line with the Slaná basin. This dividing line leads along the peaks of the Slovak karst region again in the direction of the Republic of Hungary [17].

The Bodva River arises in the Volovský Mountains at an elevation of 900 m a.s.l. The length of its flow along the state border is 48.4 km [18]. Within the territory of Hungary it flows into the Slaná River. A larger right-hand side inflowing tributary of the Bodva is the Turňa and on the left-hand side the Ida. The land use is as follows: artificial surfaces—4.7 %; agricultural areas—48.1 %; forested and semi-natural areas—46.6 %; wetland areas—0.1 %; waters—0.5 %.

2.5 Description of Floods in the Basin

The year 2010 was from the viewpoint of rainfall exceptionally above average in the Bodva basin. The first flood of 2010 in the component basin of the Bodva occurred in the first half of January. In April a second flood occurred which resulted from long-lasting and substantial rainfall which fell over the course of 3 days. These rains caused a rapid and significant rise in water levels. The floods in May and June 2010 were exceptional from the viewpoint of time and spatial distribution in the Bodva basin. In nearly all water measuring stations water stages designated as third degree flood activities were surpassed. Further above-average rainfall occurred in November. Basins reacted to the rainfall with increased flows and in the Bodva the flow of water was at a level which can be achieved or surpassed only one time every 1 to 2 years. Precipitation activities continued even into the early part of December and the larger part of the precipitation was repeatedly in the form of rainfall [19].

In the scope of a flood risk assessment in the Bodva river basin 5 areas (17.2 km) were identified with the occurrence of significant flood risk [19], from this:

- 2 areas with the existing potential for significant flood risk (6.7 km),
- 3 areas with probable potential for significant flood risk (10.5 km).

The list of the areas in which the occurrence of significant flood risk was determined is graphically presented in Fig. 1.

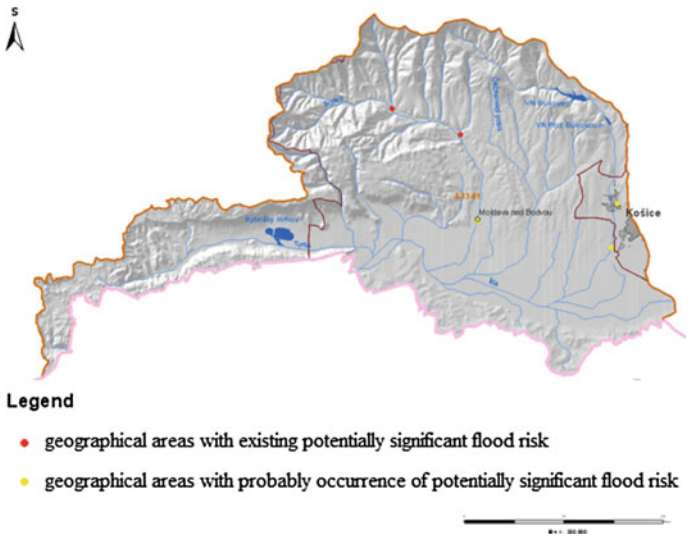


Fig. 1 Geographical area with potentially significant flood risk in the component basin of the Bodva [19]

2.6 Data

With the assessment of flood risk from flash floods the following data foundations are utilized:

- an analysis of flash floods on the territory of the Slovak Republic,
- a digital model of terrain of the assessed territory,
- the edges of built-up municipalities,
- CORINE Land Cover 2006,
- a map of values of the sum of one-day precipitation with a period repeating of 100 years (Slovak Hydrometrological Institute—SHMÚ),
- a map of soil types (Research Institute of Soil Sciences and Soil Protection—VÚPOP).

The documents, analysis and conversion of data are prepared in the GIS environment, specifically ArcGIS 9—version ArcView 9.3 with Spatial Analyst and ArcHydro.

2.7 Results

The aim of the preliminary assessment of flood risk from flash floods is determining the critical points in the basin and their contributing surfaces on the basis of the

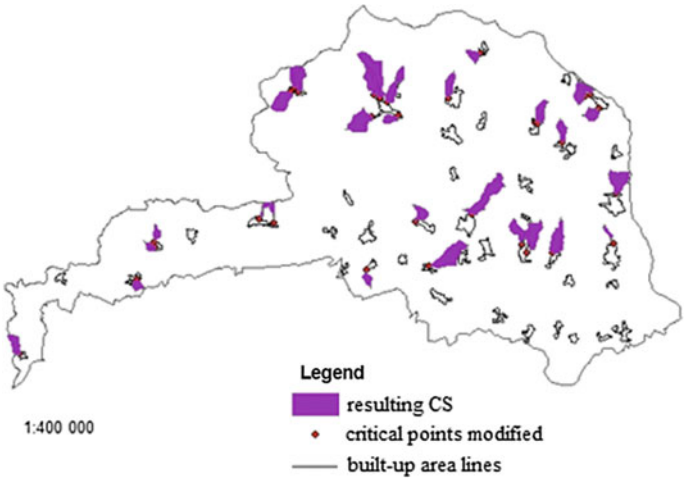


Fig. 2 Resultant contributing surfaces (*CS*) of the identified critical points in the component of the Bodva basin

physiogeographic and geometric characteristics of the contributing surfaces. Individual characteristics enable the determining of the flood hazard; therefore, before the flood risk assessment itself it is important to know the natural relationships of the resolved territory.

The sequence of individual steps of the preliminary assessment of flood risks from flash floods in the partial basin of the Bodva emerges from the proposed methodological procedure described in previous chapter.

Overall, 32 critical points are identified in the basin. For all critical points identified the relevant contributing surfaces are generated (Fig. 2) and the parameters calculated which enter into the analysis of the resulting assessment.

The preliminary parameters are used for calculation of the H criteria, i.e. indicator of critical conditions, according to the relationship (1). All critical points whose relevant contributing surfaces satisfy the given criteria are assigned to another assessment—calculation of risk. From the total number of 32 critical points, 8 critical points, or 8 contributing surfaces, satisfy the entered criteria ($H \geq 5.3$).

For the needs of calculation of flood risk from flash floods of the assessed contributing surfaces, it is necessary to determine the hazard of the selected contributing surfaces and the vulnerability of the territory below the critical points. The class of hazard (A, B and C) depends on the value according to Table 1.

A graphic presentation of the determined hazard classes of the selected contributing surfaces (8 in total) is shown in Fig. 3.

Overall, 5 contributing surfaces in the component basin of the Bodva were determined to be class B hazard (moderate hazard) and 3 contributing surfaces were assigned to class A (high hazard) (Fig. 3).

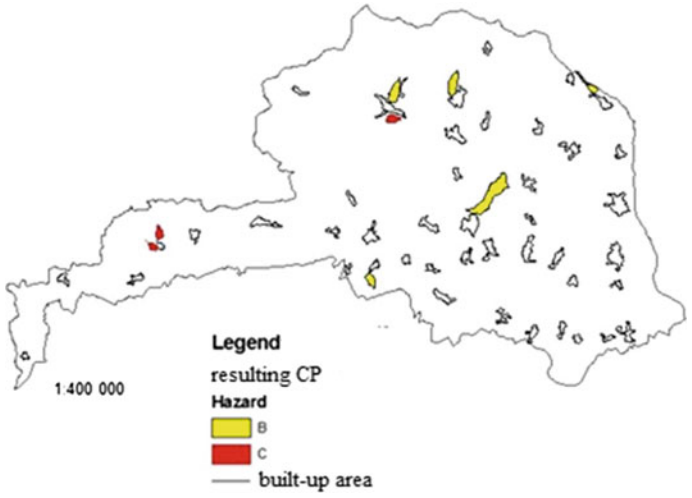


Fig. 3 Resulting hazard of selected contributing surfaces and critical points (*CP*) in the Bodva basin

Vulnerability below the critical point is determined according to the relation (2). The determined values of the individual criteria and resulting vulnerability require reconnaissance of the terrain. Given this fact, vulnerability is determined illustratively for only one critical point, which is located in the northeast part of the town of Medzev. The resolved segment for assessing vulnerability is determined only by a professional estimate, and territories adjacent to the flow, i.e. a built-up area directly adjacent to the bank of the flow, are taken into consideration.

In the resolved segment built-up areas make up only 16.6 % of the total endangered territory, i.e. less than 30 % of the territory is built-up. On the basis of this fact, it is assigned a value of 1 of criteria D (density of built-up areas). Greater emphasis is placed on the type of built-up area below the critical point—criterion T. In the case of this critical point, almost only residential areas are involved, where damage in the case of torrential floods could possibly also lead to loss of human lives. One building is non-residential (for industry). The value of this criteria is calculated as the weighted average of the values of the class of vulnerability, on the basis of the share of built-up areas of individual types of buildings in the endangered (adjacent) territories (a building with vulnerability 1 represents 2.7 % share of built-up areas and the buildings with a vulnerability of class 3 represent a 13.9 % share of the built-up area). Criterion T (type of built-up area) acquires a value of 2.7.

The resultant vulnerability is calculated according to relation 2 and in the case of this solution of the critical point is numbered by the value 2.01. This involves moderate vulnerability, which is in the range from 1.5 to 2.3.

The risk according to the proposed method and the determined matrix (Table 1) is calculated as a combination of hazard and vulnerability. In the case of this critical point the contributing surface is assigned to the risk category BB, which means that the locality is at moderate risk in terms of flash flooding of the basin.

3 Conclusion

The contribution, dealing with solving the problem of assessment and management of flood risk with the goal of effective control aiming at reducing flood risk and thus increasing the measure of flood protection, is developed in the sense of currently valid legislation in the area of flood protection, primarily in the sense of the already mentioned directive 2007/60/EC on the assessment and management of flood risk. The aim of the work was expanding the set of scientific knowledge in the field of assessment and management of flood risk in Slovakia and in the world, and a proposal for directing the management of flood risks with the goal of reducing the adverse effects on human health, the environment and economic activities connected with floods. A goal so conceived had a primary task, namely "Proposal of a methodology for the process of preliminary assessment of flood risk—a methodological procedure for preliminary flood risk assessment from flash floods with respect to the need for its updating following from directive 2007/60/EC and its application in the conditions of a modelled territory".

Acknowledgments This paper was written thanks to support from project VEGA 1/0609/14.

References

1. Hanák, T., Korytárová, J.: Risk zoning in the context of insurance: comparison of flood, snow load, windstorm and hailstorm. J. Appl. Eng. Sci. **12**(2), 137–144 (2014). doi:10.5937/jaes12-6098
2. Pietrucha-Urbanik, K.: Assessment model application of water supply system management in crisis situations. Global NEST J. **16**(5), 893–900 (2014)
3. Petrow, T., Thieken, A.H., Kreibich, H., Bahlburg, C.H., Mercz, B.: Improvements on flood alleviaton in Germany: lessons learned from the Elbe flood in August 2002. Environ. Manag. **38**(5), 717–732 (2006)
4. Korytárová, J., Šlezingr, M., Uhmannová, H.: Determination of potential damage to representatives of real estate property in areas afflicted by flooding. J. Hydrol. Hydromech. **55**(4), 282–228 (2007)
5. Hall, J.: Journal of Flood Risk Management. J. Flood Risk Manag. **3**, 1–2 (2010)
6. Pietrucha-Urbanik, K.: Failure analysis and assessment on the exemplary water supply network. Eng. Fail. Anal. **57**, 137–142 (2015)
7. Chandran, R., Joisy, M.B.: Flood hazard mapping of Vamanapuram river basin—a case study. In: Proceedings of the 10th Conference on technological trend [online]. http://117.211.100.42:8180/jspui/bitstream/123456789/572/1/CE_HE_05.pdf. Accessed February 2014
8. Kandilioti, G., Makropoulos, C.: Preliminary flood risk assessment. The case of Athens. Nat. Hazards **61**(2), 441–468 (2012)
9. Tanavud, C.H., Yongchalermchai, C.H., Bennui, A., Densreeserekul, O.: Assessment of flood risk in Hat Yai Municipality, Southern Thailand, using GIS. J. Nat. Disaster Sci. **26**(1), 1–14 (2004)
10. Scheuer, S., Haase, D., Meyer, V.: Exploring multicriteria flood vulnerability by integrating economic, social and ecological dimensions of flood risk and coping capacity: from a starting point view towards an end point view of vulnerability. Nat. Hazard **58**(2), 731–751 (2011)

11. Yahaya, S., Ahmad, N., Abdalla, F.R.: Multicriteria Analysis for flood vulnerable areas in Hadejia-Jama'are river basin, Nigeria. Eur. J. Sci. Res. **42**(1), 71–83 (2010)
12. Yalcin, G., Akyurek, Z.: Analysing flood vulnerable areas with multicriteria evaluation. In: Proceedings from XXth ISPRS Congress of Geo-Imagery Bridging Continents, pp. 359–364 (2004)
13. Grešková, A.: Identification of risky areas and flood causal risky factors in small basins. Geogr. J. **53**(3), 247–268 (2001)
14. Acreman, M.C., Sinclair, C.D.: Classification of drainage basins according to their physical characteristics: an application for flood frequency analysis in Scotland. J. Hydrol. **84**, 365–380 (1986)
15. TGM WRI: Methodology for critical points identification (2009). http://www.povis.cz/mzp/KB_metodicky_navod_identifikace.pdf. Accessed July 2012
16. Gaňová, L.: Assessment and management of flood risk in the selected river basins in respect of Directive 2007/60/EC. Thesis, Faculty of Civil Engineering, Košice (2014)
17. WRI: General description of Bodva river basin (2009). http://www.vuvh.sk/rsv/docs/PMP/prilohy/priloha_2/priloha_2_1/Bodva.pdf. Accessed March 2014
18. ME SR: Management plan of Bodva river basin (2009). http://www.vuvh.sk/download/RSV/07_PMP_Bodva/01_Plan%20manazmentu%20ciastkoveho%20povodia%20Bodva/PMCP_Bodva.pdf. Accessed March 2014
19. ME SR: Preliminary flood risk assessment in Bodva river basin (2011). http://www.minzp.sk/files/sekcia-vod/bodva-suhrnny.pdf. Accessed March 2014

Invariant-Based Performance Analysis of Timed Petri Net Models

W.M. Zuberek

Abstract In timed Petri nets, temporal properties are associated with transitions as transition firing times (or occurrence times). For net models which can be decomposed into a family of place invariants, performance analysis can be conveniently performed on the basis of its components. The paper presents an approach to finding place invariants of net models and proposes an incremental method which, for large models, can significantly reduce the required amount of computations.

Keywords Timed Petri nets · Place invariants · Performance analysis · Incremental model analysis

1 Introduction

Petri nets [6, 7, 10] have been proposed as a formalism for modeling and analysis of discrete-event systems with asynchronous, interacting components. Computer and communication networks, manufacturing systems and transportation networks are just a few examples of such systems. Popularity of net models is due to a simple and 'natural' representation of concurrent and asynchronous activities, typical for many discrete-event dynamical systems, that, however, cannot be modeled easily using queueing theory or other traditional modeling and evaluation techniques. Moreover, a well-developed mathematical foundations exists for the description and analysis of net models.

In order to study the performance aspects of Petri net models, the duration of activities must also be taken into account. Several types of Petri nets 'with time' have been proposed by assigning 'firing times' to transitions or 'enabling times' to places [1, 8, 11]. In timed Petri nets [2, 9, 12, 13], the events occur in 'real time',

W.M. Zuberek (✉)
Department of Computer Science, Memorial University,
St.John's, NL A1B 3X5, Canada
e-mail: wlodek@mun.ca

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2_52

i.e., there is a (deterministic or stochastic) duration associated with each transition's firing, and different (concurrent) firings of transitions correspond to (concurrent) activities in the modeled systems. For timed Petri nets, the concept of 'state' and state transitions can be formally defined, and used to derive different performance characteristics of the model [13].

For a class of Petri net models, structural analysis, based on place invariants [10], is an attractive approach because it provides an analytical characterization of the model's performance, and also it eliminates the exhaustive analysis of the state space (which can be huge for large models). Structural analysis is based on the set of place invariants, which—for complex models—can be difficult to find. Incremental approach reduces the process of finding basic place invariants by first finding the invariants for very simple submodels of the original models, and then combining the submodels into more complex ones with invariants determined in a way that eliminates many steps of the direct approach to finding the invariants.

Section 2 recalls basic concepts of Petri nets and timed nets, their place invariants and (structural) performance analysis. Finding place invariants is discussed in Sect. 3 while Sect. 4 introduces the incremental approach and compares it with the direct approach of Sect. 3. Section 5 provides some concluding remarks.

2 Nets, Net Invariants, Timed Nets and Performance Analysis

A place/transition (ordinary, i.e., with no arc weights) net $\mathbf{N}$ is a triple $\mathbf{N} = (P, T, A)$ where P is a finite, nonempty set of places, T is a finite, nonempty set of transitions, and A is a set of directed arcs, $A \subseteq P \times T \cup T \times P$, such that for each transition there exists at least one place connected with it. For each place p (and each transition t) the input set, $Inp(p)$ (or $Inp(t)$), is the set of transitions (or places) connected by directed arcs to p (or t). The output sets, $Out(p)$ and $Out(t)$, are defined similarly.

A marked Petri net $\mathbf{M}$ is a pair $\mathbf{M} = (N, m_0)$ where $\mathbf{N}$ is a Petri net, $\mathbf{N} = (P, T, A)$, and m_0 is an initial marking function, $m_0 : P \to \{0, 1, \ldots\}$ which assigns a (nonnegative) integer number of tokens to each place of the net.

Let any function $m : P \to \{0, 1, \ldots\}$ be called a marking in a net $\mathbf{N} = (P, T, A)$.

A transition t is enabled by a marking m iff m assigns at least one token to every input place of this transition. Every transition enabled by a marking m can fire (or occur). When a transition fires, a token is removed from each of its input places and a token is added to each of its output places. This determines a new marking in a net, new set of enabled transitions, and so on. The set of all markings that can be derived from the initial marking is called the set of reachable markings. If this set if finite, the net is bounded, otherwise it is unbounded.

A place p is shared iff it is an input place for more than one transition. A net is (structurally or statically) conflict-free if it does not contain shared places.

A marked net is (dynamically) conflict-free if for any marking in the set of reachable markings, and for any shared place, at most one of transitions sharing the place is enabled. Only bounded conflict-free nets are considered in this paper.

Each place/transition net $\mathbf{N} = (P, T, A)$ can be represented by a connectivity (or incidence) matrix $\mathbf{C} : P \times T \to \{-1, 0, +1\}$ in which places correspond to rows, transitions to columns, and the entries are defined as:

$$\forall p \in P \forall t \in T : \mathbf{C}[p,t] = \begin{cases} -1, & \text{if } t \in Out(p) - Inp(t), \\ +1, & \text{if } t \in Inp(p) - Out(p), \\ 0, & \text{otherwise.} \end{cases}$$

If a marking m_j is obtained from another marking m_i by firing a transition t_k then (in vector notation) $m_j = m_i + \mathbf{C}[k]$, where $\mathbf{C}[k]$ denotes the k-th column of $\mathbf{C}$, i.e., the column representing t_k.

Connectivity matrices disregard 'selfloops', that is, pairs of arcs (p, t) and (t, p); any firing of a transition t cannot change the marking of p in such a selfloop, so selfloops are neutral with respect to token count of a net. A pure net is defined as a net without selfloops [10].

A P-invariant (place invariant) of a net $\mathbf{N}$ is any nonnegative, nonzero integer (column) vector I which is a solution of the matrix equation

$$\mathbf{C}^T \times I = 0,$$

where $\mathbf{C}^T$ denotes the transpose of matrix $\mathbf{C}$. It follows immediately from this definition that if I_1 and I_2 are P-invariants of $\mathbf{N}$, then also any linear (positive) combination of I_1 and I_2 is a P-invariant of $\mathbf{N}$.

A basic P-invariant of a net is defined as a P-invariant which does not contain simpler invariants. All basic P-invariants I of ordinary nets are binary vectors [10], $I : P \to \{0, 1\}$.

A net $\mathbf{N}_i = (P_i, T_i, A_i)$ is a P_i-implied subnet of a net $\mathbf{N} = (P, T, A)$, $P_i \subset P$, iff:

(1) $A_i = A \cap (P_i \times T \cap T \times P_i)$,
(2) $T_i = \{t \in T | \exists p \in P_i : (p,t) \in A \vee (t,p) \in A\}$.

It should be observed that in a (pure) net $\mathbf{N}$, each P-invariant I of $\mathbf{N}$ determines a P_I-implied (invariant) subnet of $\mathbf{N}$, where $P_I = \{p \in P | I(p) > 0\}$; P_I is sometimes called the support of the invariant I; all nonzero elements of I select rows of $\mathbf{C}$, and each selected row i corresponds to a place p_i with all its input (+1) and all output (–1) arcs associated with it.

For the Petri net shown in Fig. 1, the connectivity matrix is:

C	t_1	t_2	t_3	t_4	t_5	t_6
p_1	-1	$+1$	0	0	0	0
p_2	$+1$	-1	0	0	0	0
p_3	0	-1	$+1$	0	0	0
p_4	0	$+1$	-1	0	0	0
p_5	$+1$	0	0	-1	0	0
p_6	0	0	-1	$+1$	0	0
p_7	0	0	$+1$	0	0	-1
p_8	0	0	0	0	-1	$+1$
p_9	-1	0	0	0	$+1$	0

It can be easily observed that the sum of rows 1 and 2, as well as 3 and 4 are equal to zero, so $\{p_1, p_2\}$ and $\{p_3, p_4\}$ are basic place invariants. Similarly, $\{p_5, p_6, p_7, p_8, p_9\}$ is also a basic place invariant. Subnets implied by these invariants are shown in Fig. 2.

The net shown in Fig. 1 has two more place invariants: $\{p_1, p_3, p_5, p_6\}$ and $\{p_2, p_4, p_7, p_8, p_9\}$.

In timed Petri nets each transition takes a 'real time' to fire, i.e., there is a 'firing time' associated with each transition of a net which determines the duration of transition's firings.

A conflict-free timed Petri net $\mathbf{T}$ is a pair $\mathbf{T} = (\mathbf{M}, f)$ where:

M is a conflict-free marked Petri net, $\mathbf{M} = (\mathbf{N}, m_0)$, $\mathbf{N} = (P, T, A)$,

f is a firing time function which assigns the nonnegative (average) firing time $f(t)$ to each transition t of the net, $f : T \to \mathbf{R}^{\oplus}$, and $\mathbf{R}^{\oplus}$ denotes the set of non-negative real numbers.

The behavior of a timed Petri net can be represented by a sequence of 'states' and state transitions where each 'state' describes the distribution of tokens in places as well as firing transitions of the net; detailed definitions of states and state transitions are given in [13]. The states and state transitions can be combined into a graph of reachable states; this graph is a semi-Markov process defined by the timed net $\mathbf{T}$. For cyclic conflict-free timed nets, such state graphs are simple cycles which

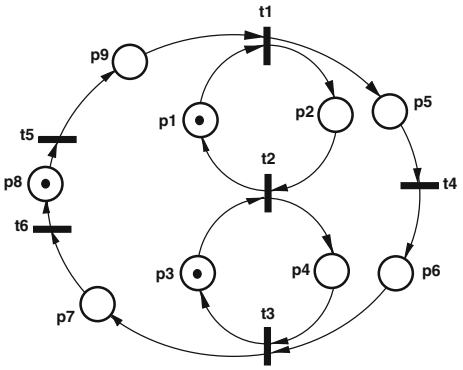


Fig. 1 Petri net

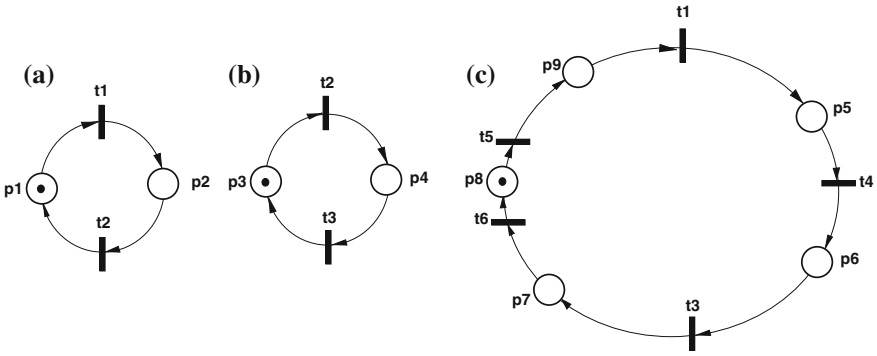


Fig. 2 Invariant implied subnets

represent the cyclic behavior of such nets. Each such timed Petri net contains a basic invariant subnet with the cycle time equal to the cycle time of the whole net. All other subnets, with smaller cycle times, will be subjected to some synchronization delays, imposed by the 'slowest' subnet that determines the cycle time of the whole net.

The cycle time of the net, τ_0, is thus equal to the maximum cycle time if its invariant subnets [11]:

$$\tau_0 = \max(\tau_1, \tau_2, \ldots, \tau_k)$$

where k is the number of subnets covering the original net, and each τ_i, $i = 1, \ldots, k$, is the cycle time of the subnet i, equal to the sum of occurrence times associated with the transitions divided by the total number of tokens assigned to the subnet:

$$\tau_i = \frac{\sum_{t \in T_i} f(t)}{\sum_{p \in P_i} m(p)}.$$

In many cases, the number of basic P-invariants can be reduced by removing from the analyzed net all these elements which do not affect the performance of models. Some of such reductions are discussed in [14].

3 Finding Place Invariants

Finding place invariants can be done in several ways [4, 5]. A polynomial algorithm for finding all place invariants can be derived from the property that the sum of rows of the connectivity matrix corresponding to a place invariant is equal to zero. For each place p_i, the algorithm starts with the i-th row of a connectivity matrix and uses other rows to eliminate nonzero elements in the original row. This is performed

by a recursive procedure "Eliminate" with three parameters, "w" which is the current working vector (initialized to the *i*-th row of the connectivity matrix), "u" which is a set of places constituting the invariant and "r" which ia a set of columns of the connectivity matrix (i.e., transitions) used for reductions of "w":

```
fun Eliminate (w : array [1..n_t] of int, u : set of int, r : set of int);
begin
    if w = 0 then
        add(u,Inv)
    else
        for i := 1 to n_t do
            if w[i] ≠ 0 then
                for j := 1 to n_p do
                    if w[i] + M[j,i] = 0 ∧ r ∩ check(w,j) = {} then
                        Eliminate(w + M[j,*],u ∪ { j },r ∪ { i })
                    fi
                od
            fi
        od
    fi
end
```

"Inv" is the set of invariants and "add(u,Inv)" adds the invariant "u" to the set. "Inv" if it is not there; n_t is the number of transitions and n_p is the number of places. The function "check(w, j)" returns the set of indices of nonzero elements of the sum of "w" and row "j" of "M".

"Eliminate" is invoked for consecutive places of the net model:

```
program Invariants1;
    Inv := {};
    for i := 1 to n_p do
        w := M[i,*];
        u := { i };
        r := {};
        Eliminate(w,u,r)
    od
end
```

The initial steps of finding place invariants for the model shown in Fig. 1 are presented in Table 1; the table shows the vector "w", the set "u" and the set "r" for consecutive invocations of "Eliminate").

Table 1 Finding place invariants for the net shown in Fig. 1

i	w	u	r	
1	−1,+1,0,0,0,0	{1}	{}	
	0.0,0,0,0,0	{1,2}	{1}	invariant {1,2}
	0,+1,0,−1,0,0	{1,5}	{1}	
	0,0,+1,−1.0,0	{1,5,3}	{1,2}	
	0,0,0,0,0,0	{1,5,3,6}	{1,2,3}	invariant {1,3,5,6}
2	+1,−1,0,0,0,0	{2}	{}	
	0,0,0,0,0,0	{2,1}	{1}	invariant {1,2}
	0,−1,0,0,0,+1	{2,9}	{1}	
	0,0,-1,0,0,+1	{2,9,4}	{1,2}	
	0,0,0,0,−1,+1	{2,9,4,7}	{1,2,3}	
	0,0,0,0,0,0	{2,9,4,7,8}	{1,2,3,5}	invariant {2,4,7,8,9}
3	0,−1,+,0,0,0	{3}	{}	
	−1,0,+1,0,0,0	{1,3}	{1}	
	0,0, +1,−1,0,0	{3,1,5}	{1,3}	
	…	…	…	…
9	−1,0,0,0,0,+1	{9}	{}	
	0,−1,0,0,0,+1	{9,2}	{1}	
	0,0,−1,0.0,+1	{9,2,4}	{1,2}	
	0,0,0,0,−1,+1	{9,2,4,7}	{1,2,3}	
	0,0,0,0,0,0	{9,2,4,7,8}	{1,2,3,5}	invariant {2,4,7,8,9}

The complete set of basic place invariants for the net shown in Fig. 1 is:

i	*place invariant*	*implied transitions*
1	$\{p_1, p_2\}$	t_1, t_2
2	$\{p_3, p_4\}$	t_2, t_3
3	$\{p_5, p_6, p_7, p_8, p_9\}$	t_1, t_3, t_4, t_5, t_6
4	$\{p_1, p_3, p_5, p_6\}$	t_1, t_2, t_3, t_4
5	$\{p_2, p_4, p_7, p_8, p_9\}$	t_1, t_2, t_3, t_5, t_6

so the cycle time is:

$$\tau_0 = \max(\tau_1, \tau_2, \tau_3, \tau_4, \tau_5)$$

where:

$$\begin{aligned}
\tau_1 &= f(t_1) + f(t_2),\\
\tau_2 &= f(t_2) + f(t_3),\\
\tau_3 &= f(t_1) + f(t_3) + f(t_4) + f(t_5) + f(t_6),\\
\tau_4 &= (f(t_1) + f(t_2) + f(t_3) + f(t_4))/2,\\
\tau_5 &= f(t_1) + f(t_2) + f(t_3) + f(t_5) + f(t_6).
\end{aligned}$$

Other performance characteristics can be derived in a similar way [3].

4 Incremental Approach

In many cases, the components of the model are known in advance and can be used for incremental approach to finding place invariants of a net model.

For the example shown in Fig. 2, place invariants of simple subnets are obvious (and even formally can be determined in a single step). Submodels (a) and (c) (i.e., subnets implied by place invariants $\{p_1, p_2\}$ and $\{p_5, p_6, p_7, p_8, p_9\}$) are combined using a single transition (t_1) which does not change the invariants. The combined subnet implied by $\{p_1, p_2, p_5, p_6, p_7, p_8, p_9\}$ is merged with the remaining subnet by transitions t_2 and t_3, as shown in Fig. 3. This integration can create new invariants, but all such invariants must contain places connected to t_2 and/or t_3, so only these places should be checked for new place invariants.

In general, if two subnets are merged by transitions in a set T_{shared}, new place invariants are found by the same procedure as before but restricted to places in the set $Inp(T_{shared})$ and $Out(T_{shared})$:

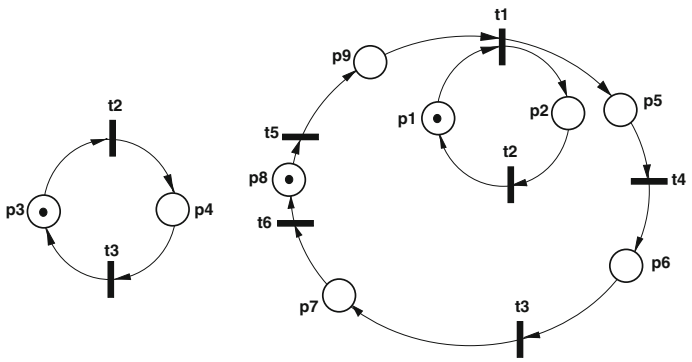


Fig. 3 Merging subnets

```
program Invariants2;
    Inv := {};
    for each p_i in Inp(T_shared) ∪ Out(T_shared) do
        w := M[i,*];
        u := { i };
        r := {};
        Eliminate(w,u,r)
    od
end
```

Analysis of the case shown in Fig. 3 corresponds to a part of Table 1 that includes places p_1 to p_4, p_6 and p_7, which is about one half of the total Table.

It should be observed that the gains of the incremental approach actually increase with the size of the model.

5 Concluding Remarks

Invariant-based performance analysis derives analytical characterization of the model's performance provided the model is covered by a family of conflict-free subnets. If this is the case, the subnets are implied by place invariants of the net model. Efficient method of finding place invariants uses an incremental approach of merging simple submodels into more complex ones.

It should be noted that the efficiency of the incremental approach depends upon ordering the merged models. For instance, if the first step of merging the submodels shown in Fig. 2 combines submodels (a) and (b) rather than (a) and (c), the potential advantages of the approach will be lost.

The process of finding place invariants can be simplified in several ways, for example, by model reductions. Each simple path in the model can be reduced to a single element because any place invariant must either contain the whole path or none of its elements. Similarly, parallel paths (i.e., simple paths originating and terminating in a single transition) can be merged because if a place invariant contains one of these paths then there must exist another invariant containing the other path—there is no need to repeat the steps of finding these two invariants independently.

A similar considerations in the context of deadlock analysis (and finding siphons in net models) showed that simple net reductions can significantly reduce the required computations [14].

Acknowledgments The Natural Sciences and Engineering Research Council of Canada partially supported this research through grant RGPIN-8222.

References

1. Ajmone Marsan, M., Balbo, G., Conte, G., Donatelli, S., Franceschinis, G.: Modeling with Generalized Stochastic Petri Nets. Wiley and Sons (1995)
2. Holliday, M.A., Vernon, M.K.: A generalized timed Petri net model for performance evaluation. In: Proceedings of the International Workshop on Timed Petri Nets, Torino, Italy, pp. 181–190 (1985)
3. Jain, R.: The Art of Computer Systems Performance Analysis. Wiley & Sons (1991)
4. Krueckeberg, F., Jaxy, M.: Mathematical methods for calculating invariants in Petri nets. In: Rozenberg, G. (ed.) Advances in Petri Nets 1987 (Lecture Notes in Computer Science 266), pp. 104–131. Springer (1987)
5. Martinez, J., Silva, M.: Simple and fast algorithm to obtain all invariants of a generalized Petri net. In: Applications and Theory of Petri Nets (Informatik Fachberichte 52), pp. 301–310. Springer (1982)
6. Murata, T.: Petri nets: properties, analysis and applications. Proc. IEEE **77**(4), 541–580 (1989)
7. Peterson, J.L.: Petri Net Theory and the Modeling of Systems. Prentice-Hall (1981)
8. Popova-Zeugmann, L.: Time and Petri Nets. Springer (2013)
9. Ramchandani, C.: Analysis of Asynchronous Concurrent Systems by Timed Petri Nets. Project MAC Technical Report Mac–TR–120. Massachusetts Institute of Technology, Cambridge (1974)
10. Reisig, W.: Petri Nets—An Introduction (EATCS Monographs on Theoretical Computer Science 4). Springer (1985)
11. Sifakis, J.: Use of Petri nets for performance evaluation. In: Measuring, Modelling and Evaluating Computer Systems, pp. 75–93. North-Holland (1977)
12. Wang, J.: Timed Petri Nets. Kluwer Academic Publ. (1998)
13. Zuberek, W.M.: Timed Petri nets—definitions, properties and applications. Microelectron. Reliab. (Special Issue on Petri Nets and Related Graph Models) **31**(4), 627–644 (1991)
14. Zuberek, W.M.: Siphon-based verification of component compatibility. In: Proceedings of 4-th International Conference on Dependability of Computer Systems (DepCoS-09); Brunow Palace, Poland, pp. 123–132 (2009)

Author Index

W. Zamojski et al. (eds.), *Dependability Engineering and Complex Systems*,
Advances in Intelligent Systems and Computing 470,
DOI 10.1007/978-3-319-39639-2