Modern Computational Methods and their Applications in Engineering Science

Monografie – Politechnika Lubelska

Politechnika Lubelska
Wydział Mechaniczny
ul. Nadbystrzycka 36
20-618 LUBLIN

Modern Computational Methods and their Applications in Engineering Science

edited by
Katarzyna Falkowicz
Mirosław Szala

Wydawnictwo Politechniki Lubelskiej
Lublin 2020

Reviewer:
Dr hab. inż. Dariusz Czerwiński, prof. Politechniki Lubelskiej

Publikacja wydana za zgodą Rektora Politechniki Lubelskiej

© Copyright by Politechnika Lubelska 2020

ISBN: 978-83-7947-454-7

Wydawca: Wydawnictwo Politechniki Lubelskiej
www.biblioteka.pollub.pl/wydawnictwa
ul. Nadbystrzycka 36C, 20-618 Lublin
tel. (81) 538-46-59

Druk: DjaF – 30-092 Kraków, ul. Kmietowicza 1/1
www.djaf.pl

Elektroniczna wersja książki dostępna w Bibliotece Cyfrowej PL www.bc.pollub.pl

CONTENTS

Łukasz Gałka, Jakub Machowski, Kamil Żyła

APP INVENTOR PERFORMANCE IN IMAGE CLASSIFICATION 6

Sławomir Przyłucki, Paweł Powroźnik

MANAGEMENT OF FAIR ALLOCATION OF NETWORK RESOURCES IN STREAM SERVICES BASED ON SDN PRINCIPLES 21

Stanisław Skulimowski, Jerzy Montusiewicz

OPTIMIZATION METHODS OF SEARCHING ALGORITHMS FOR 2D ELEMENTS MATCHING 35

Jolanta Słoniec, Anna Kaczorowska

ON ETHICS IN ARTIFICIAL INTELLIGENCE 48

Monika Kaczorowska, Mikhail Tokovarov

USER EXPERIENCE DRIVEN ASSESSMENT OF SELECTED FEATURES AND PERSPECTIVES OF INSTANT MESSAGING APPLICATIONS 62

Agnieszka Żarko, Paweł Cielma, Małgorzata Plechawska-Wójcik

A TOOL SUPPORTING PRIMER GENERATION IN THE STUDY OF DNA METHYLATION 71

Mateusz Mazurek, Jakub Mielniczuk, Marcin Badurowicz

COMPARISON OF POTHOLE DETECTION ALGORITHMS BASED ON ACCELEROMETER DATA 84

Bernadetta Stachura-Terlecka, Antoni Ligeza

EPIDEMIC MODELLING WITH CELLULAR AUTOMATA 94

Łukasz Gałka, Jakub Machowski, Kamil Żyła[1]

APP INVENTOR PERFORMANCE IN IMAGE CLASSIFICATION

Keywords: App Inventor, Java for Android, image classification, neural networks.

Abstract

Image classification technologies are constantly gaining importance in industrial and personal use, similar to mobile devices and applications. In parallel, the authors observe the popularity of developer tools with lower entry levels than classical programming languages, like Java, the tools being a part of the so-called "Do It Yourself" approach. One such tool is App Inventor with recently added support for image classification. Due to the abovementioned factors, the authors have decided to check App Inventor performance against Java for Android in image classification based on the TensorFlow library. The purpose of such a test was to determine whether the App Inventor is able to classify images in the mobile environment of limited resources with reasonable outcomes. Analogous applications were built in each technology, then the applications were run on the same set of images depicting a few different kinds of objects. The effectiveness and time of image classification were measured. It was revealed that both applications had comparable effectiveness of image classification and most of the objects were correctly recognised. Unfortunately, the application in App Inventor was on average two times slower than the other application, although the time of a single classification did not exceed 0.3s. It could be concluded that when deciding about App Inventor usage, it has to be considered what is more important – the performance whether the simplicity of coding applications.

1. Introduction

The image classification technologies are constantly gaining importance in industrial and personal use [1, 2]. They are applied in a variety of domains – from fruit classification [3] to the detection of defects in materials [4]. Similar trends could be observed in the case of mobile devices and applications, just to mention their wide range of applications and the growth of their market [5].

In parallel, the authors observe the popularity of developer tools with lower entry levels than classical programming languages, like Java [6], the tools being a part of a so-called "Do It Yourself" (DIY) approach [7]. The approach could be simply explained as providing the ability to develop everyday use applications, either simple business applications, for people who are not necessarily IT experts [7-9]. One of the extremely popular and successful tools within the DIY approach is App Inventor [10], which has recently received support for image classification [11].

[1] Lublin University of Technology, Department of Computer Science, Nadbystrzycka 36B, Lublin, Poland, k.zyla@pollub.pl

Machine learning, and in particular image classification, on the Android platform is under dynamic evolution that involves software as well as hardware [12]. After initial performance problems a lot of improved making application possibilities, available to the users, similar to non-mobile platforms. Java as the native language for the Android platform complies with that trend. These also imply that the research on image classification in the case of Java for Android is broad and is going well. [13] Much less evidence concerns App Inventor, thus it needs further research. During literature analysis it was revealed that the main topics in the context of App Inventor and machine learning concern the following: the teaching of machine learning (e.g. [14–16]), creating apps associated with external solutions that collect and process data (e.g. [17–19]), creating apps acquiring data to be processed elsewhere (e.g. [20, 21]).

Thus, the goal of this work was stated as comparing the performance of apps for image classification made in Java for Android and App Inventor environment. Due to the unsatisfactory state of research concerning machine learning and App Inventor, the authors think that this work is valuable and needed, what is more, it can be seen as supplemental to other works critically evaluating the quality and performance of the App Inventor platform, to mention [8, 9, 22–25].

2. Overview of development tools and environment

2.1. Android platform

The goal of the Android operating system is to provide the full functionality of advanced mobile devices to their users. Android is an open-source solution developed by Google, which is based on the Linux kernel. It was designed to manage devices that are usually equipped with additional modules like touchscreen, cameras, gyroscope, GPS, Wi-Fi, Bluetooth, and many more [26].

Android architecture could be divided into 4 main layers (Fig. 1). The lowest layer is the Linux kernel, which manages all necessary device drivers, and provides abstraction mechanisms for the higher layers. The next one is the Libraries layer incorporating SQL database, website browser engine, libraries responsible for the security, among others. Runtime environment (RE) exists at the same level. It contains a Dalvik Virtual Machine (DVM), which is a kind of virtual machine optimized for Android. It uses operating system functionalities like memory management and multithreading. Another important part of RE is the set of Java libraries – thanks to them, Android applications can be written in Java. Next, the Application Framework layer provides a set of high-level services and facilitates its usage for Android applications, that are intended to exist in the highest architecture layer [27].

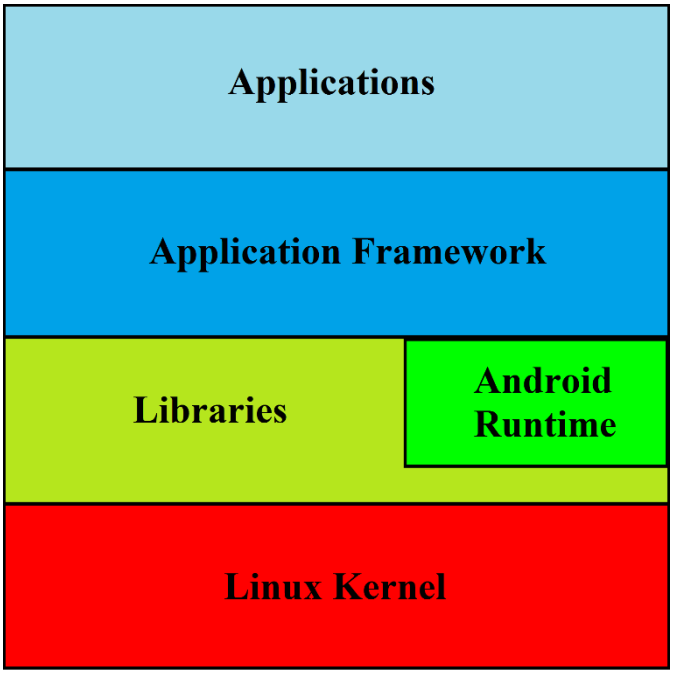


Fig. 1. Layers of the Android operating system architecture (see [27] for details).

Android, as a full-blown platform for mobile devices, provides the full programming stack. Thanks to dedicated software development kits, programmers could write their applications in Java as well as in C/C++ languages or even in assembler. The first possibility is provided by the SDK (Software Development Kit), the later ones (being called together as the native code) are provided by the NDK (Native Development Kit). The native code is usually used in custom performance (speed/memory) optimisations, especially fruitful when utilising hardware-specific instruction sets. Further details can be found e.g. in [27, 28].

Android developers could choose from among a variety of tools. One group of tools is designed specifically for programmers, like Android Studio (supports Java and Kotlin) [29], either Visual Studio (supports Xamarin) [30]. On the opposite are integrated environments, for less advanced developers, supporting a model-driven engineering paradigm, like MIT App Inventor [31].

Android is the most popular mobile operating system. According to the PCWorld.pl [32], its worldwide share in the mobile operating systems market was at the level of 75% in 2019. The next one was iOS-almost 23% of the market share.

2.2. App Inventor

MIT App Inventor (Fig. 2) was developed to simplify the development of everyday use mobile applications by people who are not professional

programmers; as well as to teach the basics of programming and critical thinking [10, 31]. Applications are built in 2 stages. First, the interface is designed by dragging and dropping appropriate components from the palette. Second, the logic of the application is created by mixing methods and attributes of components from the previous stage with other language constructs typical for event-driven and visual programming. All elements creating the application logic are depicted as puzzles.

It would take another monograph to address all advantages and disadvantages of the App Inventor environment, thus the authors would like to mention here only the 3 main aspects. First, App Inventor editor is a web application and developed applications are stored in a cloud – this is a big advantage when organising the workplace. Second, the logic of the application is stripped off of many irrelevant elements in a way that allows developers to fully focus on the functionality of the application instead of implementation details (these are managed by the code generator provided with App Inventor). Third, by design ability to easily extend the puzzles by new concepts is lost, nonstandard tasks require classical programming, and logic made of puzzles occupy a lot of screen space. Although these are not seen as disadvantages in some specific applications, like introductory computer science classes.

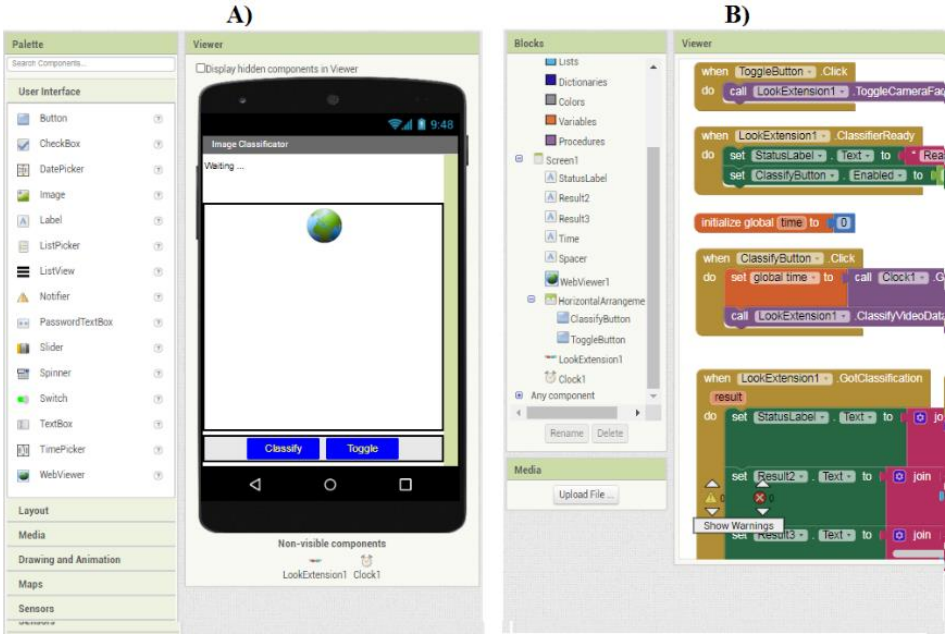


Fig. 2. Interface of App Inventor for designing an application interface (A) and for developing an application logic (B).

2.3. Java and Android Studio

Android Studio is an integrated programming environment [27, 33], which is the native development environment for Android recommended for professional applications (Fig. 3). Android Studio allows to visually design an application interface, which is similar to App Inventor. The application logic could be written using languages like Java, either Kotlin [33]. These are textual object-oriented languages used by professionals – in other words, serious business, no more multicolour puzzles.

As the biggest advantage of Android Studio, the authors could mention complying with industrial needs and standards of software development for the Android platform. As a disadvantage, the authors could mention a higher technological barrier for people than in the case of App Inventor.

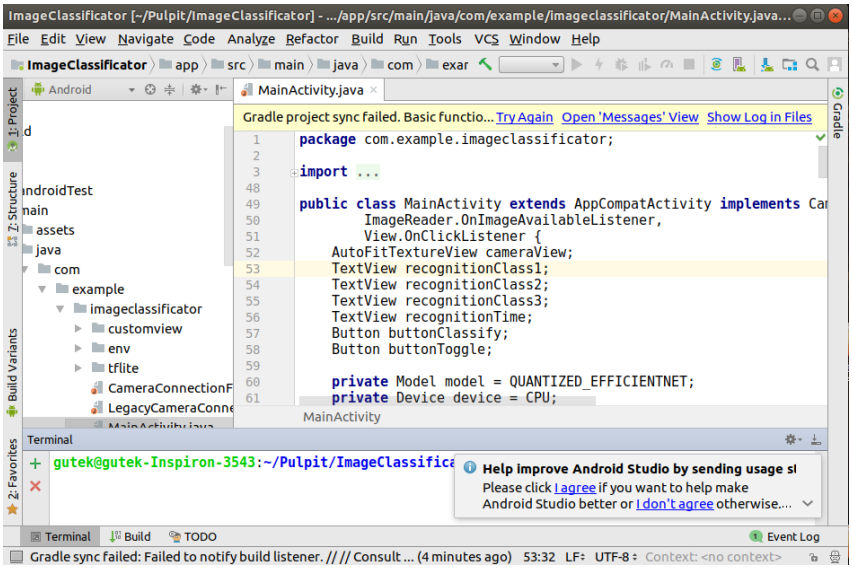


Fig. 3. Android Studio interface.

3. Aim and research procedure

The aim of this work is to compare the performance of applications for image classification made in Java for Android and the App Inventor environment. The following research questions were stated:

1. Is Java for Android significantly faster than App Inventor in image classification?
2. Is the classification quality significantly different between App Inventor and Java for Android?

In order to answer these questions, two applications providing the same functionality were developed in App Inventor and Java for Android (development details are presented in Chapter 4). In both cases, the newest versions of tools were used. Then the applications were installed on the Motorola Moto G7 Power device running Android 9 Pie (API 28).

Classification tests were performed for 9 scenes where the objects to be recognised/classified included: TV receiver, hand watch, keys, pen, mug, laptop screen, a roll of toilet paper, paper notebook, and electrical outlet. For each scene, the Motorola device was put still (immobilised) and aimed at the object. Then the series of 10 classifications (one after another) occurred in each application. Before switching the application, the device was restarted, and the authors waited for the operating system to load completely. The time of each classification as well as its effectiveness (probability associated with the assigned object category) were measured.

The TensorFlow library [34] was used for the purpose of classification – TensorFlow.js edition in the case of App Inventor, and TensorFlow Lite in the case of Java for Android. TensorFlow, specifically TensorFlow.js, was the only option available for App Inventor due to the specificity of implementation of the "LookExtension" [16] responsible for machine learning. The authors acknowledge that they might hurt comparability by confronting TensorFlow.js against Tensor Flow Lite, although the authors would like to make the comparison close to the real practice, and it is very unlikely that Java developers would choose JS over Lite edition.

Network pre-trained on 1,000 labels, described in [16], was used in both cases. According to [16] the training procedure involved the ImageNet dataset [35]. The authors could have trained the network themselves – both tools (App Inventor and Java for Android) provide such a possibility. Nevertheless, due to the high complexity of the methodologically correct procedure of network training, it was assumed that the majority of people would choose an off-the-shelf solution. Thanks to such choice the authors have also avoided at least some potential flaws and made the results closer to the real-life use cases by referring to publicly accessible resources.

4. Application for image classification testing

4.1. Application design

In order to test the performance of image classification, a dedicated application was designed and implemented in App Inventor and Java (the native development tool for Android). It provided the following functionality:

- ability to take a photo using the front either the rear camera,
- ability to classify an object being the main theme of the taken photo;
- ability to measure time and accuracy of the classification.

The application interface consisted of the following elements, which were the same in both implementations (Fig. 4):

- text fields for displaying results of the classification process (3 best guesses), probability of each guess, and a total time of the classification;
- a field for displaying an image from a smartphone camera;
- buttons for starting the classification process and switching between cameras.

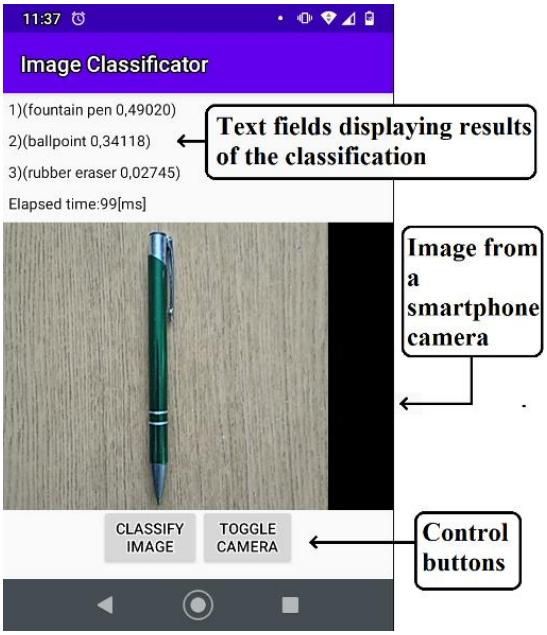


Fig. 4. Interface of the application for image classification testing.

Additional constraints were issued on the application, i.e.: requirement to be compatible with Android Pie (API 28) run on the test device; the time limit for a single classification set to 1s.

4.2. Implementation in App Inventor

This subchapter presents the specificity of App Inventor, as well as some more important parts of the application logic implementation. Due to the space constraints, these parts are related to the image classification.

Image classification was based on the App Inventor extension called "LookExtension", using the TensorFlow.js library, provided by the Massachusetts Institute of Technology. The extension was configured to work with the pre-trained network (for details on the training see chapter 3).

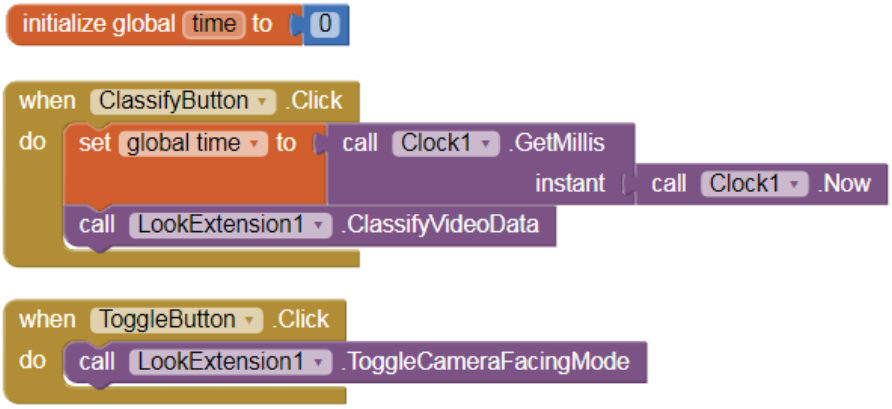


Fig.5. Application logic for initialising the classification process.

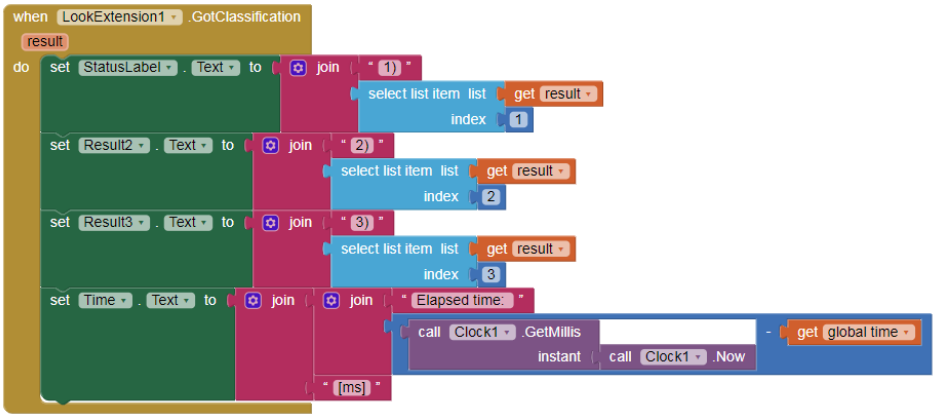


Fig. 6. Application logic for receiving the classification results.

Figure 5 presents the implementation of events occurring after pressing the buttons. “ToggleButton” implementation calls the method of the extension that is responsible for switching cameras. “ClassifyButton” implementation first registers the current time in milliseconds – it will later be used to compute the time of classification. Second, there is a call to the “ClassifyVideoData” method of the extension – this method classifies images captured by the smartphone camera. When the classification is done the “GotClassification” event is triggered – see Figure 6 for its logic implementation. First, the current time is captured, and the time of the process is computed. Then results of the classification are displayed – “result” argument of the event procedure is a list that contains results, i.e. a recognised object class and accuracy (probability to be more precise) of the recognition. The logic shown in Figure 6 displays on the screen the 3 most probable results of each classification.

4.3. Implementation in Java for Android

This subchapter presents the specificity of Java and Android Studio, as well as some more important parts of the application logic implementation. Due to the space constraints, these parts are related to the image classification, which was based on the Tensor Flow Lite library. The pre-trained network was used – for details on training see chapter 3.

Listing 1 shows code serving the event of switching cameras and the event of classifying images captured by a camera. In order to implement the "changeCamera()" method, it took 32 lines of code (without empty lines). Another 19 lines of code (without empty lines) were written to check whether camera usage is configured properly.

Listing 1. Implementation of events of pressing buttons within the application.

```
@Override
public void onClick(View v){
   switch (v.getId()) {
       case R.id.classifyButton: {
                         processImage();
                         break;
                        }
       case R.id.toggleButton: {
                         changeCamera();
                         break;
                        }
   }
}
```

The code of the "processImage()" method is shown in Listing 2. It calls methods processing a snapshot captured by a camera to the form acceptable by the method performing classification. The time in milliseconds before and after the classification is measured as well. Results of the classification, in the form of a list, are passed to the "showResults()" method. For an excerpt of code from this method, see Listing 3.

Listing 2. Implementation of the "processImage()" method.

```
protected void processImage() {
   if (isProcessingFrame) {
       return;
   }
   rgbFrameBitmap.setPixels(getRgbBytes(), 0, previewWidth, 0,
           0, previewWidth, previewHeight);
   final int cropSize = Math.min(previewWidth, previewHeight);

   if (classifier != null) {
       final long startTime = SystemClock.uptimeMillis();
```

```
        final List<Classifier.Recognition> results =
                classifier.recognizeImage(rgbFrameBitmap,
                sensorOrientation);
        lastProcessingTimeMs = SystemClock.uptimeMillis() -
                startTime;
        LOGGER.v("Detect: %s", results);

        showResults(results);
    }
    readyForNextImage();
}
```

Listing 3. Excerpts from the "showResults()" method.

```
@SuppressLint("SetTextI18n")
protected void showResults(List<Recognition> results) {
    if (results != null && results.size() >= 3) {
        Recognition recognition = results.get(0);
        if (recognition != null) {
            if (recognition.getTitle() != null &&
                    recognition.getConfidence() != null)
                recognitionClass1.setText("1)(" +
                        recognition.getTitle() + " " +
                        String.format(Locale.getDefault(), "%.5f",
                        recognition.getConfidence()) + ")");
        }

        //...

        recognitionTime.setText("Elapsed time:" +
                lastProcessingTimeMs + "[ms]");
    }
}
```

5. Research results

In order to obtain comparable outcomes, both applications analysed images of the objects presented in Figure 7. The results are presented in Table 1. Columns titled "avg time" present the average time of all measurements concerning a particular object and particular application. By "avg probability" the authors mean average probability of all measurements concerning a particular object. The column "The best guess" presents the result of an object classification with the highest probability associated. Finally, the column "The correct guess" presents the correct result of an object classification altogether with its probability. "N/A" means that an object was not correctly recognised, in other words despite obtaining the 3 best results of classification, the correct category name was absent.

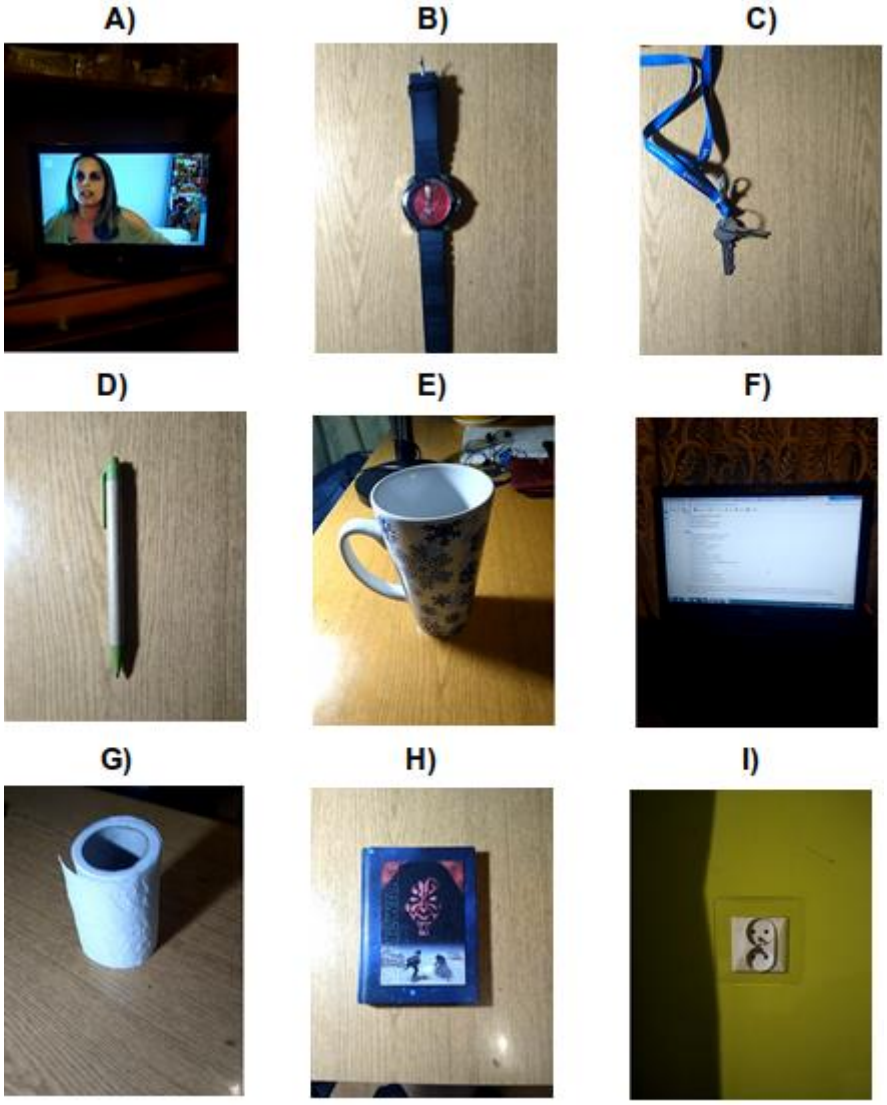


Fig. 7. Objects being the subject of the classification process: (A) TV receiver; (B) hand watch; (C) keys; (D) pen; (E) mug; (F) laptop screen; (G) roll of toilet paper; (H) paper notebook; (I) electrical outlet.

Problems with correct recognition and classification of the objects were observed. 2 of 9 object categories were not returned as the results with the highest probability associated, although they were included in the top 3 results. Keys were recognised as scissors in the first place, most likely due to their layout/arrangement imitating scissors. Although it is hard to explain why the pen was recognised as a hammer in the first place. Further 2 object categories were

not recognised at all. The authors think that the paper notebook was incredibly hard to recognise, although the electrical outlet not recognised at all is surprising. Nevertheless, the main culprit of the classification quality is not App Inventor either Java for Android itself, but rather the quality of training of the network that was used, as well as compromises typical for the mobile environment, like sacrificing the quality of predictions to obtain faster response times.

Table 1. Results of the classification process performed on the images from Figure 7.

Object	Object name	App Inventor	Java	The best guess	The correct guess
		avg time [ms]	*avg time [ms]*	*name/avg probability*	*name/avg probability*
A	TV receiver	204	100	TV/0.46	TV/0.46
B	hand watch	203	100	watch/0.12	watch/0.12
C	keys	213	104	scissors/0.30	key/0.16
D	pen	206	100	hammer/0.26	pen/0.07
E	mug	210	100	cup/0.37	cup/0.37
F	laptop screen	201	102	display/0.56	display/0.56
G	roll of toilet paper	205	102	toilet tissue/0.38	toilet tissue/0.38
H	paper notebook	208	103	handkerchief/ 0.19	N/A
I	electrical outlet	203	102	lighter/0.35	N/A

Wilcoxon signed-rank test revealed that Java for Android is faster than App Inventor in the image classification in the statistically significant degree (p-value<0.01; H1: times obtained by Java for Android are lower than in case of App Inventor). Another test of the same kind revealed that there is no statistically significant difference between Java for Android and App Inventor in case of the probability of the best and the correct guess of the object category (each time – p-value<0.01; H1: scores of Java for Android and App Inventor are equal). The confidence level was set to 0.05.

6. Conclusions

The main goal of this work was to assess the App Inventor performance against Java for Android in image classification based on the TensorFlow library. The purpose of such a test was to determine whether the App Inventor is able to classify images in the mobile environment of limited resources with reasonable outcomes.

In order to fulfil the goal, analogous applications were developed and used in a series of tests involving 9 different scenes exposing diversified objects, from a TV screen, through an electrical outlet, to a pen. After analysis of the results, it was revealed that both applications had comparable effectiveness of image classification. Unfortunately, the application in App Inventor was significantly slower than in the case of Java for Android – on average two times slower. Nevertheless, the time of a single classification has never exceeded 300 ms, thus the difference in speed might not be perceived by many users, especially in the case of simple applications.

Another aspect is the development of the application logic. In the case of App Inventor, it was really fast and simple. The implementation was significantly shorter as well. In the case of Java for Android, its main strength over App inventor is being the environment for professionals, where the performance of code matters, and where the full control over the code is possible. Although a lot more skills are needed during development.

When it comes to future works, it would be interesting to check the performance of App Inventor in solutions for production monitoring, either house surveillance, when a video stream of significant fps rate has to be analysed. Another goal could be improving the quality of classification.

Finally, the authors conclude that when deciding about App Inventor usage, it has to be considered what is more important – the performance whether the simplicity in coding applications.

References

[1] D. Lu, Q. Weng, "A survey of image classification methods and techniques for improving classification performance", *International Journal of Remote Sensing*, vol. 28(5), pp. 823-870, 2007. DOI: 10.1080/01431160600746456

[2] S. M. Jameel, M. A. Hashmani, H. Alhussain, A. Budiman, "A fully adaptive image classification approach for industrial revolution 4.0", *Advances in Intelligent Systems and Computing*, vol. 843, pp. 311-321, 2019. DOI: 10.1007/978-3-319-99007-1_30

[3] M. S. Hossain, M. Al-Hammadi, G. Muhammad, "Automatic fruit classification using deep learning for industrial applications", *IEEE Transactions on Industrial Informatics*, vol. 15 (2), pp. 1027-1034, 2019. DOI: 10.1109/TII.2018.2875149

[4] T. Czimmermann, G. Ciuti, M. Milazzo, M. Chiurazzi, S. Roccella, C. M. Oddo, P. Dario, "Visual-based defect detection and classification approaches for industrial applications - a survey", *Sensors*, vol. 20(5), pp. 1-25, 2020. DOI: 10.3390/s20051459

[5] K. Żyła, "Attention Deficit Hyperactivity Disorder detection – from psychological checklists to mobile solutions", *Studies in Logic, Grammar and Rhetoric*, vol. 60(1), pp. 85-100, 2019. DOI: 10.2478/slgr-2019-0047

[6] Java homepage. Available: https://www.java.com (access date: 20.06.2020)

[7] B. Kawasaki, "App development should be democratized", *DZone / Mobile Zone*, May 20, 2016. Available: https://dzone.com/articles/app-development-should-be-democratized (access date: 20.06.2020)

[8] K. Żyła, “Simplified graphical domain-specific languages for the mobile domain – perspectives of learnability by nontechnical users”, *Applied Computer Science*, vol. 13(3), pp. 32-40, 2017. DOI: 10.23743/acs-2017-19
[9] K. Żyła, “Perspectives of simplified graphical domain-specific languages as communication tools in developing mobile systems for reporting life-threatening situations”, *Studies In Logic, Grammar And Rethoric*, vol. 43(1), pp. 161-175, 2015. DOI: 10.1515/slgr-2015-0048
[10] App Inventor homepage. Available: http://appinventor.mit.edu/ (access date: 20.06.2020)
[11] App Inventor - personal image classifier. Available: https://appinventor.mit.edu/explore/resources/ai/personal-image-classifier (access date: 20.06.2020)
[12] A. Ignatov, R. Timofte, W. Chou, K. Wang, M. Wu, T. Hartley, L. van Gool, “AI benchmark: running deep neural networks on Android smartphones”, *LNCS*, vol. 11133, pp. 288-314, 2019. DOI: 10.1007/978-3-030-11021-5_19
[13] Y. Deng, “Deep learning on mobile devices: a review”, *Proceedings Volume 10993, Mobile Multimedia/Image Processing, Security, and Applications 2019*, 2019. DOI: 10.1117/12.2518469
[14] J. V. Brummelen, “Conversational artificial intelligence development tools for K-12 education”, Available: https://pdfs.semanticscholar.org/c837/3eb721d292f33176c73e813c32eac97ecb5c.pdf (access date: 25.06.2020)
[15] K. Zhu, “An educational approach to machine learning with mobile applications”, M.Sc. Eng. thesis, MIT, 2019.
[16] D. Tang, “Empowering novices to understand and use machine learning with Personalized Image Classification models, Intuitive Analysis Tools, and MIT App Inventor”, M.Sc. Eng. thesis, MIT, 2019.
[17] E. K. Jose, S. Veni, “Vacant parking lot information system using transfer learning and IoT”, *Journal of ICT Research and Applications*, vol. 12(3), pp. 207-218, 2018. DOI: 10.5614/itbj.ict.res.appl.2018.12.3.1
[18] S. Merugula, A. Chakka, S. Muppidi, “Smart health monitoring system using MIT App Inventor 2”, *Advances in Intelligent Systems and Computing*, vol. 1076, pp. 243-253, 2020. DOI: 10.1007/978-981-15-0947-6_24
[19] W. Chuah, R. Tennakoon, R. Hoseinnezhad, A. Bab-Hadiashar, “Machine Vision-Enabled Traffic Controller for Safer and Smoother Traffic Flow Around Construction Sites”, *2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand*, pp. 4255-4260, 2019. DOI: 10.1109/ITSC.2019.8917194
[20] L. Pingali, “Personal oral health advisor using multimodal sensing and machine learning with smartphones and cloud computing”, *2019 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM), Bengaluru, India*, pp. 17-24, 2019. DOI: 10.1109/CCEM48484.2019.000-3
[21] S. Mitini-Nkhoma, “Using mobile phones for computer aided detection of anomalies on chest radiographs: proof of concept”, *medRxiv*, pp. 1-3, 2019. DOI: 10.1101/19012146
[22] J. Smołka, B. Matacz, E. Łukasik, M. Skublewska-Paszkowska, “Performance analysis of mobile applications developed with different programming tools”, *MATEC Web of Conferences*, vol. 252, pp. 1-6, 2019.

[23] R. Kowalczyk, Ł. Turczyński, K. Żyła, “Comparison of APP inventor 2 and Java in creating personal applications for android on example of a notepad”, *Advances in Science and Technology Research Journal*, vol. 10(31), pp. 247-254, 2016. DOI: 10.12913/22998624/64058
[24] K. Żyła, M. Iwachow, “Aplikacja mobilna do raportowania problemów środowiskowych – analiza możliwości App Inventor 2”, Logistyka, no. 4, pp. 7160-7169, 2015.
[25] K. Żyła, „Wydajność implementacji podstawowych metod całkowania w środowisku APP Inventor”, *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*, vol. 5(1), pp. 45-48, 2015. DOI: 10.5604/20830157.1148047
[26] C. Wang, W. Duan, J. Ma, C. Wang, “The research of Android System architecture and application programming”, *Proceedings of 2011 International Conference on Computer Science and Network Technology, Harbin*, pp. 785-790, 2011. DOI: 10.1109/ICCSNT.2011.6182081
[27] Android-developers section. Available: https://developer.android.com/ (access date: 20.06.2020)
[28] H. Guihot, “Pro Android apps performance optimization”, Apress, 2012. ISBN-13: 978-1430239994
[29] J. Annuzzi Jr., L. Darcey, S. Conder, “Introduction to Android application development: Android essentials (5th Ed.)”, Addison-Wesley Professional, 2015. ISBN-13: 978-0134389455
[30] D. Hermes, “Xamarin mobile application development”, Apress, 2015. ISBN: 978-1-4842-0215-9
[31] F. Kamriani and K. Roy, “App Inventor 2 Essentials”, Packt Publishing, 2016. ISBN-13: 978-1785281105
[32] PCWorld portal. Available: https://www.pcworld.pl (access date: 20.06.2020)
[33] M. van Drongelen, “Android Studio Cookbook”, Packt Publishing, 2015. ISBN-13: 978-1785286186
[34] TensorFlow homepage. Available: https://www.tensorflow.org/ (access date: 15.06.2020)
[35] J. Deng, W. Dong, R. Socher, L. Li, K. Li, L. Fei-Fei, "ImageNet: A large-scale hierarchical image database", *2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL*, pp. 248-255, 2009. DOI: 10.1109/CVPR.2009.5206848

Sławomir Przyłucki[1], Paweł Powroźnik[2]

MANAGEMENT OF FAIR ALLOCATION OF NETWORK RESOURCES IN STREAM SERVICES BASED ON SDN PRINCIPLES

Keywords: streaming services, Software Defined Networks SDN, Dynamic Adaptive Streaming over HTTP (DASH), Server and Network Assisted DASH (SAND), fairness resource allocation.

Abstract

With the spread of real-time streaming services, the problem related to the management of the network resources appeared. The characteristics of streaming services are different from the typical data transmissions on the Internet and therefore require other, usually more advanced, methods of managing network traffic. Software Defined Networks (SDN) is one of the tools that can be used in this context. SDN is an emerging architecture used to build next-generation networks, specially fitted to the management of network environments characterised by frequent changes of transmission conditions. The paper is focused on a video streaming domain and analysis of Quality of Service (QoS) mechanisms in the SDN. It proposes a solution of fair resource allocation, compatible with the Server and Network Assisted DASH (SAND) standard. The structure of the system is described, and results of tests conducted on a testbed are discussed. Also, the parameters and features of our system have been compared witch well-known techniques of traffic queuing and network resource allocation. It allowed to generalise the obtained results and link them to typical real-time streaming scenarios in end-user networks.

1. Introduction

One of the most commonly used types of content on the Internet is multimedia content and video streaming is ahead. At the same time, we are witnessing the rapid development of consumer electronics and Information and Communications Technology (ICT), which allows implementing increasingly complex systems of streaming video services. The need to meet the requirements resulting from this state of affairs led to the creation of the Dynamic Adaptive Streaming over HTTP (DASH) standard. It was published by IETF in 2012 [1].

In the most general terms, DASH defines how to implement two concepts: adaptation of video stream bitrate to minimise packet buffering time and video data encoding in several predefined resolutions and bit rates. According to the DASH standard, each client can "order" the best representation of video material,

[1] Technical University of Lublin, Dept. of Computer Science, Nadbystrzycka 36B Str., 20-618 Lublin, Poland, s.przylucki@pollub.pl
[2] Technical University of Lublin, Dept. of Computer Science, Nadbystrzycka 36B Str., 20-618 Lublin, Poland, p.powroznik@pollub.pl

adapted to the technical parameters of the receiver, and the conditions of the network. The decision to choose a specific representation can be changed during video transmission [2]. Video streaming systems based on DASH have become the basis for the functioning and development of numerous modern video streaming service platforms.

Despite the rapid and wide adaptation to the DASH standard, it has not solved all the problems encountered by users of Internet streaming services. One of the main problems is emerging due to the fact that clients may incorrectly estimate available capacity on the transmission path, which leads to either underutilisation of the network resources or congestions. Research and proposed solutions to this problem are related to the concept of fair distribution of available network resources between individual competing video streams [3]. In a typical case of inability to meet the requirements of individual video streams, the concept of fairness means a way of distributing the available bandwidth to obtain an even degree of degradation of the quality of the received video for all recipients, regardless of the parameters of the receiving device used. To address the described above issues, in 2016 MPEG published an extension to the DASH standard called Server and Network Assisted DASH (SAND) [4]. Thanks to the extensive structure of message exchange between devices participating in video streaming, this extension gives the possibility of effective monitoring of selected parameters and collecting statistics on the status of the service. This data can then be used by a whole range of new generation algorithms to guarantee the fair allocation of network resources.

The question arises whether the SAND standard allows obtaining significantly better results than the currently used resources management algorithms in the end-user network. In other words, if and what improvement in resource allocation can be obtained based on the SAND solution in a typical scenario of many clients simultaneously using a video streaming service. This article presents an analysis of the system based on the SAND standard and the Software Defined Networks (SDN). In parallel, the same network infrastructure was used to implement the video streaming service in accordance with the legacy methods. In this case, the widely known Hierarchical Token Bucket (HTB) algorithm and commonly used network traffic queuing methods were responsible for resource allocation management. The combination of these solutions allowed the analysis of their properties concerning the reception of typical video content.

The rest of the article is organised as follows. Section 2 reviews the related works. Section 3 introduces and describes the components of the SAND standard. Sections 4 and 5 evaluates the performance of selected mechanisms for the allocation of system resources. Finally, section 6 provides concluding remarks as well as directions for future work.

2. Related Papers

The key issue when designing and testing any solutions related to video streaming is defining or adopting a set of metrics that allow for ongoing assessment of the quality of received content. Using the most general classification, objective or subjective assessment methods can be used for this task [5]. In research on DASH, systems dominate the first of these solutions, i.e. objective methods. This is mainly due to the possibility of using a whole range of metrics supported by online data monitoring and analysis methods. Most authors of publications try to propose methods bonding measurable parameters in a given network segment (buffer usage level, available bandwidth, delay) with the parameters of the streaming video (resolution, bitrate) in order to estimate the current value of the Quality of Experience (QoE) metric. Conceptually, the simplest approach is to compare the original video with the received video content, and based on that, assess the quality degradation [6]. It should be emphasised, however, that the specificity of DASH systems operation is based on cyclical segment transmission. Therefore, no element on the transmission path (except the server) has knowledge about the entire video material and only about a fragment of it [7]. For this reason, modifications to typical quality metrics are much more commonly used in video streaming research. These modifications involve the use of a segment bitrate as one of the fundamental parameters within the metric which characterise the quality of the received video. An important factor that also affects the assessment of QoE is the consideration of factors such as frequency of changes in the received video bitrate during the selection of subsequent segments, or buffer overflow causing the so-called stalls. Generally, these factors can be attributed to the specificity of human video reception. An example of their inclusion in the QoE estimation process is the UFair [8] model.

The second area of research, which is related to the topic of this article, are algorithms that aim to improve (maximise) selected QoE metrics in given network conditions in which the streaming service is performed. As mentioned earlier, the introduction of the SAND extension has significantly expanded possibilities in this area. However, in typical Internet provider networks or home networks, dedicated hardware, and complex algorithms for monitoring and traffic shaping, could not be easily applied [9]. That is why solutions based on the idea of Software Defined Network (SDN) are getting more and more attention [10]. The main advantage of SDN is the separation of the network management layer from the data transmission layer. Thanks to this, relatively simple and quick software implementation of advanced solutions is possible, even those with SAND. At the same time, thanks to virtualisation methods it is possible to avoid most hardware restrictions. According to previous information, the DASH standard does not define the algorithms responsible for network management, in particular for the proper allocation of resources [11]. Such solutions are proposed externally to the DASH standard. For example, in the reference open-source video player DASH-

JS [12] an algorithm called Buffer Occupancy based Lyapunov Algorithm (BOLA) is implemented. Its operation is based on the definition of utility function whose parameters are average bitrate and stall ratio. The BOLA algorithm provides methods to maximise the value of the utility function defined in this way. In turn, solutions based on a SAND focus on a centralised algorithm that allocates network resources for different video flows, taking into account the maximum fairness of this process.

3. The SAND Standard

The DASH standard describes mechanisms suitable to transfer segmented video content over HTTP [4]. It also defines the segmented data structure, stored on the server DASH as dedicated Media Presentation Description files (MPD). DASH clients, based on the information contained in these files, can decide on requested video representations (segments) during the video playback. Unfortunately, the DASH standard does not contain mechanisms related to the mutual influence of the competing video streams. This means that DASH clients are securing (they try to), in a very selfish way, the highest available bandwidth what usually corresponds to the best quality of the received video. From the points of view of the network with multiple clients, it leads to suboptimal streaming performance.

To address the mentioned above challenges, the extension of the DASH standard was proposed. It is called SAND and its main assumptions can be summarised as follows:

- SAND offers streaming enhancements based on a few collaborating mechanisms which allow analysing and processing feedback from clients. It gives a way to develop an intelligent system of service management on the server/network side.
- SAND provides a set of dedicated messages in order to improve the adaptation of the client. This leads to the development of an intelligent client/controller side.

The SAND reference architecture is presented in Figure 1. The SAND reference architecture contains four types of components:

- DASH streaming clients,
- Regular Network Elements (RNE), which are loosely related to the streaming process because they are on the path of video transmission but they treat DASH-relate video streams like any other transmissions inside the network.
- DASH-Aware Network Elements (DANE), which are able to prioritize, parse or modify objects related to DASH service (MPD files or DASH segments),
- Metrics server, which is responsible for gathering data from DASH clients.

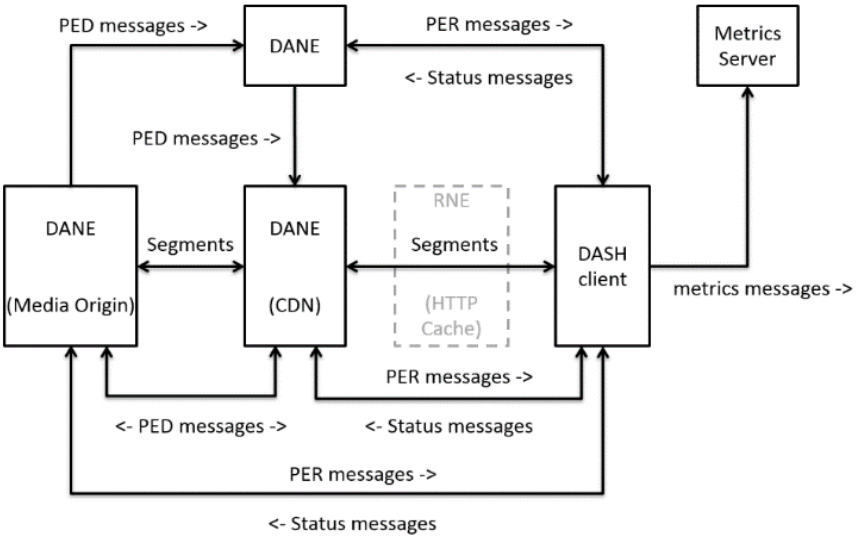


Fig. 1. The SAND reference architecture [4].

All the mentioned components exchange the defined set of od messages. According to SAND standard, these messages can be categorised into four main groups:

- Parameters Enhancing Delivery (PED): exchanged between DANEs,
- Parameters Enhancing Reception (PER): sent from DANEs to DASH clients,
- Status messages: sent from DASH clients to DANEs,
- Metrics messages: sent from DASH clients to Metrics servers.

Using the above-described assumptions of the SAND standard, the implementation of the mechanism of fair allocation of network resources in the video streaming service has been developed.

4. Benchmarking Methods and Tools

4.1. SAND Implementation

To carry out comparative tests, a system of fair allocation of available bandwidth was developed in a typical network of Internet service providers or an extensive home network. Conceptually, this solution is in line with our earlier research into DASH systems [13]. It combines the functionality of the SDN network, and the mechanisms defined in the SAND standard. Therefore, later in the article, this solution will be referred to as S-SAND (SDN for SAND). The structure of the S-SAND system consists of four functional components:

- Network resources module (NR).
- Fairness allocation module (FA).
- Traffic shaping module (TS).

The NR consists of two functional components, respectively: monitoring and policy generator. The main task associated with this module is gathering information about all the traffic within a network. The monitoring component identifies all active stream and stores their parameters (MDF content, bitrate of the last, and current video segment). The policy generator, in the general case, can include any policy declared in Service Level Agreement (SLA) for DASH service. In the scenario used during tests, it contained bandwidth limits.

The TS module is responsible for defining and processing the rules for each active video stream. The parameters required by these rules are provided by the FA module. The processing of the rules is implemented as a set of OpenFlow instructions and performed on vSwitch.

From the point of view of topics covered in the article, the most important element of the system is the Fairness distribution module (FA). It contains two databases, respectively: DASH client database and video stream database. They directly use the Status and Metrics messages which are exchanged between FA and DASH clients. The utility function (UF) implemented inside the FA module is the non-linear function that binds the video stream bitrate and utilisation factor (UF). A more detailed description of the function can be found in our earlier research [14]. In turn, the UF ratio is the ratio of the number of representations that the DASH client can order (e.g. due to limitations resulting from the player's technical parameters) to the number of all video representations contained in the segments described in the MDF file. The final task of the FA component is to find a set of bitrates Y_{op} associated with active streams that maximise fairness. It can be formulated as follows:

$$Y_{op} = max[min_i(UF_i(x_i))] \quad \sum(x_i) \leq B_{limit} \tag{1}$$

where:

- UF_i is a utility function,
- x_i represents the bitrate for an i-th video stream,
- B_{limit} represents the total amount of available bandwidth for all streams.

Formula 1 is solved based on the integer-programming algorithm presented in [15]. The calculated value is sent to the DASH clients as SAND PER Messages (message: SharedResoureAssignment) [4x]. The structure of the S-SAND system and the relationship between its components are presented in Figure 2. The elements which utilise the SAND messages are indicated by black backgrounds.

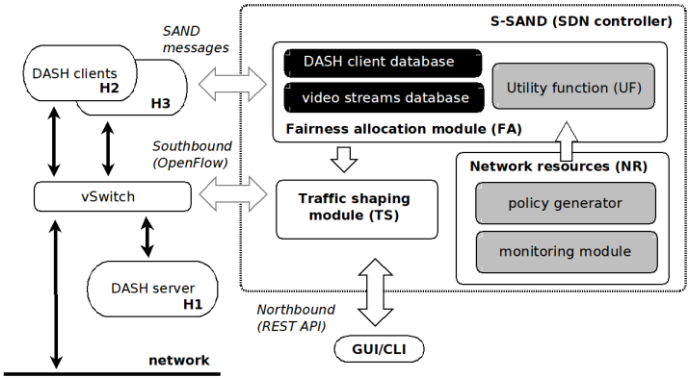


Fig. 2. The architecture of the proposed S-SAND system.

4.2. The Tests Scenarios

The issue of shaping and prioritising traffic in IP networks is not a new topic. However, according to the information presented earlier, the spread of solutions based on the SDN network creates the possibility of more flexible implementation of a wide range of algorithms, including the algorithms responsible for the allocation of network resources. SDN allows the implementation of classic algorithms and new solutions adapted to the specifics of a given network and the set of services provided. In the context of streaming video content, one can implement classic packet queuing methods or traffic shaping algorithms (e.g. HTB algorithm). One can also use the DASH standard and its extension SAND to define a new class of solutions. An example can be the S-SAND system presented in the previous section. Therefore, the comparative analysis, which is presented in the further part of the article, includes representatives of the above-mentioned solutions, both classic and dedicated to DASH systems. The relationships between individual solutions and test scenarios are as follows:

- Scenario 1. This scenario implemented a resource allocation based on competition between clients of video streaming services. In other words, it is a solution in accordance with the DASH standard and its reference implementation in the DashJS player [12]. Therefore, no QoS parameters have been set on the SDN controller. DASH clients (H2, H3) compete for resources (delivered from the DASH H1 server) in the same network segment. In this scenario, the test network has been configured to limit the total available bandwidth to 10Mbit/s for each client.
- Scenario 2. In this scenario, the video streams were downloaded by H2 and H3 clients, like during the scenario 1. Besides, however, these streams were allocated in separate queues. The same QoS policy was set for both queues, which guaranteed minimum bandwidth of 2 Mbit/s and a maximum of 10 Mbit/s for each client.

- Scenario 3. This scenario is an extension of the previous test scenario (scenario 2). Two queues, for two clients (H2 and H3) had a common QoS policy based on the HTB algorithm [17]. According to this algorithm, any "free resources" of the available bandwidth from one client could be "borrowed" for streaming by other clients.
- Scenario 4. This scenario took into account the fact that the DASH H2 client represented a higher resolution receiver (1080p) and the H3 client was a lower resolution receiver (720p). Therefore, a separate queue was created for each DASH client, and each queue used a separate QoS policy. The first policy (for the H1 client) guaranteed minimum bandwidth of 5Mbit/s and the second policy (for the H3 client) a minimum bandwidth of 1Mbit/s. In both cases, the maximum bandwidth was 10Mbit/s.
- Scenario 5. This scenario implements the S-SAND solution described in the previous section. To guarantee comparable test conditions, the maximum bandwidth was also limited to 10 Mbit/s for each client.

The popular Mininet network emulator was used to build the network. The OpenDayligth package [16] was the basis for configuring the SDN controller. DASH clients were implemented based on the DashJS reference client [12] and the server based on the gpac-dash [18] package.

All test scenarios were implemented for two selected video types (animation and action movie). Selected video types differ in the nature of the multimedia content contained. Scenes in action films (source video file: Tears of Steel [19]) change dynamically and from the recipient's point of view require maintaining smoothness and detail of subsequent scenes. In turn, the animated film (source video file: Big Buck Bunny [19]) is less demanding in the context of the impact of changes in available bandwidth and usually less susceptible to the occurrence of related artifacts. Both videos consisted of 4-second segments described in the MDF manifest file located on the DASH server.

4.3. The Fairness Assessment

In the above-mentioned scenarios, the Ufair [8] model was used to measure the fairness of the available bandwidth allocation. The architecture of this model incorporates three fairness metrics (video quality, switch-ing impact, and cost efficiency). It is worth noting that this model takes into account the specific properties of human image perception (including the fact that the selection of the next segment with a lower bitrate is more noticeable than the change in the bit rate up). UFair uses contextual data regarding the selected video stream such as its bit rate, the desired video resolution as well as the total network bandwidth and bandwidth available on the given device. The basic relationship describing the quality of video material with a specific resolution that is defined within this model is described by Formula 2.

$$Q_r = aT^b + c \tag{2}$$

where:

Q_r – video quality with resolution *r*.

T – video stream bit rate.

The values of *a*, *b*, *c* coefficients depend on the screen resolution (360/720/1080p) and have been defined in the UFair model as follows:

– resolution 720p: *a=-4.85, b=-0.647, c=1.011*

– resolution 1080p: *a=-3.035, b=-0.5061, c=1.022*

The video quality *Q(T)* is normalised by referring to the highest value of bit rate $Q_r\ (T^{TOP})$ defined in the MPD file for the given video stream.

$$Q(T) = \frac{Q_r(T)}{Q_r(T^{TOP})} \tag{3}$$

The fairness of bandwidth allocation S_{VQ} between given video streams is described by Formula 4.

$$S_{VQ} = \sqrt{\frac{1}{M-1}\sum_{j=1}^{M}\left(Q_j - Q^E\right)^2} \tag{4}$$

where:

M – number of video streams

Q_j – the normalised quality of the i-th video stream

Q^E – average quality value for a given video

The value of the fairness factor I_{VQ} for a given stream is calculated using the Formula 5

$$I_{VQ} = 1 - \left(100 \times \frac{S_{VQ}}{Q^E}\right) \tag{5}$$

5. Test Results

The comparison of results obtained during the realisation of five scenarios presented in the previous chapter allowed for a comparative analysis of typical solutions for allocating network resources during the implementation of the streaming video service. These results are presented in the following sections.

5.1. Scenario 1

The results obtained during the implementation of the 1st scenario can be treated as a reference for all subsequent scenarios. They describe the functioning of the test system in conditions of unrestricted competition between two DASH clients for the available bandwidth. The values of the fairness factor for the two types of video content are shown in Figure 3.

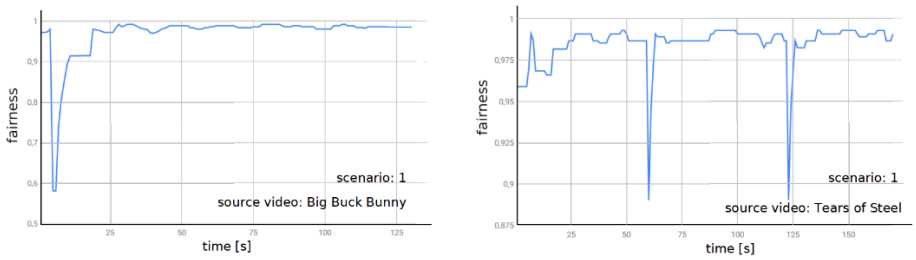


Fig. 3. The values of the fairness factor for the two types of video during scenario 1.

In the case of the animated film "Big Buck Bunny", strong competition occurred in the initial phase of the streaming service. Due to the fact that none of the clients knew the maximum, required bit rate for the transmission, individual attempts to increase the bit rate quickly occurred at the expense of other clients. After downloading about 5-6 segments, the situation stabilised. The relatively small variability of the bit rate for this video meant that there were no more violent periods of competition and the fairness factor remained in a satisfactory corridor of values, from 0.96 to 0.99.

The same algorithm behaved differently in the case of an action movie in which there were frequent changes in bit rate resulting from the specificity of the transmitted video content. In this case, one can also observe the initial strong competition. However, its effects were visible throughout the entire transmission with several episodes of an extremely strong decrease in the value of the fairness factor. These episodes occurred at a time of significant increase and then a decrease in the bit rate of the received video. Adaptation to these changing requirements has intensified the competition for available bandwidth between DASH clients.

5.2. Scenarios 2 to 4

The goal of the next three scenarios was to assess whether and to what extent classic management methods can reduce the undesirable phenomena observed in scenario 1. The results of tests carried out under individual scenarios are presented in Figure 4.

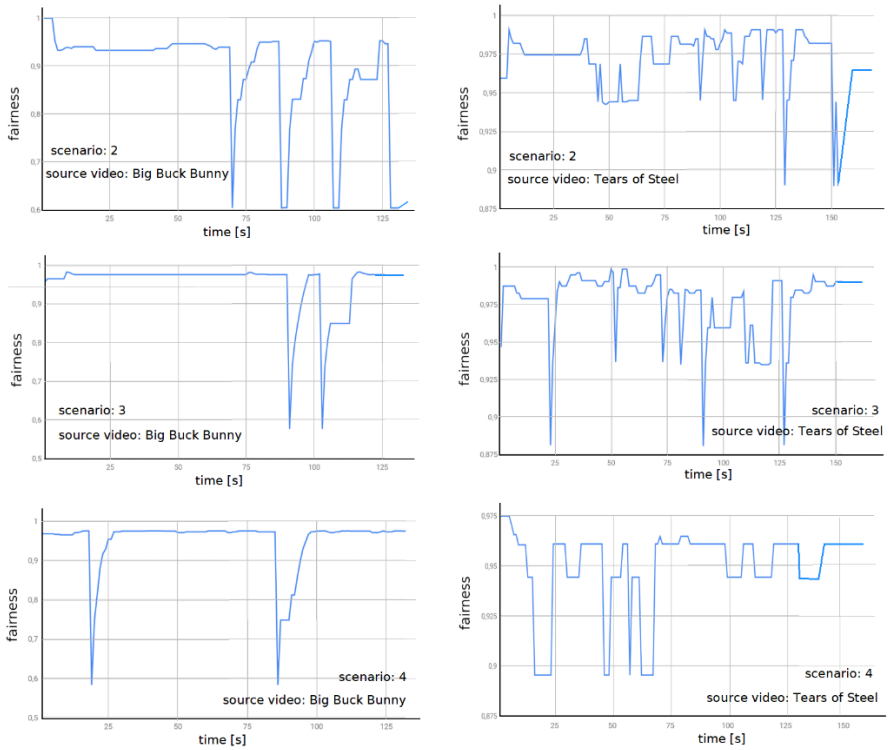


Fig. 4. The values of the fairness factor for the two types of video during scenario 2 to 4.

The results obtained during tests performed according to scenarios 2 to 4 do not allow to state that the use of classic QoS mechanisms improves the conditions of competition for available network resources. In all tested cases, the egoistic pursuit of increasing the bandwidth used by DASH clients led to a sharp decline in the value of the fairness factor. This was even the case with the video containing the animation, which in scenario 1 did not cause such large fluctuations in the value of the fairness factor.

It should also be noted that in all tests, the results of which are presented above, the number of changes in the bit rate between successively downloaded video segments was higher by 28% for the video film and by 43% for the action film compared to scenario 1. From our subjective observations conducted during tests, this had a negative impact on the perceived quality of the received video.

5.3. Scenario 5

The last of the tested scenarios were based on a test system in which the SDN controller received data on the ordered video materials as well as the current status of the allocation of network resources for active DASH clients. This information

was provided in accordance with the SAND standard described in the previous chapter. The results obtained for this scenario are shown in Figure 5.

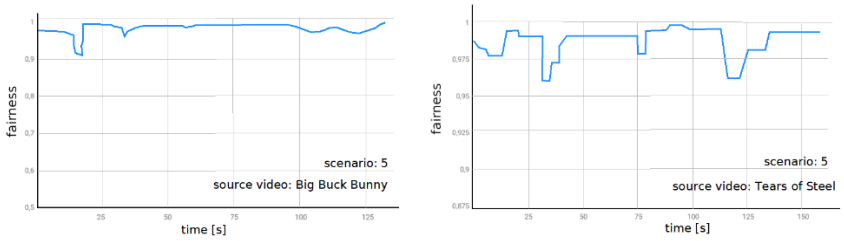


Fig. 5. The values of the fairness factor for the two types of video during scenario 5.

The basic observation that results from the tests carried out is the fact that regardless of the type of video material, fluctuations in the value of the fairness factor are smaller. This was also reflected in the number of switching events (bit rate changes) recorded during the implementation of individual streaming. For video containing animation, the number of switches was as much as 62% lower than for transmissions carried out under scenario 1. For the video containing the action movie, this improvement was smaller and averaged 27%.

It should also be emphasised that the direct implementation of the basic messages defined in the SAND standard is extremely sensitive to the relationship between the frequency with which these messages are transmitted and the time of calculating the set of optimal bit rates for DASH clients. We have found that based on our hardware platform, this relationship begins to significantly affect the operation of the system with 8 or more active clients.

6. Final Conclusions

In the last several years, many standards and mechanisms have been proposed for managing resources in IP networks. The vast majority of them were developed with conventional internet services in mind. The rapid growth in popularity of real-time services, including video streaming services, has led to the need to develop new solutions or to define new rules for the use of already known solutions. An important limitation in this area is the possibilities offered by currently used network devices. This is particularly evident in the networks of small Internet service providers and an increasing number of home networks. An important breakthrough in this respect is the spread of SDN networks, which allow for a much more flexible configuration of network resource management policies.

The article analyzes a set of solutions based on the SDN controller, which can be relatively easily implemented in local networks. The obtained results indicate that in the case of streaming video services, classic QoS solutions do not improve the functioning of the computer network in terms of the fair allocation of network resources. The obtained results for test scenarios based on classic packet queuing

methods did not confirm the advantages of these solutions in the case of video streaming. The values of the fairness factor were on the one hand much lower than in the case of the reference mechanism of DASH clients competition and on the other hand, showed much greater volatility. However, the best results in terms of fair allocation of resources were obtained based on the SAND solution. By taking into account the current state of the transmission network together with information about the streamed video content, it was possible to obtain the fairest allocation of network resources and minimise the number of bit rate switches between successive segments of downloaded video content.

The conducted comparative analysis proves that if SDN mechanisms are available, there is a relatively simple and effective method of ensuring a fair distribution of network resources among clients of the video streaming service based on DASH. This solution is to use the mechanisms contained in the SAND standard. However, the subject of further research should be the selection of such algorithms, using SAND, which, based on the available network and computer equipment, will allow for fair management of network resources in systems with several or more DASH clients.

References

[1] ISO/IEC 23009-1:2014, Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats. http://standards.iso.org/ittf/PubliclyAvailableStandards/c065274_ISO_IEC_23009-1_2014.zip (Accessed:19 June 2020)

[2] Stockhammer, T.: Dynamic Adaptive Streaming over HTTP Standards and Design Principles. In: Second Annual ACM Conference on Multimedia Systems, ACM, 2011

[3] Akhshabi, S., Begen, A., Dovrolis, C.: An Experimental Evaluation of Rate-adaptation Algorithms in Adaptive Streaming over HTTP. In: Second Annual ACM Conference on Multimedia Systems, ACM, 2011

[4] ISO/IEC FDIS 23009-5:2016: Information Technology. Dynamic adaptive streaming over HTTP (DASH) Part 5: Server and network assisted DASH (SAND), 2016

[5] Cermak, G., Pinson, M., Wolf, S.: The Relationship Among Video Quality, Screen Resolution, and Bit Rate. In: IEEE Trans. Broadcast., vol.57, 2011

[6] Daronco, L.C., Roesler, V., Lima, J.V.: Subjective Video Quality Assessment Applied to Scalable Video Coding and Transmission Instability. In: ACM Symposium on Applied Computing, ACM, 2010

[7] Z. Li, A. C. Begen, J. Gahm, Y. Shan, B. Osler, and D. Oran,: Streaming video over HTTP with consistent quality, In: Proc. 5th ACM Multimedia Syst. Conf., 2014

[8] Mu M., Broadbent M., Farshad A.,: A Scalable User Fairness Model for Adaptive Video Streaming Over SDN-Assisted Future Networks, In: IEEE Journal on Selected Areas in Communications, vol. 34, 2016

[9] Chang, X., Li, J., Wang, G., Zhang, Z., Li, L., Niu, Y.,: Software defined backpressure mechanism for edge router. In: 2015 IEEE Proceedings of the 23rd International Symposium on Quality of Service (IWQoS), 2015
[10] Durner, R., Blenk, A., Kellerer, W.,: Performance study of dynamic QoS management for OpenFlow-enabled SDN switches. In: Proceedings IEEE IWQoS, 2015
[11] M.-N. Garcia et al.,: Quality of experience and HTTP adaptive streaming: A review of subjective studies, In. Proc. 6th Int. Workshop Quality Multimedia Exper. (QoMEX), 2014
[12] DASH Reference Client http://reference.dashif.org/dash.js (Accessed: 7 june 2020)
[13] Przylucki, S., Czerwinski, D.: The ability to ensure a fair distribution of network resources for video streaming services based on DASH standard, In. Studia Informatica, vol.36, 2015
[14] Przylucki S., Czerwinski D. and Sierszen A.,: QoE-Oriented Fairness Control for DASH Systems Based on the Hierarchical Structure of SVC Streams, In: Proc. of International Conference on Computer Networks (CN 2016), 2016
[15] Dakin R.J.: A tree-search algorithm for mixed integer programming problems, The Computer Journal, vol.8, 1965
[16] OpenDayLight "Magnesium" https://www.opendaylight.org/ (Accessed: 26 May 2020)
[17] Balan D., Potorac A.: Extended Linux HTB Queuing Discipline Implementations, In. International Journal of Information Studies, Vol 2, 2010
[18] GPAC Multimedia OpenSource Project https://gpac.wp.imt.fr/category/gpac/dash/ (Accessed: 22 May 2020)
[19] ITEC – Dynamic Adaptive Streaming over HTTP, https://dash.itec.aau.at/download/ (Accessed: 12 May 2020)

Stanisław Skulimowski[1], *Jerzy Montusiewicz*[2]

OPTIMIZATION METHODS OF SEARCHING ALGORITHMS FOR 2D ELEMENTS MATCHING

Keywords: element matching, string codes, linguistic metrics, algorithm optimization, exhaustive recursive search

Abstract

The problem of matching 2D and 3D elements is an issue that affects many fields of science, such as archaeology or medicine. The article concerns devising an original method of constructing effective algorithms for matching 2D elements that would allow to automate the process of arranging broken archaeological and museum elements.
The described method of matching 2D elements is based on the contours data in the form of string codes, where each character reflects both length and direction. Linguistic metrics were used to compare the elements. Furthermore, the developed algorithms perform an exhaustive recursive search. The results of numerical experiments showed that the time of calculations, and thus the number of potential element matching, increases along with the number of elements and with the length of the compared substrings. Additionally, it depends on the uniqueness of the fragments of the contours of each element.
The applied optimizations which included changing the length of the compared substrings and replacing elements random sampling with ordered sampling, resulted in both a reduction in computation time as well as increasing the knowledge about the functioning of the method and algorithm.

1. Introduction

Various methods which allow matching 2D and 3D elements in different areas: forensics [1], medicine [2, 3] or archaeology [4], can be found in contemporary literature. Usually, the developed methods are tailored to solve specific problems and require different data about the objects to be connected with each other, such as the information about the contour and colour [5], about the edges of the elements [6, 7] or about texture and geometry [8]. For pottery objects, from a known cultural area and with a symmetrical axes, the information about the curvature signature can be used to arrange them, provided that someone has previously developed their database [9]. With a large number of elements that can be used to assemble complete or almost complete museum exhibits, we are dealing with a very high complexity – these are NP-complete problems [10]. In the

[1] Lublin University of Technology, Department of Computer Science, Nadbystrzycka 36B 20-618 Lublin, Poland, s.skulimowski@pollub.pl
[2] Lublin University of Technology, Department of Computer Science, Nadbystrzycka 36B 20-618 Lublin, Poland, j.montusiewicz@pollub.pl

previous works, the concept of defining the value of change of short vector angles placed on the outline of the element along the whole circumference [11] or placed vectors of different lengths and averaged directions [12], was used to describe the edges. There have also been studies which, in order to describe the contour of various types of objects, resorted to string codes: language [13], technical [14] or images [15]. However, the coding techniques used in them do not contain full information about the size or length of individual objects, but rather the qualitative information about the shape.

The article concerns an original method of constructing special mechanisms in the algorithm allowing to connect 2D objects [16]. The algorithm enables multiple searches for the possible ways of arranging elements of broken 2D objects through recursive action. The authors have proposed their own method of 2D objects arrangement based only on the contour information of individual elements, which have been saved in the form of strings codes from an acceptable character set. Individual characters reflect both the length and direction of the element outline and create abstract words with different number of characters. Linguistic metrics have been used to compare pairs of two elements, as in the case of comparing the similarities between classic words written in any language. The developed algorithm performs an in-depth, recursive search of a set of 2D elements from the initial pool using the Levenshtein metric.

In many situations, the activities related to arranging parts of the object into a whole are usually performed manually, without the support of additional hardware and software. Therefore, the specialists who deal with this issue on a daily basis have developed the methodologies for dealing with sets of such elements. One of the main principles used in the assembly of elements is the initial sorting of elements by their size. Most often it is assumed that the set of elements should be arranged starting from the largest available fragments, whereas element matching should start from the largest ones and end with the smallest ones.

The aim of this article was to optimise the operation of the bonding method by examining the impact of the change in:

- the lengths of the sequences to be compared in the process of comparing two elements in pairs,
- random sampling of elements from the initial pool to orderly sampling from the largest to the smallest, and from the smallest to the largest elements,
- Levenshtein's assessment limit value from 0 to 1,

for the duration of the calculation process, and thus the number of possible alternatives and memory consumption.

2. Methods and materials

2.1. Method of making string codes and elements arranging

While conducting the initial research on the effectiveness of the constructed algorithm and its implementation, an eight-element character rose was used to describe the contour of the elements, see Figure 1.

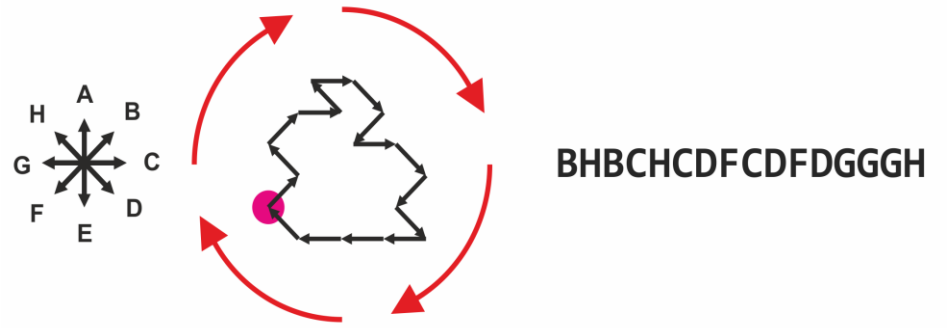


Fig. 1. Example of the element transformation, from the left: (1) the wind rose used, (2) the contour of the object divided into sections of length and direction in accordance with the wind rose, (3) result string code.

Creating a string code starts from the point selected by the authors and also progresses in the chosen direction. For each subsequent fragment of the object contour, its similarity to the wind rose directions is checked and then a sign corresponding to the direction is assigned. After this operation, for all consecutive fragments of the segment, the whole contour is checked, and as a result the whole object obtains an equivalent and invariant string code in the form of a sequence of characters. It is worth mentioning that shifting the starting point to a new position in the direction of the contour description will cause these characters to appear at the end of the string and will not significantly change the structure of the entire string code.

The objects prepared in this way can be compared with each other using linguistic methods – a detailed description is found in the article [16]. In the program employed in the experiment, the Levenshtein method was used, describing the difference between two texts based on deletion, insertion and substitution. Greater divergence between the two texts results in a higher value of the Levenshtein measure (LM).

Fragments of string code characters, to be compared using the Levenshtein method, are selected in the process of searching for solutions using wrapping (comparing contours sections), see Figure 2. Since the tested algorithms check the substrings that are moved by one character along the entire string code, it does not affect finding the matching elements.

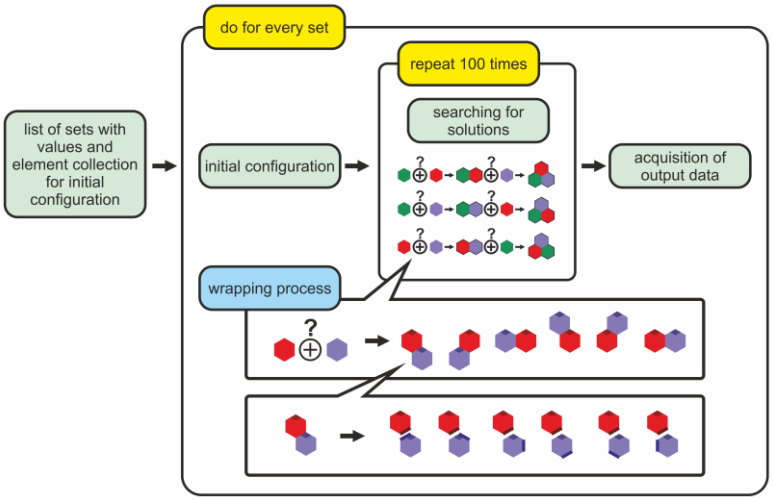


Fig. 2. Simplified block diagram of the program used to conduct the experiment.

2.2. Description of the experiment

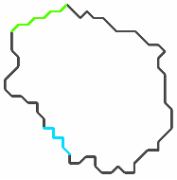
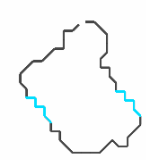
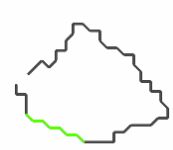
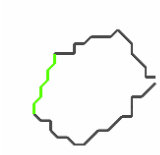
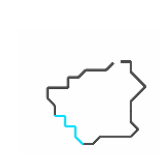
The aim of the numerical experiment was to examine the impact of the possible search options, e.g. changing the length of the analysed strings, the way of taking the elements to be compared, the acceptable value of nonconformity determined by the Levenshtein measure, on the number of alternative solutions found. While testing the developed algorithm to connect the elements with each other, the calculation time, computer memory consumption and the number of considered assemblies were determined. The experiment was also supposed to help determine whether certain search variants could give better results, in terms of the number of unique and more compact solutions. The experiment consisted in rerunning the prepared program with new values of decision variables for the same set of the initial pool.

2.3. Description of the tested set

While conducting the initial research on the effectiveness of the constructed algorithm and its implementation, an eight-element character rose was used to describe the contour of the elements, see Figure 1. For testing, a set of artificially prepared elements was used, which means that the contours do not reflect the shapes of real museum or archaeological artefacts, see Table 1. The graphical representation of the prepared elements may create unclosed contours, which is connected with the limitations resulting from the resolution of the compass rose, see Figure 1 (the total length of vectors does not always coincide with the actual length of the element's circumference). The algorithm to compare

elements in order to reduce the possibility of their erroneous bonding does not take into account two extreme vectors (first and last) from the starting point.

Table 1. Character representation of object edges with the number of characters, pi.

Object number	Object graphic representation	Object's contour string codes
12		GFFEGHFGHAGFGAHAGAGAHAGHGHABBA HAABCBCBCBDDBDCDDCCDDCDFEDEEEDE EFFE, 65 characters blue substring – 5 characters green substring – 7 characters
2		CDDEEFECEDECECEDEFFEGHFFGGGHHAGA HAGAGAHABBCBBCAACB, 50 characters
1		BBDBACABACDCEDCDCEDECECEDEGFGGF GEGGGHGHGHGHAAGA, 47 characters
3		FFGEEGFFGFFGHGHAGHABABABACCACBC CBDDEDDEC, 40 characters
4		CEDDFFEEDFGGGFGHAGAGAHACCACBCCB , 31 characters

The set has been prepared in such a way that its elements differ significantly in the number of characters describing the outline. The lengths of the strings ranged from 32 to 65 characters, which gives more than twice the difference in their length. Moreover, the elements were characterised by uniqueness of shape, which means low repeatability in relation to the fragments of other elements. In order to make the process of sticking together more difficult, some

substrings of characters describing the contour of particular objects were present in many elements, which made it possible to generate alternative sticking solutions, e.g. strings visible at the first glance: a predictive substring of 10 characters and a blue of 5 elements. Of course, there are more such recurring substrings (e.g. elements 1, 3 and 12).

The description of their shape with the use of the adopted compass rose looks different in the form presented in Table 1., but after the transformation consisting in the compass rose rotation by angles being a multiple of 45°, identical sign strings are obtained (it results from the applied principle of invariant characterisation).

2.4. Research scenarios

The numerical experiment consisted in arranging together the available elements when entering different values of decision variables into the algorithm. The possible variants of sticking together for the following variables were sought: P1 – ordering a set of initial elements, P2 – length of compared fragments in elements (substrings), P3 – acceptable margin of error of matching the elements measured by the Levenshtein measure. The following values were adopted for individual variables.

- P1 – ordering the set of initial elements: value 1 – random element downloading, 2 – downloading from the shortest to the longest element, 3 – downloading from the longest to the shortest element.
- P2 – length of compared string fragments in elements: values equal to: 5, 6, 7, 8, 9, respectively (number of consecutive string characters in elements),
- P3 – the acceptable margin of error for the matching elements, values can take numbers starting from zero depending on the Levenshtein measure value. For example, when the calculated Levenshtein measure is zero (ML = 0) – the so-called tooth to tooth matching exists or if the value is one (ML = 1) – there is one error in matching strings. Potential matches are always sorted in ascending order of the achieved ML value. This means that matches with a given ML value are only taken into account when checking arrangements if the matches with a lower ML value are missing or have already been checked before.

On the basis of the assumed values of variables, a list of test scenarios (S) was prepared, consisting of 30 possible combinations of values of individual variables ($|P1| * |P2| * |P3| = 3*5*2= 30$).

For each of the tested experimental scenarios, the bonding program calculated the following values:

- R1 – number of comparisons made,
- R2 – number of solutions found,
- R3 – average time of program execution [ms],

- R4 – average memory consumption needed to perform calculations for the realised scenario [kB], calculated with Visual Studio (version 16.6.5) program diagnostic tools as isolated memory usage snapshots with no IDE overhead intention.

These values allow evaluating the efficiency of the developed algorithm for the assumed values of variables.

3. Results and discussion

The detailed results of the conducted experiment for all 30 scenarios are presented in Table 2. The table presents averaged data (R3, R4) for 100 repetitions of each scenario (Fig. 2.).

Table 2. A list of variables differentiating the scenarios and results obtained after testing each of the scenarios.

S	P1 [-]	P2 [-]	P3 [-]	R1 [-]	R2 [-]	R3 [ms]	R4 [kB]
1	random	5	0	312199	23	274	115
2	shortest to longest	5	0	308203	22	205	110
3	longest to shortest	5	0	242079	18	154	94
4	random	6	0	250674	17	123	91
5	shortest to longest	6	0	282828	20	142	100
6	longest to shortest	6	0	192281	14	77	79
7	random	7	0	174167	9	72	69
8	shortest to longest	7	0	203953	12	81	75
9	longest to shortest	7	0	147717	8	61	63
10	random	8	0	167421	10	75	69
11	shortest to longest	8	0	208381	13	89	77
12	longest to shortest	8	0	146556	9	67	65
13	random	9	0	49603	3	22	46
14	shortest to longest	9	0	49603	3	21	46
15	longest to shortest	9	0	42391	2	19	246
16	random	5	1	713649	60	687	233
17	shortest to longest	5	1	700799	60	760	246
18	longest to shortest	5	1	710739	60	645	172
19	random	6	1	449158	38	394	173
20	shortest to longest	6	1	451476	38	404	167
21	longest to shortest	6	1	425211	37	323	104
22	random	7	1	277431	21	151	110
23	shortest to longest	7	1	284173	22	163	114
24	longest to shortest	7	1	301071	23	162	79
25	random	8	1	205301	12	97	81
26	shortest to longest	8	1	211477	13	101	84
27	longest to shortest	8	1	223434	14	105	79
28	random	9	1	196655	12	106	82
29	shortest to longest	9	1	199621	13	100	78
30	longest to shortest	9	1	192163	11	103	157

Source: own study

3.1. Results with a change in length of the substring being sought

The analysis of the results obtained in the experiment indicates the existence of a general downward trend in the amount of used memory (R4) depending on the length of the compared element strings (see Figs. 3 and 4). This trend is maintained for most cases, regardless of the values of the P1 and P3 variables. The exception are the scenarios S15 and S30, in which the calculation uses the order of elements from the longest to the shortest (P1) with the highest assumed length of compared strings (9 characters). It may indicate reaching the critical value for input variables – excessive complexity of calculations does not bring measurable effects for a relatively simple set of tested elements.

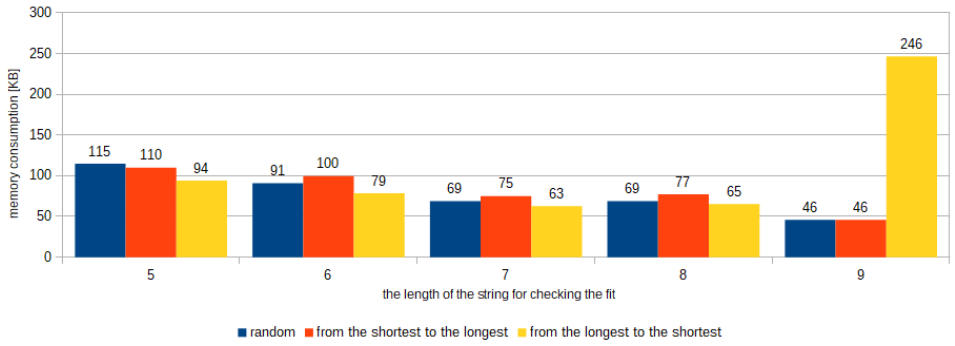


Fig. 3. Influence of the length of a substring for checking the elements' compatibility on the memory consumption for ML=0.

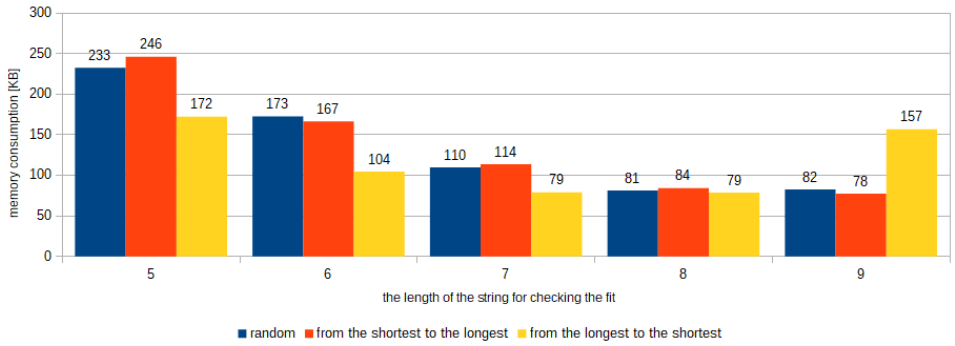


Fig. 4. Influence of the length of the substring for checking the compatibility of elements on the memory consumption for ML<=1.

There was also a downward trend in the average time of scenario realisation (R3) with the increasing length of the compared element strings (Figs. 5 and 6).

The graphs presented in Figures 4–7 show that the increase in value for the P3 variable resulted in a significant increase in memory consumption (R4) and

time of scenario execution (R3). The increase in the value of this parameter also caused an increase in the number of solutions (R2). Unfortunately, it also resulted in a significant increase in the number of comparisons made (R1).

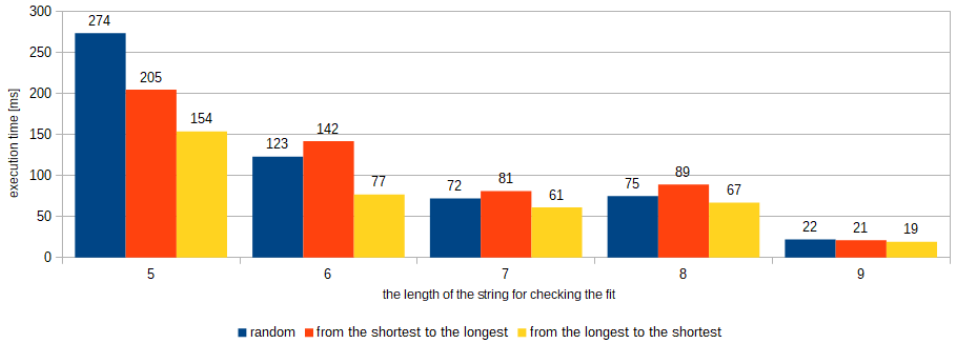


Fig. 5. Influence of the length of the substring to compare the compatibility of the elements on the time of the solution search process for ML=0.

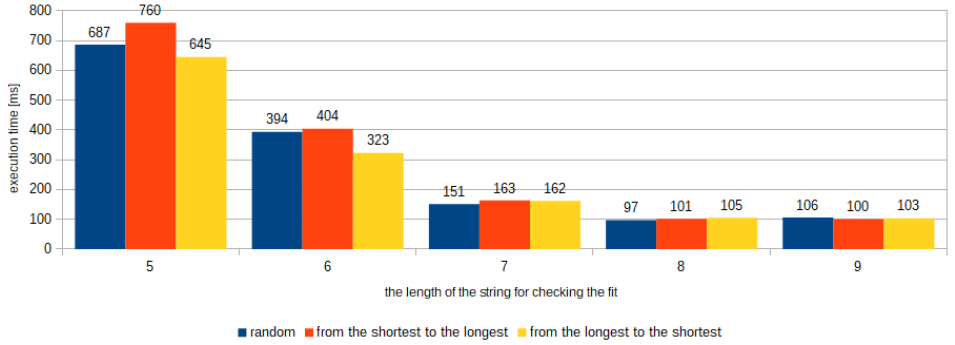


Fig. 6. Influence of the length of the substring to compare the compatibility of the elements on the time of the solution finding process for ML<=1.

While comparing the obtained values for the values of R1, R3 and R4, it turned out that the most complex scenarios are those in which the length of the compared substrings is shortest (P2=5) and the ML value can take the value of 1. These are scenarios S16, S17 and S18, representing 3 different ways of sorting elements (P1). Using a low value of P2 resulted in comparing the fragments that may not be quite unique, while a high value of P3 means an increase in tolerance in the comparison process. As a result, only seemingly matching element lists were created. Such approach guarantees the possibility of obtaining a large number of results (which is confirmed by the R2 values for the aforementioned scenarios), while the quality of these matches remains a matter of dispute.

While analysing the output values, one can see a correlation between the values obtained by R1, R2 and R3. With the increase in the number of comparisons

necessary to make, the time needed for calculations and the number of solutions found increases (Fig. 7). The values R1, R2 and R3 are the higher when P2 is lower and P3 is higher.

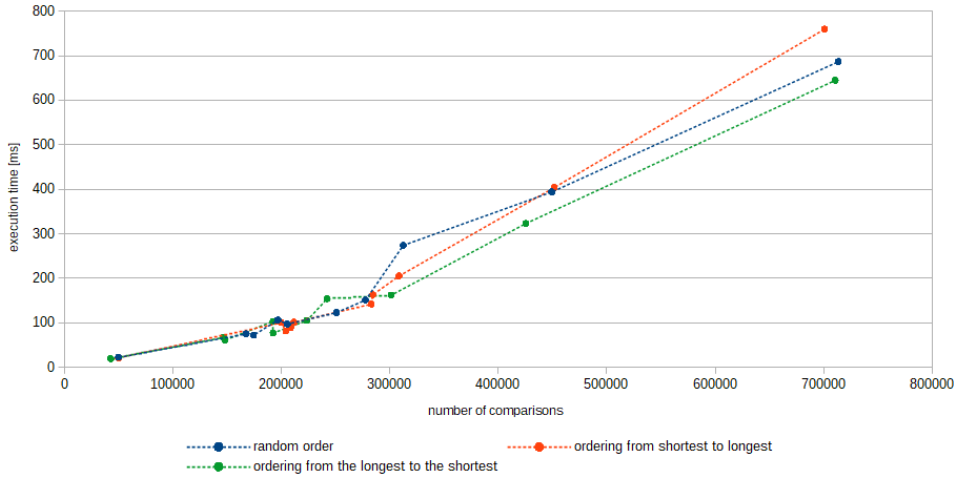


Fig. 7. The relation between the time needed to solve the scenarios as a function of the number of comparisons necessary to make.

3.2. Results including sorting of elements

While comparing the obtained values for the quantities R1, R2, R3 and R4 for all the accepted methods of sorting the initial elements, it can be concluded that the lowest values are achieved when conducting the arrangements for a set sorted from the largest to the smallest element. The tested algorithm has managed well with this setup. The largest element, no. 12, was an element that was out of line with the other elements, which means that in the correct arrangement of elements from the analysed pool, it will not be the main element. Thus, starting the arrangement process from the largest element from the available pool could introduce some interference into the algorithm.

However, on the basis of the data collected it is not possible to clearly determine whether sorting has an influence on improving the quality of results (obtained arrangements) – the number of realised comparisons increases with the decrease of P2 and the increase of P3 similarly for all types of element arrangement. Moreover, some of the solutions found when searching for a randomly arranged or sorted crop are repeated. For example, for P2 = 5, P3 = 0, and for a random arrangement 23 solutions were found, but only 4 of them did not appear in the other sorting, from the largest to smallest arrangement – only 3 unique solutions out of 18, from the smallest to largest arrangement – only 4 unique solutions out of 22. A similar pattern was found for scenarios with other values of P2 and P3.

3.3. Arrangement examples

Figure 8 presents several solutions that have been generated by the implementation of subsequent scenarios. It can be noted that the contours are not closed, which results directly from the adopted method of discretization elements in the form of string codes (Fig. 1). There will often be some gap resulting from the difference between the length of the real/original contour and the contour using fixed lengths of component segments.

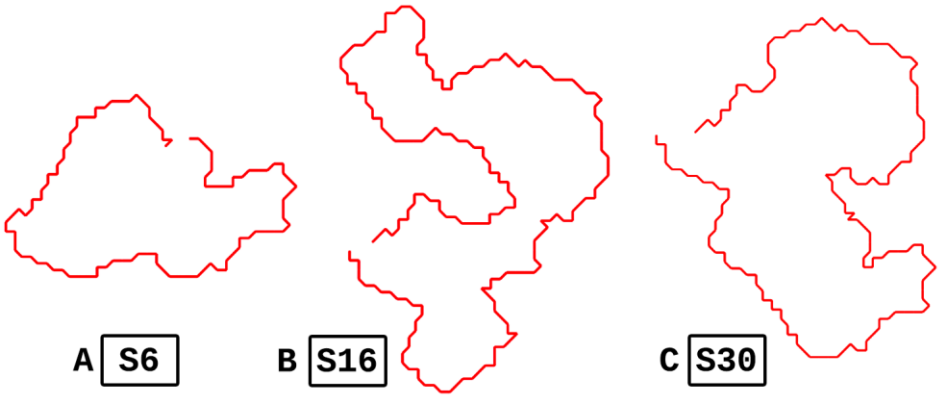


Fig. 8. Examples of solutions: A – one of the solutions for scenario S6, B – one of the solutions for scenario S16, C – one of the solutions for scenario S18.

Figure 8-A (from S6 – the longest to the shortest, P2 = 6, P3 = 0) represents the solution closest to the intended one. To a large extent, it resembles the source picture from which the elements for the experiment were prepared. Figure 8-B (from S16 – random order, P2 = 5, P3 = 1) presents an example that constitutes the majority of those that were determined in the course of all experiments. These arrangements are formally correct, but do not present compact figures. Figure 8-C (from S30 – the longest to the shortest, P2 = 9, P3 = 1) also presents an example of non-compact merge but thicker, giving the impression of being good connections.

It also turned out that the arrangements that were a solution in the scenarios with ML = 0 belonged to the set of components of solutions in the scenarios with ML<=1. This is a confirmation of the correctness of the method in the sense of sorting of the testable ways of elements arrangement. It turned out that for all solutions generated in the scenarios with ML=0, which did not use all 5 elements from the initial pool, the change of ML value to a higher one resulted in suggestions for new ways of joining.

4. Conclusions and future works

The results of the conducted numerical experiments lead to the following conclusions:

- the built algorithm is able to correctly find many alternative matching elements from the available pool,
- the selection of variables and their values have been correctly formulated, which is indicated by the significantly different values obtained: calculation time, number of the analysed assemblies, number of matches and amount of memory occupied,
- there is a clear trend that increasing the analysed character string simultaneously reduces the number of analysed assemblies, the number of obtained matches and the calculation time, which is important when performing calculations for a significantly larger number of elements from the initial pool.

The authors are convinced that further work should be performed to improve the existing version of the algorithm by building a mechanism that will automatically adjust the length of the analysed substring to the original number of characters describing the edges of the compared elements.

References

[1] E. Justino, L.S. Oliveira and C. Freitas, "Reconstructing shredded documents through feature matching", *Forensic Science International*, vol. 160, pp. 140–147, 2006

[2] T. Buschmann and L.V. Bystrykh, "Levenshtein error-correcting barcodes for multiplexed DNA sequencing", *BMC Bioinformatics*, vol. 14, pp. 272, 2013

[3] D.B. Cooper, A. Willis, S. Andrews, J. Baker, Y. Cao, D. Han, K. Kang, W. Kong, F.F. Leymarie, X. Orriols, S. Velipasalar, E.L. Vote, M.S. Joukowsky, B.B. Kimia, D.H. Laidlaw and D. Mumford. "Assembling virtual pots from 3D measurements of their fragments" in *Proceedings of the 2001 conference on Virtual reality, archeology, and cultural heritage (VAST '01),* Association for Computing Machinery, USA, pp. 241–254, 2001

[4] N.A. Rasheed and M.J. Nordin, "A Survey of Computer Methods in Reconstruction of 3D Archaeological Pottery Objects" in *International Journal of Advanced Research*, pp. 712–724, 2015

[5] M. Zhou, G. Geng, Z. Wu, X. Zheng, W. Shui, K. Lu, and Y. Gao, "A System for Re-assembly of fragment Objects and Computer Aided Restoration of Cultural Relics" in *Virtual Retrospect*, pp. 21-27, 2007

[6] W. Kong and B. Kimia, "On solving 2D and 3D puzzles using curve matching, Computer Vision and Pattern Recognition" in *Proceedings of the 2001 IEEE Computer Society Conference*, pp. 583–590, 2001

[7] H. C. da Gama Leitao and J. Stolfi, "A multiscale method for the reassembly of two-dimensional fragmented objects", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, pp. 1239–1251, 2002

[8] K. Guo, X. Chen, Y. Liu, B. Zhou and Y. Guo, “Geometric Based Structure Propagation and Texture Matching for 3D Texture Completion” in *International Conference on Virtual Reality and Visualization*, Xi'an, pp. 87–93, 2013
[9] A.R. Willis, “Computational Analysis of Archaeological Ceramic Vessels and their Fragments” in *Digital Imaging for Cultural Heritage Preservation*, pp. 323-352, 2011
[10] E.D. Demaine and M.L. Demaine, “Jigsaw Puzzles, Edge Matching, and Polyomino Packing: Connections and Complexity”, Graphs and Combinatorics vol. 23, pp. 195–208, 2007
[11] H. Freeman and L. Davis, “A Corner-Finding Algorithm for Chain-Coded Curves” in *IEEE Transactions on Computers*, pp. 297–303, 1977
[12] H. Freeman and L. Garder, “Apictorial Jigsaw Puzzles: The Computer Solution of a Problem in Pattern Recognition” in *IEEE Transactions on Electronic Computers*, pp. 118–127, 1964
[13] N.A. Aini, N. Dewi and A.S. Azurah, “Freeman chain code as representation in offline signature verification system” in Jurnal Teknologi, pp. 89–94, 2016
[14] C. Grabowik, K. Kalinowski, G. Ćwikła, A. Gwiazda and Z. Monica, “Zastosowanie kodów łańcuchowych do opisu konstrukcji oraz identyfikacji konstrukcyjnych obiektów elementarnych” in *Innowacje w zarządzaniu i inżynierii produkcji*, pp. 180–190, 2017
[15] P. Karczmarek, A. Kiersztyn, W. Pedrycz and M. Dolecki, “An application of chain code-based local descriptor and its extension to face recognition” in *Pattern Recognition*, pp. 26-34, 2017
[16] J. Montusiewicz, S. Skulimowski, “A Search Method for Reassembling the Elements of a Broken 2D Object”, Advances in Science and Technology Research Journal, vol. 14, pp. 49-56, 2020

Jolanta Słoniec[1], *Anna Kaczorowska*[2]

ON ETHICS IN ARTIFICIAL INTELLIGENCE

Keywords: ethics, artificial intelligence.

Abstract

The development of numerical methods and information technologies is occurring at a rapid pace. Hence, the main research question addressed by this work is whether ethics is required for artificial intelligence (AI), and thus (1) 'Are the existing documents regulating the ethics of AI sufficient?' and (2) 'Is there a need to develop new regulations? If so, in what areas?' The presented investigations involved the use of literature review method, case study, analysis, and synthesis.
Based on the analysis of scientific documents and articles, as well as examples related to the use of artificial intelligence, the main and detailed research questions have been answered in the affirmative. Nonetheless, there is a pressing need to develop documentation regulating: military AI use, AI applications in social networks, robot ethics, AI in the automotive industry, or ecology. The presented case studies indicate other unregulated areas: AI applications in medicine or legal provisions on specific issues. The lack of an AI code of ethics may hinder the development of new applications for intelligent devices in the future.

1. Artificial intelligence. Etymology, definition and historical outline

The term artificial intelligence (AI) was first introduced by John McCarthy in 1955, who defined it as the construction of machines whose operation resembles human intelligence.

AI practitioners have come from such disciplines as logic, mathematics, engineering, philosophy, psychology, linguist sticks, and computer science [1].

The concept of artificial intelligence is not explicit. The field of AI research was founded at a workshop held on the campus of Dartmouth College during the summer of 1956. [2] An attempt to define the concept of artificial intelligence was made by A. Turing, a British mathematician, cryptologist, and one of the creators of computer science. He is widely regarded as the father of AI.

Below are selected definitions of the term *artificial intelligence*.

- It is a field of knowledge covering fuzzy logic, evolutionary calculations, neural networks, artificial life and robotics. It is also an IT department dealing

[1] Lublin University of Technology, Department of Enterprise Organization, Faculty of Management, 20-618 Lublin, Nadbystrzycka St 38, Poland, j.sloniec@pollub.pl
[2] University of Lodz, Department of Computer Science, Faculty of Management, 90-237 Lodz, Matejki St 22/26, Poland, anna.kaczorowska@uni.lodz.pl

with intelligence – creating models of intelligent behaviour and computer programmes simulating these behaviours; [3]

- It is a process of automating certain activities such as decision-making and learning, which are a reflection of human intelligence; [4]
- It is a field of research seeking to explain and implement intelligent behaviour through the use of computational processes; [5]
- It is a science that covers the topics of algorithmics, fuzzy logic, evolutionary computations, neural networks, artificial life, and robotics. Artificial Intelligence is a branch of computer science whose subject is the study of rules governing intelligent human behaviour, creating formal models of these behaviours, and – as a result – computer programmes simulating these behaviours; [6]
- It is a cross-disciplinary approach to understanding, modelling, and replicating intelligence and cognitive processes by invoking various computational, mathematical, logical, mechanical, and even biological principles and devices; [1]
- These are systems that display intelligent behaviour; [7]
- Artificial intelligence (AI) refers to systems that display intelligent behaviour by analysing their environment and taking actions – with some degree of autonomy – to achieve specific goals. AI-based systems can be purely software-based, acting in the virtual world (e.g. voice assistants, image analysis software, search engines, speech and face recognition systems) or be embedded in hardware devices (e.g. advanced robots, autonomous cars, drones or Internet of Things applications). [8]

In Stanford Encyclopedia of Philosophy, the entry on Ethics of Artificial Intelligence and Robotics was first published on 3 April 2020. Seven aspects of AI use are distinguished: (1) issues of privacy, (2) manipulation, (3) opacity, (4) bias, (5) human-robot interaction, (6) employment, (7) effects of autonomy. 'Artificial Intelligence (AI) and robotics are digital technologies that will have a significant impact on the development of humanity in the near future' [9].

The beginnings of modern AI can be traced to classical philosophers' attempts to describe human thinking as a symbolic system. But the field of AI was not formally founded until 1956, at a conference at Dartmouth College, in Hanover, New Hampshire, where the term "artificial intelligence" was coined [10]. Among the participants, there was an MIT cognitive scientist, Marvin Minsky, who was then extremely enthusiastic about the future of the technology and have since been dubbed as the father of AI.

Initial works on artificial intelligence were developed and financed as part of US government projects. After the completion of the projects, there was a period of stagnation in the development of AI, and the resumption and intensification of work on AI occurred in the nineties of the twentieth century. A significant achievement in the development of AI was the Deep Blue computer, created in

1997 by IBM, which defeated Grandmaster Garry Kasparov at chess. The next significant step in the development of AI was the computer programme AlphaGo, which defeated the top-ranked human Go player, Lee Sedol, after winning four out of five games. The Chinese game of Go is much more complicated than chess [11]. Since then, works on artificial intelligence have gained momentum and are currently being continued in many countries.

2. Artificial intelligence and ethics

2.1. Outline of the state of AI ethics

Serious reflections and the debate on the ethics of machines began a few years ago when artificial intelligence reached a level and possibilities exceeding in many ways the level and capabilities of a thinking person. Earlier it seemed a distant future pertaining to science fiction (robot rights I. Asimov [12], D. Langford and M. Tilden [13]). The rapid development of IT technology has made the considerations about the ethics of machines of interest to scientists.

Tom Sorell from the University of Warwick, UK, in a summary of research results published in 2017 has argued that Asimov's three laws seem a natural response to the idea that robots will one day be commonplace and thus need internal programming to prevent them from harming people. Sorell worked on the EU-funded project that developed a robotic companion to help elderly people live independent lives. He concluded that the time was nigh to re-evaluate their effectiveness and prompt the discussion on a new set of laws that would accompany the ongoing, awe-inspiring breakthroughs in robotics and AI technology taking place in Europe and across the world [14].

At the beginning of the 20th century, scientific papers began to appear on 'friendly artificial intelligence' [15]. The central questions were: Will the creation of AI benefit the world? Will the AI take actions that are benevolent or malevolent, safe or uncaring, helpful or hostile? The author also analysed the ways in which AI and human psychology were likely to differ and the ways in which those differences were subject to design decisions.

The rapid development in IT in the 21st century caused that machine ethics has been receiving considerable scholarly attention. A significant number of scientific papers on AI ethics have been published in the last decade. One of the notable works in the field is *The Cambridge Handbook of Artificial Intelligence* [1]. Published in 2014, it provides a comprehensive discussion of AI-related issues, including the ethics of artificial intelligence.

2.2. Regulations regarding AI ethics

Various regulations regarding the AI code of ethics have been proposed to date. In the EU context, the most important document is the report *The ethics of*

artificial intelligence: Issues and initiatives, published in March 2020 [7]. This very recent report is a pioneering work in the field of legal regulations behind AI, whose subject had been to that point the concern of futurists rather than scientists or law-makers.

The publication is a comprehensive study organised along the following chapter-ideas: Mapping the main ethical dilemmas and moral questions associated with the deployment of AI; Ethical initiatives in the field of artificial intelligence; AI standards and regulation; National and International Strategies on AI; Emerging Themes.

Following the introductory chapter one, in the second chapter, after the presentation of the ethical, social and legal benefits resulting from the use of artificial intelligence, the impact of AI on the labour market, demography and human rights is discussed. Next, the impact of AI on human psychology is considered along with the impact of AI on fiscal systems of countries (considering stock turnover, intentional market manipulation). The attention is drawn to the use of AI for criminal activities (drone smuggling, theft and fraud). There is also an unresolved problem of liability for accidents involving autonomous vehicles (damages). Increasing implementation of AI technologies can also cause overuse of natural resources and excessive energy consumption; therefore, transparency, accountability, fairness and regulation of the process are highlighted.

Chapter three presents ethical initiatives in the field of AI, divided into the following areas: human rights and well-being; emotional harm; accountability and responsibility; security, privacy, accessibility and transparency; safety and trust; social harm and social justice; lawfulness and justice; control and the ethical use (or misuse) of AI; environmental harm and sustainability; informed use; existential risk.

It is emphasised that AI cannot affect basic human rights and should support the growth and development of prosperity. The problem of AI influence on human emotions is also identified (risk of creating intimate relationships between intelligent robots and people, manipulation of Internet users). The use of artificial intelligence is discussed as a factor that could increase the disparity between developed and developing countries. The use of AI should help preserve diversity in the plant and animal world.

For better illustration, the chapter contains case studies, including the participation of medical robots in the diagnosis, surgery and monitoring of patients. They also participate in the training of healthcare professionals. The potential risks that are highlighted are patient data protection and patient rights. In the second case study, ethical issues related to the use of autonomous vehicles are examined. Six levels of automation are distinguished: no automation, hands-on (e.g. Cruise Control), hands-off (driver still monitors driving), eyes-off (driver can turn attention elsewhere, but must be prepared to intervene), mind-off (no driver attention required) and steering wheel optional (human intervention is not required). There are no legal regulations related to the use of autonomous vehicles,

especially in the event of collision or accident. Issues of data protection for drivers and passengers are also recalled in connection with the collection of significant amounts of data by autonomous vehicles. The third case study explores the use of AI in warfare and the potential AI applicability as weapons, including lethal autonomous weapons, drone technologies, robotic assassination, and mobile-robotic-improvised explosive devices. In this case, ethical issues concern the possibility of minimising human decisions in the application of AI in warfare.

Chapter four discusses emerging AI ethics standards and regulations. The earliest of these is the *Guide to the Ethical Design and Application of Robots and Robotic Systems*. Ethical hazards were grouped under four categories: societal, application, commercial & financial, and environmental.

Chapter five describes the regulations regarding AI in various countries. The first countries that have developed documents on AI ethics were Canada and Japan. However, it was in the European Union that the first international document was developed.

Chapter six deals with new problems that need to be addressed. These are environmental issues, human psychology, workforce, particularly in relation to inequality and bias, democracy and finance.

Another international study prepared by the European Commission is *Ethics Guidelines for Trustworthy AI* [17], developed in 2019. According to the report from this study [18], trustworthy AI should have three characteristics: (1) it should be lawful, complying with all applicable laws and regulations, (2) it should be ethical, ensuring adherence to ethical principles and values, and, (3) it should be robust, both from a technical and social perspective.

AI systems should be developed and used according to the following rules: respect for human autonomy, prevention of harm, fairness and explicability. The attention is paid to certain social groups: children, disabled people and other groups at risk of exclusion. AI may be beneficial, but it comes with risks that are sometimes difficult to predict and identify.

Seven requirements have been formulated with respect to Trustworthy AI: (1) human agency and oversight, (2) technical robustness and safety, (3) privacy and data governance, (4) transparency, (5) diversity, non-discrimination and fairness, (6) environmental and societal well-being and (7) accountability.

Appropriate methods must be used to implement these requirements. Research and innovation should be supported so as to enable the assessment and ensure compliance with the requirements. An appropriate way should be provided to indicate that they are dealing with AI. AI should be easily identifiable and controlled. Educational activities should be supported to acquire knowledge of reliable artificial intelligence. Attention is drawn to the fact that conflicts may arise between rules and requirements, in which case, the conflicts should be identified, assessed and documented.

AI should be subject to constant control in accordance with the presented assessment list. The list is never complete and new requirements must be

constantly identified to further improve the functioning of AI, which may necessitate introducing a sectoral framework for the functioning of AI.

The document should be reviewed and updated over time (ensuring adequacy in relation to technological, social and knowledge changes) and encourage the development of an ethical framework for AI systems at a global level.

2.3. Characteristics of cross-sectional publication analysing AI ethics

The somewhat obvious consensus is that the functioning of AI systems should be subject to regulation. In his 2019 work [16], T. Hagendorff conducted a study on the degree of implementation of ethical principles and values in the practice of research on the development and application of AI. The article presents an analysis of 21 papers, grouped into 22 areas (in parentheses is the number of articles discussing a given research area): privacy protection (17); accountability(17); fairness, non-discrimination, justice (17); transparency, openness (15); safety, cybersecurity (15); common good, sustainability, well-being (15); human oversight, control, auditing (12); explainability, interpretability (10); solidarity, inclusion, social cohesion (10); science-policy link (10); legislative framework, legal status of AI systems (9); responsible/intensified research funding (8); public awareness, education about AI and its risks (8); future of employment (8); dual-use problem, military, AI arms race (7); field-specific deliberations (health, military, mobility etc.) (7); human autonomy (7); diversity in the field of AI (6); certification for AI products (4); cultural differences in the ethically aligned design of AI systems (2); protection of whistle-blowers (2); hidden costs (labelling, clickwork, contend moderation, energy, resources) (1). The author points out that while working on AI systems, ethical issues are often treated as additional, insignificant requirements, or as a marketing strategy; in addition, he specifies a list of risk areas that have been largely neglected: the danger of a malevolent AI, machine consciousness, the reduction of social cohesion by AI ranking and filtering systems on social networking sites, the political abuse of AI systems, a lack of diversity in the AI community, links to robot ethics, dealing with trolley problems, the weighting between algorithmic or human decision routines, “hidden” social and ecological costs of AI, to the problem of public-private-partnerships and industry-funded research.

According to the author, the turn to AI-powered technologies brings the following consequences: closing the gap between ethics and technical discourses and when constructing systems in AI, focusing more on ethical rather than technical problems and balancing actions between the two approaches.

2.4. Case study of problems using AI

2.4.1. AI in the financial sector [19]

The financial sector employs this new technology in the areas of analytics, algorithmic trading, credit score analysis, fraudulent transaction monitoring, customer interaction (chatbot). The use of artificial intelligence in this sector encompasses a range of areas that are not yet regulated by the law. The first problem is whether AI can be granted legal personality at all. This entails consequences – allowing algorithms to conclude contracts (civil and criminal liability).

Currently, AI in the financial sector is used to assess creditworthiness, monitor payment transactions, support investment analysis and decisions, in customer identification and authentication, in chatbots for external and internal communication, in risk analysis and risk management, and customer profiling.

To determine customers' creditworthiness, advanced algorithms are used that make credit decisions based on data obtained from various sources. However, the customer does not have access to the algorithm itself or to its detailed description, which raises reservations.

AI participates in trading investment products by making automated trading decisions (algorithms turn over clients' funds and must be supervised). Regulations require responsibility, monitoring and the use of algorithms.

2.4.2. AI and copyright [20]

Traditionally understood copyright protects only human artistic objects. Meanwhile, the development of intelligent systems has led to the possibility of creating artistic objects by autonomous robots. Therefore, it seems appropriate to modify the law in this respect.

2.4.3. AI supporting doctors [21]

AI is growing to become the fastest and the most reliable solution in medical diagnostics. An example is an automatic system supporting diagnostics in cardiology (efficiency 97% detection of cardiac arrhythmia).

There is a problem with the procedure for placing systems on the market. The system has the ability to learn and, therefore, can autonomously modify the algorithms. The software approval rating can easily become outdated, which is why the AI must be monitored and evaluated periodically.

2.4.4. AI and the protection of image and correspondence [22]

The problems related to human presence in cyberspace after death are subject to protection under the copyright law. The law provides for the image and correspondence when the subject functions in two worlds (real and digital) and

when biological death does not end the existence of the individual in the digital world. The issue of human digital goods after their death has not yet been regulated. Two notable examples of post-mortal Facebook profiles concern a young man and a girl. In the former case, after their son's death in an accident, the family faced the dilemma of what to do with his profile, which involved the issue of access to digital resources left by the deceased on the network. In the latter, the girl's parents turned to Facebook for access to the profile after their daughter's subway accident. The case ended up in a court's ruling that digital content was inherited similarly to letters or diaries.

There is also a problem with creating *deepfake* materials that use AI to create realistic video and audio recordings of real people doing unreal things.

2.4.5. AI in HR [23]

Artificial intelligence in HR is primarily employed in the recruitment, pre-selection, internal communication and employee service processes. The recruitment process uses, for instance, the Textio tool, which allows you to create a recruitment advertisement that would be attractive to candidates and effectively encourage applicants. The pre-selection process uses chatbots and virtual recruiters to improve the acquisition of applications.

AI can also be found in internal communication and personnel administration applications. It streamlines the process of introducing a new employee to the company (the use of an intelligent assistant with a built-in knowledge base). It also enables more efficient handling of employee queries (a significant number of employee queries are repetitive).

However, there is a problem with the possible dismissal of employees in HR departments in connection to the use of modern technology. The protection of sensitive personal data in systems using AI also requires detailed regulations.

2.5. Research questions, research methods, reference to research questions

The main research question is whether ethics is needed in artificial intelligence. Detailed research questions are: (1) Are the existing documents regulating the ethics of artificial intelligence sufficient? (2) Is there a need to develop new regulations? If so, in what areas?

In order to respond to such questions, the following research methods were used:

- literature study – learning and analysing the content of existing documents regulating the ethics of artificial intelligence on an international scale;
- case study – analysis of specific cases of regulation of the functioning of devices equipped with artificial intelligence, with particular emphasis on ethical problems;

- analysis and synthesis.

The regulations regarding the AI code of ethics were presented in the previous chapter, whereas the context of problems arising from the use of AI techniques was provided by the study cases.

After analysing the documents governing the operation of AI and selected case studies, it must be remarked that AI ethics is not only needed but it is absolutely essential. It is shown that at the current stage of development, systems and devices equipped with artificial intelligence must be constructed, developed and operated in compliance with ethical regulations. If it is necessary to apply ethical principles at the present stage of artificial intelligence development, it will be all the more necessary in the future when AI technologies find new applications.

With respect to the first question, existing documents regulating matters of AI ethics are sufficient for the current development of the technology. These are extensive regulations prepared by specialists from various fields (psychology, sociology, law, economics, management, engineering sciences, linguistics, computer science), covering a wide spectrum of issues. It should be noted that these documents include the need to update regulations as technology evolves and changes in societies. The documents prepared by the European Commission and European Union experts are primarily guidelines and initiatives. They describe the current ways of using AI, pointing to the emerging problems in this regard. They contain normative principles and recommendations as well as set out the requirements that Trustworthy AI should meet. However, they do not regulate all problems related to the functioning of AI, especially in the context of legal mechanisms.

Considering the second query, it should be stated that after analysing the case study, there is a need to develop legal regulations related to the functioning of AI in specific areas. Artificial intelligence covers and touches upon an ever-widening range of areas pertaining to our lives. In many cases, it is the law that requires a major update, in order to adapt it to new AI applications or create new documents regulating selected areas. Regarding the areas in need of new regulations, the following specific issues emerge: military AI use, applications neither in social networks, robot ethics, automotive, ecology. The case studies point to other unregulated areas: applications in medicine, legal provisions on specific issues (copyright, AI creation, image protection and correspondence, HR).

The range of applications is constantly expanding because the development of science and technology is opening up new horizons to AI. What would seem impossible until recently is now becoming real. The new regulations are to ensure that the use of AI will not only benefit but also protect humanity.

3. Reflections on ethical issues of artificial intelligence

The turn of the twenty-first century is a period of rapid technological progress in IT. Since then, intelligent machines have appeared not only in the workplace

(industrial robots) but also in public space (public transport, schools, hospitals, smart city) and in our homes (smart home). As a society, we should be aware of the ethical dilemmas of machines that take over a large part of our responsibilities, not only related to our work, but to our decisions as well. Machines not only perform heavy, mundane or dangerous works (in mines, factories, rescue) and computations, but also carry out tasks that previously were only human responsibility. Machines are becoming more intelligent, which makes interaction with them more complex.

Therefore, in the face of the rapid development of AI, it is the scientists' responsibility to create and improve the principles of machine ethics to observe in human-machine interactions. Uncontrolled technology development is a matter of mounting concern as the majority of the society declares techno-realistic or even techno-pessimistic attitudes, outnumbering the techno-optimists. Workers fear being replaced by machines, losing employment or being dominated. At the same time, it is a case of double moral standards when the most avid critics of new technologies simultaneously chase the latest models of a computer or a smartphone. Other concerns include technological determinism, i.e. whether technology can develop itself without the control of human beings. Currently, technological progress seems unstoppable. Any dilemmas may thus only concern the direction in which it will occur.

AI has already become capable of improving itself and machines are believed to soon become more intelligent than people. At the current stage of development, machines are known to make more accurate medical diagnoses or issue fairer court verdicts (based on prior databases and expert knowledge). Techno-pessimists anticipate that machines will deprive humans of work, and, secondly, that the development of AI technologies can pose a threat to the existence of our species. In their view, at best, in a world where machines take control, we could simply be neglected or people's position would be at risk.

Looking from a different angle, in a sense, machines are already dominated. Machines are controlled by machines and in turn, these are again controlled by other parent machines. This is how communication takes place when decisions made by people are kept to the minimum.

In fact, it is a two-way control. A modern aeroplane is controlled by a machine – an autopilot, in turn, is controlled by a human – a pilot. When applying for a mortgage we are stood against not only a bank director who considers granting us the loan but against an algorithm – a machine that can ultimately prevent the director – a man from approving our application. Man supervises the machine, but at the same time, the machine supervises the man, for instance, some corporate supervisory boards use algorithms with decision power. However, one should not forget that the algorithms are programmed by people and that they ultimately decide about their possibilities and limitations.

3. Summary

Reflections on the AI ethics were presented, including the documentation regulating the AI ethics on an international scale, selected case studies of unregulated issues and cross-sectional publications in the field of AI technologies. The study sought to answer the central research question (Is ethics required in AI?) and two specific research questions ((1) Are the existing documents regulating the AI ethics sufficient? (2) Is there a need to develop new regulations? If so, in what areas?) The research methods employed in the work were literature analysis, case studies, analysis and synthesis.

It was confirmed that ethics is not only necessary but must be considered as an absolutely indispensable element of AI technologies. Furthermore, it was found that the existing documents regulating AI ethics, developed by the European Commission, are sufficient. Especially given that the documents in question will be regularly updated in order to keep abreast of the developments in technology and society. Finally, from a close analysis of the cross-sectional publication, it emerges that there are numerous unregulated issues in the following spheres of AI application: military, social networks, robot ethics, automotive industry, ecology. On the other hand, the presented case studies point to other areas in need of regulatory changes: applications in medicine, legal provisions on specific issues (copyright, AI creation, image protection and correspondence, HR).

Ming Zeng [25, 26], a spectacularly successful business practitioner, has remarked that the commercialisation of cloud computing and AI technologies has made large-scale computational power and analytic capabilities accessible to anyone. The cost of storing and computing large quantities of data has dropped dramatically over the past decade. This means that real-time applications of machine learning are now applicable and affordable in more and more environments. The rapid development of internet of things technology will further digitise our physical surroundings, providing ever more data. As these innovations are set to accumulate in the coming decades, the winners will be companies that become *smart* faster than the competition.

In his book, *Deep Learning Revolution*, one of the creators of deep learning, T. Sejnowski [24] explains how deep learning has changed from a secret field of science into a breakthrough technology in information economy. He states that one day a car without a driver will know the road better than the owner of the car and will drive the car better and safer; the deep learning network will diagnose the patient's disease faster and more accurately; a personal cognitive assistant will increase the capabilities of the human brain. The evolution of human intelligence has been in progress for millions of years whereas AI has been developed over decades.

The society demands that artificial intelligence be perfect and make only good, right decisions, and, on the other hand, a human representative of the society can be imperfect and make wrong decisions. The demand for the perfection of

artificial intelligence is not bad, because in this way we strive to make artificial intelligence the best. This means that the level of requirements for new technologies is very high.

The society is becoming more complex, has more sophisticated structures, and more interrelated layers. Therefore, large areas of society management are delegated to AI.

Considering the labour market, there is a problem of job loss as a result of introducing the work of machines in place of human operators. At the same time, machines are being implemented in the decision-making process, therefore, the way machines make decisions requires urgent regulation. Machines perform work not only requiring mechanical strength or computational power, but also related to making decisions in machine operation involving ethics, i.e. the concept of good and evil – morality. An autonomous car which makes on the spot decisions during movement should have a built-in code of ethics, i.e. have such a code implemented in the decision-making unit.

People will impose ethical codes on machines, but in the future, it is also possible that machines will adopt them themselves. If machines are capable of self-improvement, they will be able to make decisions and change the original programme. Hence, answering the question posed in the title, it should be stated that yes, machines at the current stage of development require ethics. In fact, the need is bound to grow in the future as they will be expected to strive for absolute perfection.

The conclusion that emerges from this study corroborates earlier considerations reflected in the documents and other works on the ethical conditions for the use of artificial intelligence should be accelerated because new problems related to human interaction with a device equipped with AI will come to the surface. The lack of the 'code of ethics' for artificial intelligence may hinder the development of new applications for intelligent devices in the future.

References

[1] K. Frankish, W.M. Ramsey, *The Cambridge Handbook of Artificial Intelligence*, Cambridge University Press, 2014. Available: https://books.google.pl/books?hl=pl&lr=&id=RYOYAwAAQBAJ&oi=fnd&pg=PR8&dq=The+ethics+of+artificial+intelligence&ots=A000ujdExu&sig=A1GkHlc6mYz1McMFyFi9GsMB8nI&redir_esc=y#v=onepage&q=The%20ethics%20of%20artificial%20intelligence&f=false

[2] A. Kaplan, M. Haenlein, Siri, Siri, in my hand: Who's the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence. *Business Horizons*, 62: 15–25, 2019. doi: 10.1016/j.bushor.2018.08.004

[3] A. Turing, *Computing Machinery and Intelligence*, Mind, vol. LIX, no. 236, 1950, pp. 433-460. Available: http://web.archive.org/web/20110726153108/http://orium.homelinux.org/paper/turingai.pdf

[4] R. Bellman, *An Introduction to Artificial Inteligence*, Boyd & Fraser Publishing Company, San Francisco, p. 2, 1978
[5] R. J. Schalkoff, *Artificial intelligence: an engineering approach*, McGraw-Hill Education, p. 35, 1990
[6] P. Thagard, *Computational Philosophy of Science*, The MIT Press, p. 11, 1993
[7] *The ethics of artificial intelligence: Issues and initiatives*, European Union, Brussels, 2020. Available: https://www.europarl.europa.eu/RegData/etudes/STUD/2020/634452/EPRS_STU(2020)634452_EN.pdf
[8] *High-level Expert Group on Artificial Intelligence*, A Definition of AI Main Capabilities and Disciplines, 2019. Available: file:///Users/jolantasloniec/Downloads/ADEFINITIONOFAI-ENpdf.pdf
[9] *Stanford Encyclopedia of Philosophy*, Available: https://plato.stanford.edu/entries/ethics-ai/
[10] T. Lewis, *A Brief History of Artificial Intelligence, Live Science*, 2014. Available: https://www.livescience.com/49007-history-of-artificial-intelligence.html
[11] C. Brezina, *Artificial intelligence and you*, The Rosen Publishing Group, New York, 2020
[12] I. Asimov, *Runaround*, Doubleday, New York, 1942
[13] T. Marsh, *The Evolution of a Roboticist*, 2020. Available: http://www.botmag.com/the-evolution-of-a-roboticist-mark-tilden/
[14] *Why Asimov's Laws of Robotics should be updated for the 21st century,* 2017 Available: https://cordis.europa.eu/article/id/121860-why-asimovs-laws-of-robotics-should-be-updated-for-the-21st-century
[15] E. Yudkowsky, *Creating Friendly AI 1.0*, The Singularity Institute, San Francisco, 2001. Available: http://intelligence.org/files/CFAI.pdf
[16] T. Hagendorff, *The Ethics of AI Ethics an Evaluation of Guidelines*, 2019. Available: https://arxiv.org/pdf/1903.03425.pdf,
[17] *Ethics guidelines for trustworthy AI*, 2019. Available: file:///Users/jolantasloniec/Downloads/EthicsguidelinesfortrustworthyAI-PLpdf.pdf.
[18] *High-level Expert Group on Artificial Intelligence*, Ethics Guidelines for Trustworthy AI, 2019. Available: file:///Users/jolantasloniec/Downloads/AIHLEG_EthicsGuidelinesforTrustworthyAI-ENpdf.pdf
[19] M. Nowakowski, Na styku regulacji, czyli sztuczna inteligencja w sektorze finansowym – status quo i kierunki rozwoju prawa, 2020. Available: https://sip.lex.pl/#/publication/470123871/nowakowski-michal-na-styku-regulacji-czyli-sztuczna-inteligencja-w-sektorze-finansowym-status-quo...?keyword=Sztuczna%20inteligencja%20przyk%C5%82ady%20problemy&cm=SFIRST
[20] P. P. Juściński, Prawo autorskie w obliczu rozwoju sztucznej inteligencji, *ZNUJ* 2019/1/5-44, 2019. Available: https://sip.lex.pl/#/publication/151348313/juscinski-przemyslaw-piotr-prawo-autorskie-w-obliczu-rozwoju-sztucznej-inteligencji?keyword=Sztuczna%20inteligencja%20przyk%C5%82ady%20problemy&cm=STOP

[21] A. Pochrzęst-Motyczyńska, Sztuczna inteligencja to narzędzie wspomagające, a nie zastępujące lekarzy, 2020. Available: https://sip.lex.pl/#/externalnews/1795659299?keyword=Sztuczna%20inteligencja%20przyk%C5%82ady%20problemy&cm=STOP

[22] A. Nowak-Gruca, Użytkownik forever off. Wybrane problemy cyfrowego „życia po życiu" związane z ochroną wizerunku i korespondencji, 2020. Available: https://sip.lex.pl/#/publication/151361732/nowak-gruca-aleksandra-uzytkownik-forever-off-wybrane-problemy-cyfrowego-zycia-po-zyciu-zwiazane...?keyword=Sztuczna%20inteligencja%20przyk%C5%82ady%20problemy&cm=SREST

[23] J. Krajewski, Boty w HR - czy i jak dział HR może wykorzystać nowe technologie?, 2019. Available: https://sip.lex.pl/#/publication/470118751/krajewski-jacek-boty-w-hr-czy-i-jak-dzial-hr-moze-wykorzystac-nowe-technologie?keyword=Sztuczna%20inteligencja%20przyk%C5%82ady%20problemy&cm=SREST

[24] T. J. Sejnowski, *The Deep Learning Revolution*, Massachusetts Institute of Technology, USA, 2018

[25] M. Zeng, *Smart business. What Alibaba's success reveals about the future of strategy*, Publisher Harvard Business Review Press, 2018

[26] M. Zeng, Alibaba and the Future of Business, *Harvard Business Review*, 2018. Available: https://hbr.org/2018/09/alibaba-and-the-future-of-business

Monika Kaczorowska[1], Mikhail Tokovarov[2]

USER EXPERIENCE DRIVEN ASSESSMENT OF SELECTED FEATURES AND PERSPECTIVES OF INSTANT MESSAGING APPLICATIONS

Keywords: user experience, instant messaging, communication.

Abstract

The main subject of this research paper is analysing the current state of the mobile instant messaging application market and defining the possible perspectives of its evolution. The present research is addressed to architects and designers of instant messaging application. A review of the literature shows that researchers endeavour to evaluate the usability of instant messaging, but they focus on the existing features of this type of applications. The authors conducted a survey covering the features of instant messaging applications. Respondents from several countries and of various age evaluated selected functions of instant messaging applications and proposed theirs as well. The article provides the following benefits for the reader: users' opinions, promising features that can be added to applications and information about the features that users consider to be redundant. The information is going to be helpful for mobile instant messaging application designers and architects looking for new ideas. Another achievement is discovering the user's needs which are not obvious for architects and designers, thus providing them with new ideas and directions of research.

1. Introduction

Communication is one of the essential parts of modern society. People use communication applications in almost every aspect of their lives. We chitchat with our friends and relatives, study and work remotely, check the news and even visit a doctor via mobile instant messaging apps. Everyone can experience it even more intensively in the days of the COVID-19 pandemic, which has led to the appearance of completely new products, barely known before. In view of these considerations, (the authors believe that) today's situation could be used for studying the existing facilities in the field of communications as well as searching for and testing new experimental solutions in this regard.

Jung [1] wrote that social networking sites are a part of daily social communications. The author researched the level of both younger and older adults' satisfaction with Facebook. Age makes a difference in Facebook activities and expectations towards such applications. Quan-Haase and Young [2] showed how the various kinds of instant messaging applications met users' needs. The authors attempted to explain why people integrated several media into their

[1] Lublin University of Technology, Department of Computer Science, Nadbystrzycka 36B, 20-618 Lublin,, Poland, m.kaczorowska@pollub.pl

[2] Lublin University of Technology, Department of Computer Science, Nadbystrzycka 36B, 20-618 Lublin, Poland, m.tokovarov@pollub.pl

communication habits. Vazquez et al. [3] focused in their research on customer satisfaction and loyalty to instant messaging in China. Their results show that trust, service quality, customer value, including functional and emotional value contribute to customer satisfaction. Age, sex and the time of using applications have an influence on rating the usability of and loyalty to facilities. Caro-Alvaro et al. [4] point out that the popularity of instant messaging is growing all the time and the evaluation of the usability of such applications allows for providing more satisfying user experiences. The authors describe the weaknesses of instant messaging applications. Darin et al. [5] present an application they created to improve student-faculty communication at a Brazilian university. The authors evaluated the usability and user experience of their implementation. Tang and Hew [6] researched the usefulness of the mobile instant messaging application weChat which could be applied as a tool to support teaching and learning. The results show that weChat can be applied to transmissive, demonstrative, dialogic, journaling, helpline and notebook activities or promoting affective social presence. In [7], Kaya et al. compared four popular mobile applications, i.e. WhatsApp, Facebook, Youtube and Mail. They applied the System Usability Scale (SUS) with an adjective rating scale to evaluate the measure of usability. The results showed that WhatsApp had higher usability than Facebook. Sutikno et al. [8] compared three instant messaging applications: WhatsApp, Viber and Telegram in terms of free instant messaging services and encryption.

Piwek and Joinson [9] wrote about the ease of use of the Snapchat application. Popular among teenagers, Snapchat is mainly used to communicate or share selfies.. Junco and Cotton researched the influence of instant messaging on the education process of students. Over half of the students claimed that instant messaging had a negative influence on their study and they could not be focused on multi-tasks [10]. However, in [11], the authors conducted research into the impact on nursing student's clinical learning experiences. The results showed that students often used WhatsApp and highly rated the application which allowed them to communicate with others and reduced their feeling of isolation. Similar research concerning dental students was conducted by Khatoon et al. [12]. Instant messaging applications are used not only to communicate with friends, but they are applied in healthcare, sales, tourism and other branches of business. The patient can consult via instant messaging by using text, photos or audio messages. WhatsApp is researched in [13]. Instant messaging is becoming an essential part of our life. People use instant messaging at work where it has both positive and negative impacts on workers [14-15].

The authors pose the following research questions:

- Have the most popular mobile instant messaging applications reached the limit of development and would adding any new features be redundant?
- Are users satisfied with modern communication applications?

- Do users have suggestions in terms of prospective features that the applications could have?

The aim of the article is analysing the current state of the market of mobile instant messaging applications and defining the possible perspectives of their evolution. The present research is addressed to instant messaging application architects and designers. The article provides the following benefits for the reader:

- opinions of users,
- promising features that can be added to the applications,
- information about the features that the user considers to be redundant.

2. Research methods

The main research method applied here is a survey. The survey has the following structure: its first section contains the data allowing of splitting respondents into several target groups, e.g. by age, sex or education. Sections 2 and 3 examine the most popular mobile instant messaging applications and define the most popular application and the distribution of popularity among them. Section 4 discusses particular assets of the applications according to the respondents' opinions on their usefulness. The section presents the features which both already exist and the author considers to be promising. The existing features were not intentionally separated from potential ones as the author wanted to obtain the usefulness estimation not disturbed by the fact that some features existed and some did not. The existing features are presented in the first part of the list (positions a.-m.). The rest of the list are the features that, to the best knowledge of the author, are not present in any major mobile instant messaging application. Section 5 includes a text box where the respondent can propose features not listed in the previous section but which would be useful. The last section of the survey contains a text box where the respondent can suggest the features which are not listed in Section 4 and considered useless. The survey is presented in section 2.1.

2.1. Survey

The survey was anonymous and took 5 minutes to complete. The questions from section 1–4 were mandatory, while the questions from sections 5–6 were optional. The following survey was applied in the research.

1. Submit the following data about yourself
    a. Sex (Male/Female)
    b. Age (11–20 years, 21–30 years, 31–40 years, etc.)
    c. Education (basic education, secondary education, higher education, no education)
    d. Nationality

2. Choose the mobile instant messaging applications you know: Facebook Messenger, VK Messenger, Telegram, WhatsApp, Instagram, Skype, Google Hangouts, Viber, Zoom

3. Choose the mobile instant messaging application you use. If the list does not contain the ones you use, please add them: Facebook Messenger, VK Messenger, Telegram, WhatsApp, Instagram, Skype, Google Hangouts, Viber, Zoom. You can add another option.

4. Assess the usability of the listed features: the left is completely useless and the right is extremely useful.
    a. Sending text messages
    b. Sending voice messages
    c. Sending video messages
    d. Sending pictures
    e. Sending documents, e.g. docx, pdf, xlsx, txt
    f. Sending reactions
    g. Sending GPS coordinates
    h. Sending telephone contacts
    i. Audio calls
    j. Video calls
    k. Conference calls
    l. Autosaving of sent pictures
    m. Special effects such as filters and masks
    n. Mini-games and multi-user games
    o. Changing voice sound during audio calls, e.g. from your normal voice to a child's voice
    p. Sharing the device screen
    q. Direct YouTube streaming
    r. Built-in translator
    s. Storage of sent files in an organised way, e.g. in folders
    t. Adding shared shopping list
    u. Adding shared plan check list
    v. Text recognition from photos, e.g. a user can take a photo of a document and its content is recognised and transformed into an editable text

5. Please write features which were not mentioned in the above list and you consider to be useful

6. Please write features which were not mentioned in the above list and you consider to be useless

2.2. Data

The survey was completed by 255 participants from the academic environment which includes possibly the most active users of mobile instant messaging applications: students and academic teachers. The respondents of various ages took part in the survey: both full-time and extramural students as well as teaching and research staff. More than 95% of the survey participants were students of a first, second or third cycle of studies. The survey was completed by Poles and foreigners.

A possible limitation is the fact that the group covered by the survey is not going to be completely representative as some potential targets might be left out. An example could be elderly people who gradually begin to discover the world of mobile instant messaging applications. Unfortunately, they cannot be reached as easily as students who can be gathered in one room, e.g. during a lecture.

3. Results

The analysis was conducted and the results obtained are presented below. Figure 1 presents the bar plots with the percentage of familiarity and use of specific instant messaging applications by different genders.

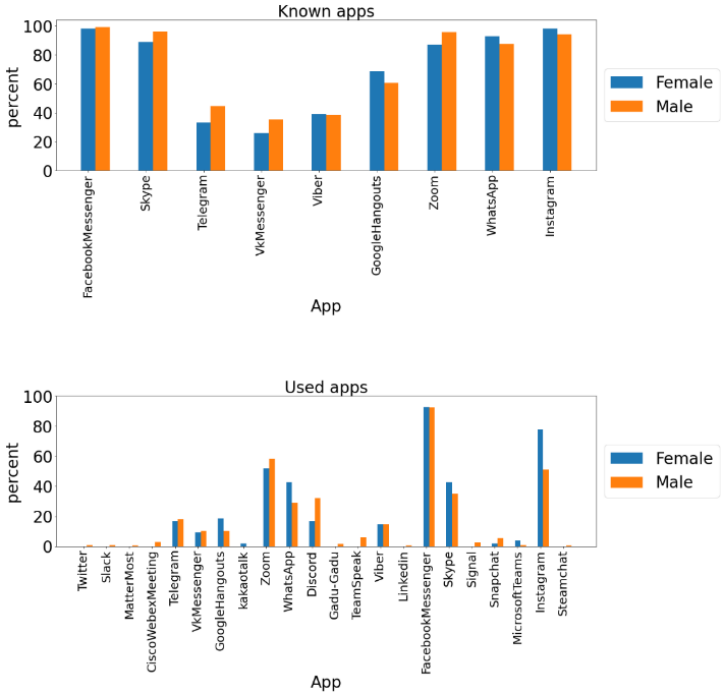


Fig. 1. The percentage of familiarity and use of specific instant messaging applications by different genders.

It can be noticed that there are no significant differences between men's and women's knowledge about the main instant messaging applications, but in terms of use, it can be observed that some applications such as Skype and, especially, Instagram present higher interest for women.

Figure 2 presents the boxplots of the rates given by the participants to the main features of instant messaging applications. The outliers were replaced with the mean values of corresponding features to ensure an unbiased estimation of the main statistical features. Quite expectedly, the ability to send text messages obtained the highest rates, which is clear from the median value of that feature equal to 10. The ability to send pictures and documents (e-d) also had the median value equal to 10, which is quite natural as the vast majority of respondents are students. The following features: sending voice messages (b), sending video messages (c), sending reactions (f), sending GPS coordinates (g), audio calls (i), video calls (j), conference calls (k), sharing the device screen (p), built-in translator (r), storage of sent files in an organised way (s), text recognition from photos (v) obtained high rates with the median value falling into the range between 7 and 9.

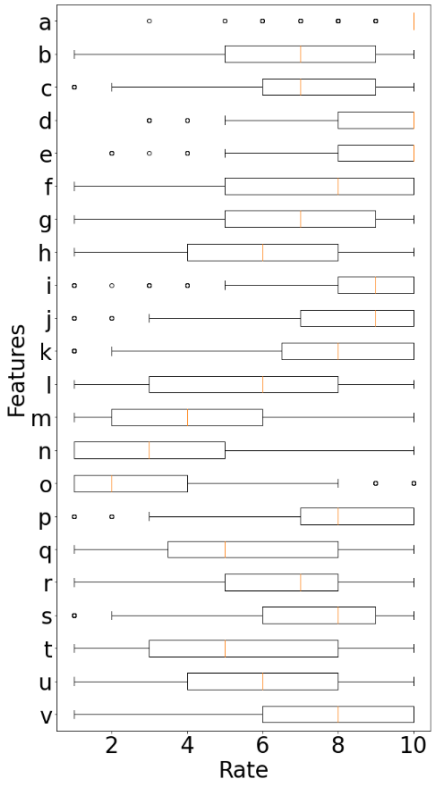


Fig. 2. The box-plots containing the main statistical indexes of the rate of use of the features of instant messaging applications.

The features listed below obtained the moderate rates from 5 to 7: sending telephone contacts (h), autosaving of sent pictures (l), direct YouTube streaming (q), adding a shared shopping list (t) and adding a shared plan check list (u). The following features were rated as less useful: special effects such as filters and masks (m), mini-games and multi-player games (n), changing the voice sound during audio calls (o).

The correlation analysis reveals the fact that some features were rated similarly to the others. Figure 3 contains the illustrations of the correlation matrix. The elements marked with dots have the p value less than 0.05. It can be observed that the following pairs of instant messaging applications demonstrate a high degree of correlation: adding a shared plan check list (u) and storage of sent files in an organised way (s), storage of sent files in an organised way (s) and adding a shared shopping list (t), built-in translator (r) and adding a shared plan check list (u), built-in translator (r) and adding a shared shopping list (t), built-in translator (r) and storage of sent files in an organised way (s), video calls (j) and conference calls (k), audio calls (i) and video calls (j), sending GPS coordinates (g) and sending telephone contacts (h), sending pictures (d) and sending documents (e), sending voice messages (b) and sending video messages (c).

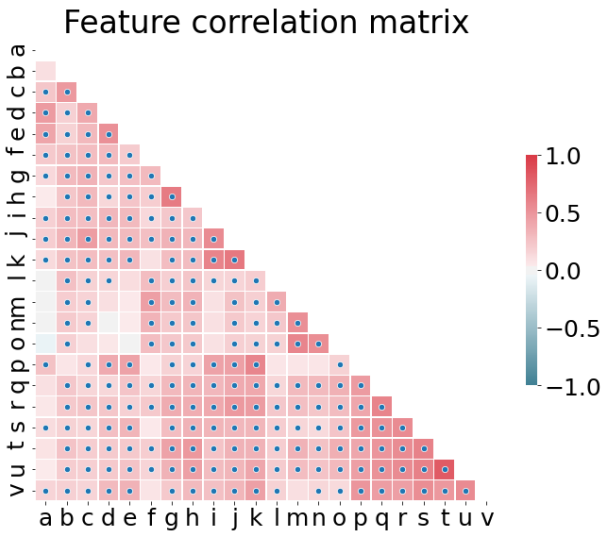


Fig. 3. Feature correlation matrix.

The participants also came up with the following features which could be potentially profitable according to them: storing data in a cloud rather than in the user's device, creating secrete conversations, live painting allowing them to create a virtual whiteboard, end-to-end encoding, shared calendar, voice typing, implementing and using chat bots, sending shots of a part of the screen, music recognition, payments via PayPal, etc.

4. Conclusions

As the main contribution, the degree of users' interest in specific features of mobile instant messaging applications was received. The information is going to be helpful for mobile instant messaging application designers and architects, looking for new ideas. Students can also gain profits out of the results of the research as they will have topics for their diploma theses. The potentially profitable features were not intentionally separated from the existing features: the author wanted the respondents to treat them as if they already existed. Emphasising that they do not exist could cause the survey participants to consider them as something useless. Another achievement would be discovering such users' needs which are not obvious for architects and designers, thus providing them with new ideas and directions of research. A higher number of participants could further increase the probability of their valuable suggestions. A review of the literature shows that the authors conduct research to evaluate the usability of instant messaging but focus on the features the tested application already has. User experience is very important in such applications because end-users of these applications are both young and older people. The proposed research focuses on the perspectives of the features of instant messaging which can be developed or removed. The correlation analysis can help to discover possible interactions between the features, showing which features are rated similarly, which can potentially be evidence of some deeper relationship between them.

References

[1] Jung, E. H. Age differences in using Facebook: younger adults vs. older adults. In Proceedings of the 30th Australian Conference on Computer-Human Interaction, pp. 590–592, 2018

[2] Quan-Haase, A., & Young, A. L. Uses and gratifications of social media: A comparison of Facebook and instant messaging. Bulletin of Science, Technology & Society, 30(5), 350–361, 2010

[3] Vazquez, D., Dennis, C., & Zhang, Y. Understanding the effect of smart retail brand - Consumer communications via mobile instant messaging (MIM) - An empirical study in the Chinese context. Computers in Human Behavior, 77, 425–436, 2017

[4] Caro-Alvaro, S., Garcia-Cabot, A., Garcia-Lopez, E., De-Marcos, L., & Gutierrez-Martinez, J. M. Mobile instant messaging apps: usability evaluation on iOS and android platforms and recommendations for developers, 2017

[5] Darin, T., Andrade, R., Macedo, J., Araújo, D., Mesquita, L., & Sánchez, J. Usability and UX Evaluation of a Mobile Social Application to Increase Students-Faculty Interactions. In International Conference on Human-Computer Interaction. Springer, Cham, pp. 21–29, 2016

[6] Tang, Y., & Hew, K. F. Examining the utility and usability of mobile instant messaging in a graduate-level course: A usefulness theoretical perspective. Australasian Journal of Educational Technology, 35(4), 2019

[7] Kaya, A., Ozturk, R., & Gumussoy, C. A. Usability measurement of mobile applications with system usability scale (SUS). In Industrial Engineering in the Big Data Era, Springer, Cham, pp. 389–400, 2019

[8] Sutikno, T., Handayani, L., Stiawan, D., Riyadi, M. A., & Subroto, I. M. I. WhatsApp, viber and telegram: Which is the best for instant messaging?. International Journal of Electrical & Computer Engineering, 6(3), pp. 2088–8708, 2016
[9] Piwek, L., & Joinson, A. "What do they snapchat about?" Patterns of use in time-limited instant messaging service. Computers in human behavior, 54, pp. 358–367, 2016
[10] Leskovec, J., & Horvitz, E. Planetary-scale views on a large instant-messaging network. In Proceedings of the 17th international conference on World Wide Web, pp. 915–924, 2008
[11] Pimmer, C., Brühlmann, F., Odetola, T. D., Dipeolu, O., Gröhbiel, U., & Ajuwon, A. J. Instant messaging and nursing students' clinical learning experience. Nurse education today, 64, pp. 119–124, 2018
[12] Khatoon, B., Hill, K. B., & Walmsley, A. D. Instant Messaging in Dental Education. Journal of Dental Education, 79(12), pp. 1471–1478, 2015
[13] Tazegul, G., Bozoglan, H., Ogut, T. S., & Balci, M. K. A clinician's artificial organ? Instant messaging applications in medical care. The International Journal of Artificial Organs, 40(9), pp. 477–480
[14] Sonnentag, S., Reinecke, L., Mata, J., & Vorderer, P. Feeling interrupted—Being responsive: How online messages relate to affect at work. Journal of Organizational Behavior, 39(3), pp. 369–383, 2018
[15] Ou, C. X. J., & Davison, R. M. Interactive or interruptive? Instant messaging at work. Decision Support Systems, 52(1), pp. 61–72, 2011

Agnieszka Żarko[1], Paweł Cielma[2], Małgorzata Plechawska-Wójcik[3]

A TOOL SUPPORTING PRIMER GENERATION IN THE STUDY OF DNA METHYLATION

Keywords: MSP, DNA methylation, NCBI, in silico

Abstract

Epigenetics is one of the most dynamically growing areas of genetics. Geneticists, laboratory assistants and scientists dealing with the field of epigenetics need technological support to facilitate their work. There is a need of tools supporting the entire process of gene analysis. The paper presents the process of the generation of primers to research on DNA methylation. To support this process, a dedicated application was developed. The application is supported by researchers dealing with the methylation process, including isolated DNA strands. The analysis procedure covers processes such as generating PCR/MSP primers, comparing genes, writing antisense strands or detecting CpG islands. The research covers the case study on the generation of primers in three different sequences of isolated DNA. For the purpose of this study, three species were chosen: the human (Homo sapiens), the red fox (Vulpes vulpes), and the domestic dog (Canis lupus familiaris). The analysis was performed on MC1R gene sequences downloaded from biomedical databases (NCBI) in the form of FASTA files. The paper presents the complete process of gene sequence analysis using the developed application. The application is a unique tool allowing analysis of a broader spectrum of problems and processes operating in genomes, instead of focusing on one single subject.

1. Introduction

Nowadays, many fields of science are developing thanks to modern technologies. Bioinformatics is one of them. This interdisciplinary science allows effective problem solving and advances in research on a variety of essential matters concerning living organisms. One of the studies that bioinformatics explores is epigenetics – a study of functions, structures and expression of genes. The research spans a wide range of subjects, including, for example, epigenetic changes in DNA sequence such as DNA methylation. The extensive body of research is meticulously analysed to aid the development of the field, which in turn will help eradicate many medical conditions or bad habits. Epigenetics is one of the most dynamically growing areas of genetics, which may revolutionise the modern world [1].

[1] Lublin University of Technology, Department of Computer Science, Nadbystrzycka 36B, 20-618 Lublin, Poland, cielmapawel@gmail.com
[2] Lublin University of Technology, Department of Computer Science, Nadbystrzycka 36B, 20-618 Lublin, Poland, agnieszkazarko@gmail.com
[3] Lublin University of Technology, Department of Computer Science, Nadbystrzycka 36B, 20-618 Lublin, Poland, m.plechawska@pollub.pl

Geneticists, laboratory assistants and scientists dealing with the field of epigenetics need technological support to facilitate their work. The biotechnology market is constantly expanding and many applications facilitating research are available. However, until now there has been no accessible tool that would allow the analysis of a broader spectrum of problems and processes operating in genomes, instead of focusing on one single subject [2].

On the market, there are various programmes dedicated to carrying out epigenetic research. Some of them are available online. One of the most popular programmes is MethPrimer. This application is designed to generate BSP and MSP primers. The programme works by taking a DNA sequence and checking it for CpG islands. Once they are found, an analysis is performed to detect possible primers. MethPrimer, however, has a limit to the number of nucleotides that can be introduced into the programme. Similar programmes dedicated to PCR primers designing are: Primo MSP and Methyl Primer Express Software. Another application for DNA research is the CpG Island Searcher. It is focused mainly on CpG island detection, however, it does not provide full automation as it requires from a user a range of information. Another available software is the BiSearch application which supports the designing of PCR, BSP and MSP primers considering different simulation parameters. Compared to the presented applications, the programme that is the subject of this study enables a comprehensive implementation of research on DNA methylation.

The purpose of this study has been to design and implement an application generating primers for the purpose of research on DNA methylation. The programme facilitates research on gene expression, which will result in the development of the branches of genetics associated with, for example, carcinogenesis and ageing.

The article is divided into several sections. The first section presents the theories of epigenetics and bioinformatics databases. The second one details the created application. The third section describes the methodology and the data analysis process. The last part focuses on the resulting conclusions.

2. Epigenetics

Modern genetics originated in the 1950s when Francis Crick and James Watson discovered that all living organisms use double-helix DNA to communicate information [3]. DNA is formed of four types of nucleobases. The universality of the genetic code spawned a number of discoveries and theories, and also explained the causes of some diseases. Nevertheless, the following years of research proved that the order of the nucleotide sequence is not the only aspect of the regulation of biological processes.

In mammals, there is an identical nucleotide sequence within each cell. The only exceptions from this rule are gametes – sperm cells (1n, no mitochondrial DNA) and ova (1n, mtDNA) – and erythrocytes which lack a cell nucleus.

Despite containing only one genome, cells of an organism become specialised to form different types of tissues and organs. This diversification originates in embryonic development when most of the zygote and blastomere lose their totipotency. Unlike plant cells, most animal cells are not capable of reversing the process of histogenesis; however, some researchers question this theory [4].

Selective expression and silencing certain genes result in the creation of unique protein sets in particular types of cells. Transformations of DNA occurring in the process are strictly related to the interactions of epigenetic nature. Currently, there are several modifications thought to be responsible for this: the most basic methylation (separate processes in animals and plants), parental genome stigma, histone modifications (acetylation, ubiquitination, methylation, phosphorylation), dosage compensation, Trithorax and Polycomb protein correlation, and RNA interference. All those interactions create an epigenotype unique to each organism [5], [6].

2.1. Methylation

Methylation is the most common post-replicative DNA modification. It requires the participation of a group of enzymes called DNA methyltransferases. Methylation is classified as an epigenetic transformation (Greek *epi* – above, over), which means that it is not caused by the nucleotide sequence on its own. It consists in the attachment of a methyl group (-CH3) to specific nucleotides. This process is fully reversible and responsible for the long term deactivation of selected genes. Global demethylation during gamete production is the best example of the process. To prevent accidental methylation, nucleic acids are equipped with protective measures to reverse the process. The main protective factor is DNA demethylase. Its effectiveness is essential – if it misses damaged proto-oncogenes or suppressor genes, this may lead to carcinogenesis. These mechanisms are observed in tumours, which makes oncologists see the possibilities of applying DNA demethylase to eliminate pathogenic cells [7].

The process of DNA methylation is enabled by specific enzymes – DNA methyltransferases whose function is to attach methyl groups to nucleobases. There are several variants: the most common C5-methylcytosine (m5C), N4-methylcytosine (m4C), and N6-methyladenine (m6A). These are typical of specific groups of organisms. In a eukaryote, m5C occurs; in a prokaryote – m4C, m5C, and m6A. Additionally, in lower eukaryote families, N6-methyladenine can also be found. Their genomes are deactivated to a much lesser extent than in other eukaryotes; sometimes they show no sign of modifications. One such species is the fruit fly (Drosophila melanogaster) [8].

In humans, the most significant modified nucleotide is C5-methylcytosine (m5C). It is found in so called CpG islands (5'CpG'3 sequences), in the promoter section of the DNA which is the strategic place where gene expression begins. CpG islands have a characteristic abundance of CpG dinucleotides, and their

length usually falls between 100 and 1500 base pairs. Statistically, around 5% of cytosines in mammal organisms are permanently methylated – which roughly corresponds to 70–80% of modified nucleotides in CpG regions. This number is not constant; for example, a positive correlation between the degree of methylation and the age of an organism has been noted [9,10].

Currently, the most popular method of genetic research is Polymerase Chain Reaction, acronymised to PCR [11,12]. There are various, specialised types of this process. In the case of epigenetic studies, it is MSP (Methylation-specific PCR). MSP is based on the comparative method. There need to be two sets of primers which are short (16–25 bp) segments of nucleotides complementary to a given part of the DNA chain. If a given sample contains the sought sequence, a reaction will occur. This results in producing a greater amount of the researched gene, which can be detected at later stages (e.g. gel electrophoresis) [13–15].

2.2. Bioinformatics databases

To improve the accessibility of data and to facilitate transfer, there have emerged international bioinformatics databases. These store, for instance, DNA sequences for different organisms that have been provided by scientists from around the world. There are three essential databases [16]:

- the American GenBank,
- the European Molecular Biology Laboratory (EMBL),
- the DNA Databank of Japan (DDBJ).

Each of them operates at a different research institute which is responsible for their maintenance and the system of data actualisation. Together, they form INSDC – International Nucleotide Sequence Database Collaboration – and exchange all new information, creating a worldwide resource with open access for every scientist. The stored data are shared in the form of text files over FTP clients or Internet tools enabling work online.

The most popular and easiest format to represent data is FASTA (standing for FAST-All, i.e. being applicable to sequencing any type of alphabet). With small modifications, it may be utilised by computer programmes, at the same time being user-friendly. FASTA is applied in GenBank, founded in the Los Alamos National Laboratory in 1982. GenBank was later relocated to NCBI, the National Center for Biotechology Information. It provides tools for bioinformatics data processing. The sequences published on the website are mostly derived from high-speed sequencing centres, individual laboratories, the US Patent and Trademark Office, and INSDC actualisations. Their updated version is shared every two months.

3. The application

The application has been developed for the purpose of research on the DNA strand. It is called “Met611” and based on given functional and non-functional

requirements and a detailed project. The documentation abundant in BPMN, DFD and ERD diagrams supported the implementation of the programme. The application is written in Java and works with different operating systems, thanks to the use of a virtual machine.

The application consists of four modules: user record, session operation, functionality and genetic module. The user record concerns registration, the possibility to edit or delete a user account, changes of a password and saving research. It is very useful for laboratory assistants who can resume a task at any time or compare results of subsequent research. All data are stored in the cloud, easily accessible via the Internet. Users of the application should not be afraid of data loss on their accounts, being protected with md5 encrypted passwords. The next module, session operation, comprises logging in and out as well as shutting down the programme.

The main purpose of the programme is the analysis of DNA, especially its methylated version, with the aid of a number of functions. The first step while working with the programme is to manually enter a nucleotide sequence or to search for it in one of the bioinformatics databases. The default database, NCBI, is the most popular one. By selecting a specific gene, a FASTA file is downloaded and then converted into a format that is readable by the programme. The file provides information such as the genetic code, the gene name, the source organism, and the nucleotide sequence.

The material prepared in this way can be deployed for the following functions:

- writing antisense strands,
- comparing genes,
- generating primers for the purpose of MSP.

The primary function, which is designing primers, is depicted in Figure 1. As shown, the user can modify the conditions of the process. There are options to set the following parameters:

- melting point,
- primer length,
- the size of CpG islands,
- the algorithm for calculations of primer temperature,
- the density of CG dinucleotides.

After making all necessary calculations, the programme displays a summary covering the areas abundant in CpG islands (presented in Fig. 3) and primers with their characteristics. Moreover, the results are adjusted to clear the duplicates of primer pairs.

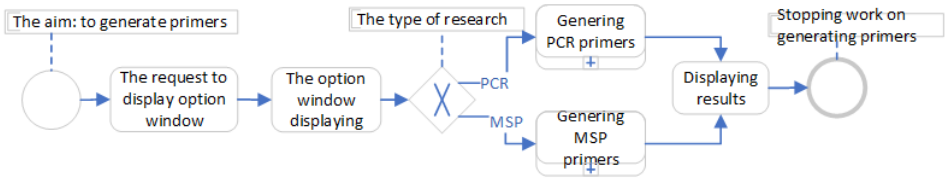


Fig. 2. The scheme of primer generation.

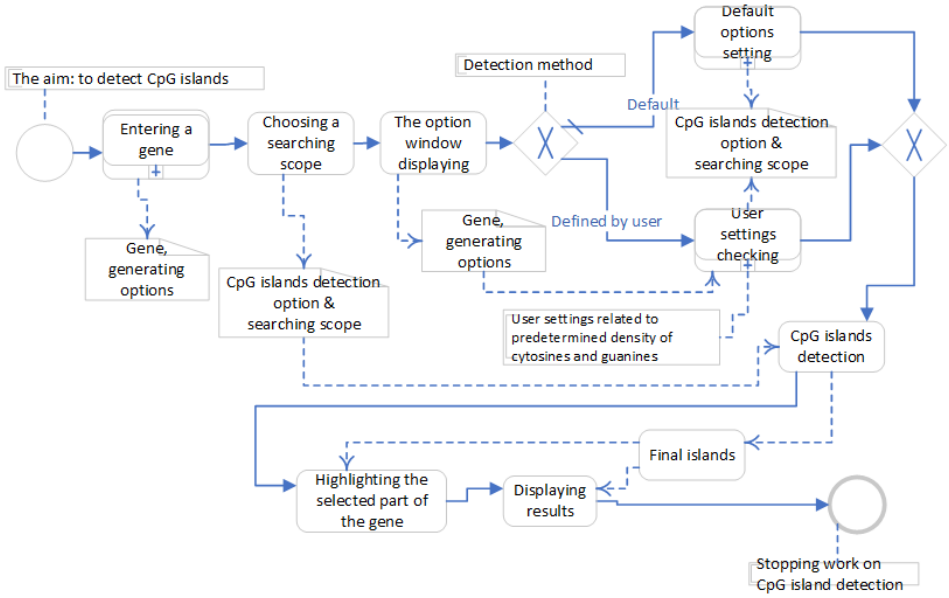


Fig. 1. The scheme of CpG islands detection.

Comparing genes is a simple function allowing, for instance, for the search for differences in genes between various organisms. This indicates the extent to which designed primers may be universal to each domain of species. The functionality of writing an antisense strand revolves around creating other part of DNA, based on the rule of nucleotide complementarity.

The last module of the programme allows to keep clear documentation in the PDF format. The user can include particular elements of research and the summative file can be saved directly on their account or the computer hard drive.

The genetic module is the most important as it is the essence of the application. It facilitates research by carrying out a range of crucial operations and allows for the analysis and comparison of results. Notably, this module features the possibility to manually enter any genetic sequence (Fig. 3). Another way of finding a gene is to look it up in one of the three main bioinformatics databases, i.e. NCBI, EMBL, and DDBJ.

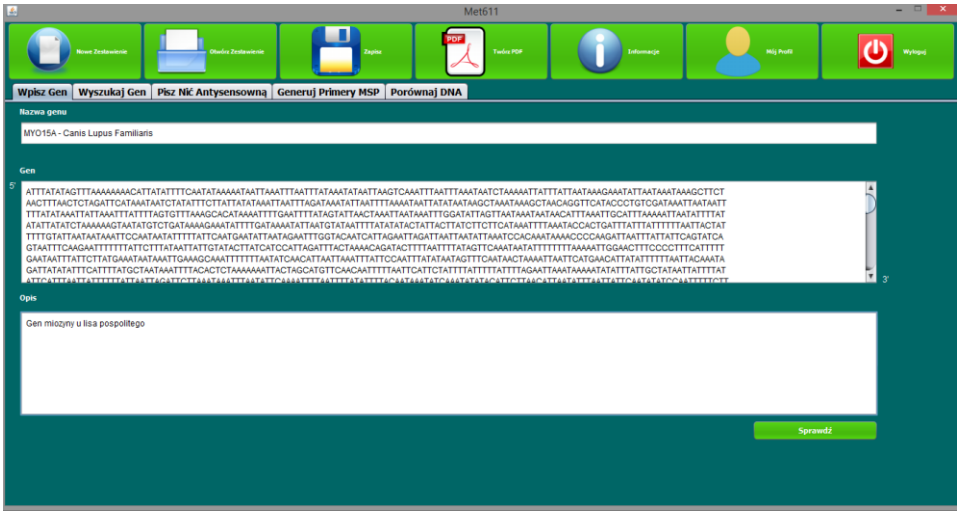


Fig. 2. Entering a gene into the programme.

Whenever one intends to collate two studied genes, the gene comparing function should be deployed. Another functionality is antisense strand writing. Anyone working with the programme can transform a chosen DNA sequence into an antisense strand. The next function is the detection of CpG islands which are found as cytosine and guanine nucleotide-rich sites. The application aptly detects islands of the predetermined density of cytosines and guanines. This is a crucial component of the programme as primers are generated on the basis of those CpG islands. The programme works by analysing a DNA sequence and then checking it for CpG islands. These places are calculated using an algorithm that checks the data flow. Once they are found, an analysis is performed to detect possible primers. A user can narrow the search area, which will result in faster and more precise analysis.

An important constituent of this module is the possibility to select any area to search for CpG islands or primers. The key element of the programme is generating primers; it allows for the designing of PCR and MSP primers. The ability to calculate their melting points is an additional advantage (Fig. 4).

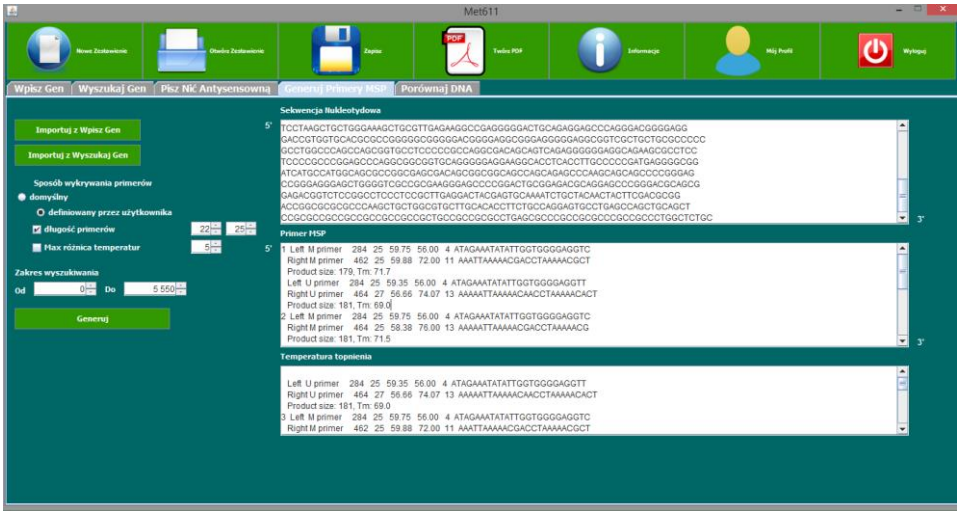


Fig. 3. Generating MSP primers.

The functionality module, in turn, serves to represent the results of conducted research with the aid of graphs or PDF files.

4. Data analysis

The subject of the study was a desktop application to generate primers for MSP research, as described in Section 3. The programme used the data sourced from the bioinformatics database NCBI and transformed the FASTA format into the processable type of information. Subsequently, CpG islands were detected in the obtained nucleotide sequence, and then complementary primers were designed. The data produced in this process were examined and analysed.

4.1. Test environment

The research employed the application running on two different test stations. Both computers were connected to the Internet to allow logging into the user's account in the cloud where all data from the previous research conducted by the assistant were stored. Additionally, Internet access was necessary for downloading the selected genes on which the study was carried out. Table 1 presents what sequences were downloaded from NCBI in the form of FASTA files. It was the MC1R gene which codes the melanocortin 1 receptor [17]. Different versions of this gene can be found in various species. For the purpose of this study, three species were chosen: the human (Homo sapiens), the red fox (Vulpes vulpes), and the domestic dog (Canis lupus familiaris).

Table 1. Specification of analysed genes.

Homo sapiens – MC1R	
Definition	Homo sapiens melanocortin 1 receptor(MC1R), mRNA
Access	NM_002386
Size	3115 bp mRNA linear
Authors	Yu S, Yu HS, Tu HP, Wu CS and Lan CC.
Source	J Eur Acad Dermatol Venereol 30 (12), E204-E205 (2016)
Vulpes vulpes – MC1R	
Definition	Vulpes vulpes isolate VVU melanocortin receptor type 1 (MC1R) gene, complete cds.
Access	JX083391
Size	954 bp DNA linear
Authors	Nowacka-Woszuk, J., Salamon, S., Gorna, A. and Switonski, M.
Source	Submitted (20-May-2012) Department of Genetics and Animal Breeding, Poznań University of Life Sciences, Wołyńska 33, Poznań, 60-637, Poland
Canis lupus familiaris – MC1R	
Definition	Canis lupus familiaris isolate L44-B melanocortin 1 receptor (MC1R) gene, complete cds.
Access	KC332686
Size	954 bp DNA linear
Authors	PLoS ONE 8 (2), E55469 (2013)

During the research, the following parameters were set:

- the size of CpG islands>100 pz
- the density of dinucleotides CG>50%
- the size of primers: 20-30 bp
- the maximum difference in the temperature of primers: 5.

4.2. Results

The outcomes of the research are gathered in Table 2. The essential data obtained in the study include the number and sizes of the detected CpG islands as well as the number of primers generated by the application.

Table 2. The research results.

	Homo sapiens	Vulpes vulpes	Canis lupus familiaris
Detected CpG islands	3	2	2
Size of the largest CpG island (bp)	240	217	188
Average CpG island size (bp)	212	169.5	170.5
Number of MSP primers (pairs)	5	4	5
Annotations	-	The application detected a duplicated pair and dismissed it	-

The sizes of CpG islands are illustrated in the form of a box plot, as shown in Figure 5. The box plot was generated by means of a Matlab simulation.

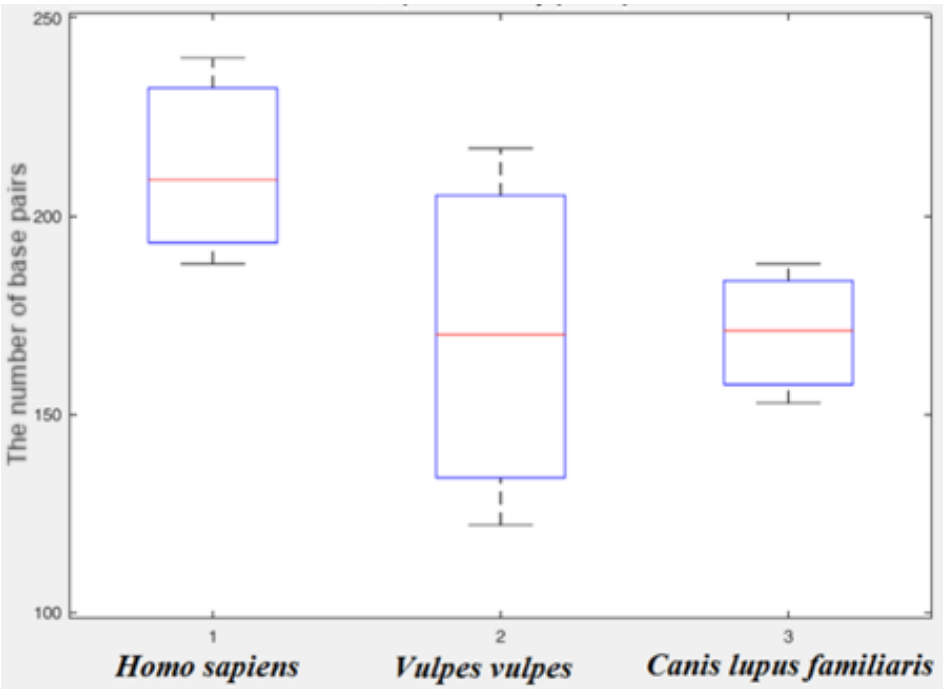


Fig. 5. The number of base pairs in CpG islands.

The results indicate that, despite the difference in the length of particular genes – the longest sequence was 4,5 times longer than the shortest one – the programme did not detect a proportionally greater number of primers as it can be predicted from their lengths. This was caused by the parameters of algorithms, and not by any limitations in the number of obtained results.

Undoubtedly, the advantage of the programme is that it has proved that one of the generated pairs of primers is a duplicate of an existing one. This fact allows for the elimination of a labour intensive stage of searching for such an error in other programmes. During the analyses, a drawback of the programme was discovered – it misconverted the information in the FASTA format. The application converted only a part of nucleotide symbols, i.e. A, C, G, T, U, R, Y, K, M, S, W. It omitted nucleotide ambiguity codes such as B, D, H, V (standing for sets of three possible nucleotides) or N (signifying a set of all four possible nucleotides). Another issue that needs to be taken into consideration is the existence of breaks of undetermined length. Those are not dependent on the programme itself but are caused by the imperfections in the gene sequencing.

5. Conclusions

The research on the generation of primers on the three different sequences of the isolated DNA described in the paper shows that the programme works correctly. It turns out that the application is of high utility due to its handling of extensive components. The programme can dismiss doubled primers, which makes the results clearer. The work on generating primers in itself is not a complicated task; however, one needs to possess expertise in epigenetics to know what has the highest scientific relevance. In the analysis of the obtained results, the additional information on the generated primers such as their temperature or possible duplication proved to be very useful. A valuable suggestion for users of the application is to accurately set the parameters affecting the outcome of their research. Moreover, the good mechanism of data theft protection ensures that user's research will not be misused by third parties. The penultimate section of the paper mentions some issues revealing the marginal imperfections of the application. The delineated difficulties in the conversion from the FASTA format should be eradicated to prevent any errors.

The DNA strands used in this research were downloaded from the bioinformatics database of NCBI. Thanks to linking the application with accessible databases, the search for genes has become easier and less time-consuming. When comparing the obtained data concerning the genetic codes of the human, the red fox, and the domestic dog, one notices that the number of detected CpG islands in the MC1R gene is the greatest in Homo sapiens and equals three. The other two species have the same number of islands, that is two. The differences in the size of the islands between the species are corresponding to

those in their numbers. On the basis of the CpG islands, MSP primers were successfully generated, which was one of the aims of this study.

References

[1] S. Rogalska, M. Achrem, A. Wojciechowski, „Chromatyna. Molekularne mechanizmy epigenetyczne [Chromatin. Molecular Mechanisms in Epigenetics]". Wydawnictwo Uniwersytetu Przyrodniczego w Poznaniu, 2010

[2] M. Łukasik, J. Karmalska, M. Szutowski, J. Łukaszkiewicz, "Wpływ metylacji na funkcjonowanie genomu [The Impact of Methylation on the Functioning of Genome]." Biuletyn Wydziału Farmaceutycznego Warszawskiego Uniwersytetu Medycznego, vol. 2, pp. 13–18, 2009

[3] T. A. Brown, "Genomes", Taylor & Francis Inc, 2012

[4] M. Gryzinska, A. Jakubczak, R. Stryjecki, G. Jeżewska-Witkowska, "In silico Analysis of Methylation of Selected Genes Using Computer Programs Based on Various Analytical Techniques." Biocybernetics and Biomedical Engineering, vol. 35, pp. 169–175, 201.

[5] A. Sulewska, W. Niklinska, M. Kozlowski, L. Minarowski, W. Naumnik, J. Niklinski, L. Chyczewski, "Detection of DNA Methylation in Eukaryotic Cells." Folia histochemica at cytobiologica, vol. 45(4), pp. 315-324, 2004

[6] M. Esteller, "DNA Methylation. Epigenetics and Metastasis". Springer, 2005

[7] S. Moran, C. Arribas, M. Esteller, "Validation of a DNA Methylation Microarray for 850,000 CpG Sites of the Human Genome Enriched in Enhancer Sequences." Epigenomics, vol. 8(3), pp. 389–99, 2016

[8] M. D. Long, D. J. Smiraglia, M. J. Campbell, "The genomic impact of DNA CpG methylation on gene expression; relationships in prostate cancer." Biomolecules, vol. 7.1, pp. 15, 2017

[9] Y. Liu, X. Jin, Y. Li, Y. Ruan, Y. Lu, M. Yang, B. Dong, "DNA Methylation of Claudin-6 Promotes Breast Cancer Cell Migration and Invasion by Recruiting MeCP2 and Deacetylating H3Ac and H4Ac." Journal of Experimental & Clinical Cancer Research, vol. 35(1), pp. 1–13, 2016

[10] W. C. Mah, C. G. Lee, "DNA mMethylation: Potential Biomarker in Hepatocellular Carcinoma." Biomarker Research, vol. 2(1), pp. 5, 2014

[11] A. M. Lim, I. L. Candiloro, N. Wong, M. Collins, H. Do, E.A. Takano, S. Kleid, "Quantitative Methodology Is Critical for Assessing DNA Methylation and Impacts on Correlation with Patient Outcome." Clinical Epigenetics, vol. 6, pp.1–22, 2014

[12] R. Slomski, "Analiza DNA. Teoria i praktyka [DNA Analysis. Theory and Practice]". Wydawnictwo Uniwersytetu Przyrodniczego w Poznaniu, 2011

[13] M. X. Draht, K. M. Smits, V. Jooste, B. Tournier, M. Vervoort, C. Ramaekers, V. Melotte, "Analysis of RET Promoter CpG Island Methylation Using Methylation-Specific PCR (MSP), Pyrosequencing, and Methylation-Sensitive High-Resolution Melting (MS-HRM): Impact on Stage II Colon Cancer Patient Outcome." Clinical Epigenetics, vol. 8(1), pp. 44, 2016

[14] R. V. Pandey, W. Pulverer, R. Kallmeyer, G. Beikircher, S. Pabinger, A. Kriegner, A. Weinhäusel, "MSP-HTPrimer: a High-throughput Primer Design Tool to Improve Assay Design for DNA Methylation Analysis in Epigenetics." Clinical Epigenetics, vol. 8(1), pp. 101, 2016

[15] L. Long-Cheng, D. Rajvir, “MethPrimer: Designing Primers for Methylation PCRs”, Bioinformatics vol. 18(11), pp. 1427–1431, 2002
[16] A. Gruza, „Bionformatics Databases”, PJWSTK, 2010
[17] L. P. Fernandez, R. Milne, J. Bravo, J. Lopez, J. Avilés, M. Longo, J. Benítez, P. Lázaro, G. Ribas, "MC1R: Three Novel Variants Identified in a Malignant Melanoma Association Study in the Spanish Population." Carcinogenesis, vol. 28 (8), pp. 1659–1664, 2007

Mateusz Mazurek, Jakub Mielniczuk, Marcin Badurowicz[1]

COMPARISON OF POTHOLE DETECTION ALGORITHMS BASED ON ACCELEROMETER DATA

Keywords: road artefact, accelerometer, detection.

Abstract

In this paper the authors are presenting the results of meta-analysis of a several algorithms used in a problem of road artefacts detection (e.g. potholes) based on the accelerometer data. The compared algorithms were implemented from scratch in a black-box method in a new computer system, and the same dataset has been used for comparisons. Six algorithms were compared on a dataset of 12 road fragments, with calculation of sensitivity, precision, false positive rate and time of execution per 100 datapoints. No significant winner has been found, but important notes about usage of the particular solutions has been noted.

1. Introduction

Road quality is one of the values to be taken into consideration when thinking about road transport security and comfort. It has a direct consequence in number of road accidents and cars' overall technical state change in the time of their usage. To cope with data acquisition in terms of road quality, several researchers has been discussing using accelerometer data and car-mounted devices, to augment usage of road profilers. So-called road artefacts, e.g. potholes, are a serious problem in a day-to-day driving, but also for professional and business cases, so different solutions to find them and publish the information have been proposed. In this paper, the authors concentrate on finding the road artefacts based on a usage of accelerometers mounted in cars. The base idea is that the accelerometer is measuring current acceleration values during the car's movement over the road. Several solutions in this topic have been already discussed since 2008 [1,2], but the authors are basing this research mostly on the already performed experiments during the CRADIA project [3]. In the presented solution, the device for data acquisition is a smartphone, recording both the acceleration values using the integrated accelerometer as well as geographical location using the integrated GSNS (Global Satellite Navigation System) device and a proper custom software. Such data presents the acceleration values directly corresponding to the current location, so it is possible to mark locations as a location for road artefacts. Other

[1] Lublin University of Technology, Department of Computer Science, Nadbystrzycka 36B, Lublin, Poland, m.badurowicz@pollub.pl

solutions are also discussed, e.g. usage of cameras or even an off-the-shelf RGB+Depth cameras like Microsoft Kinect [4].

1.1 Related research and scope selection

In this paper, the authors would like to concentrate on one of the aspects of the road quality assessment using accelerometer data – on the difference between several algorithms used for road artefacts detection. While similar meta-analysis have been performed, either due to establish the state-of-the-art [5] or just as a byproduct of creating a new approach [6]–[8], there is not yet comparison between different road artefacts detection algorithms performed on the known dataset, all statistics data has to been compared between different research performed on different datasets in different years. The authors have decided to concentrate on the road artefacts detection with algorithms using different thresholds, instead of classification-based methods [9,10], which is the topic in line with the research already being published by the authors earlier.

As for the lack of the comparison based on the known dataset, the authors decided to reimplement all approaches from scratch, in a clean-room manner, basing only on a description of the selected algorithms in the related papers.

To narrow the scope of the presented research, out of all of the proposed solutions already available in the works in the described fields, the authors decided to implement and compare six algorithms: MOD-Z-THRESH, F-THRESH, Z-THRESH, G-ZERO, STDEV and Z-DIFF, as the most commonly used baselines for searching better solutions. All of them are trying to detect the outliers in the accelerometer data, based on the thresholds – so where the acceleration is exceeding the threshold, the outlier – possible road artefact – is detected. The most analyzed is the Z-axis acceleration, which corresponds to the car movement upwards, perpendicular to the Earth's surface. That means the orientation of the data is particularly important factor in the research, which has been discussed in the previous works [11]. In this paper, the data have been reoriented using the rotation matrix by usage of magnetometer data, which is described later in the section 3.1.

The compared algorithms may be divided into two groups – first of them have the static threshold value, while in the second the threshold value is dynamically calculated for the analyzed datapoint, based on the current data, the difference between currently analyzed datapoint and their neighborhood or the overall road quality. The difference between the static and dynamic approach will be an important factor later in the results analysis.

1.1.1 Z-THRESH

The Z-THRESH algorithm [12] is the most basic of the set of algorithms decided to be compared. It is based on the principle of static threshold. If the current value of acceleration in Z axis, which is perpendicular to the Earth's

surface, and is corresponding to the movement of the car up or down is exceeding the threshold, as presented in the Figure 1 (red), the analyzed data point is being marked as possible road artefact. Such simple solution is highly dependent on the chosen threshold value.

1.1.2 Z-DIFF

The Z-DIFF [12] is based on the difference between two subsequent datapoints, as presented in the Figure 1 (green). The absolute difference of raw acceleration value in the Z axis between two subsequent datapoints is being compared to the defined static threshold. The possible modifications include dividing the time difference between the datapoints.

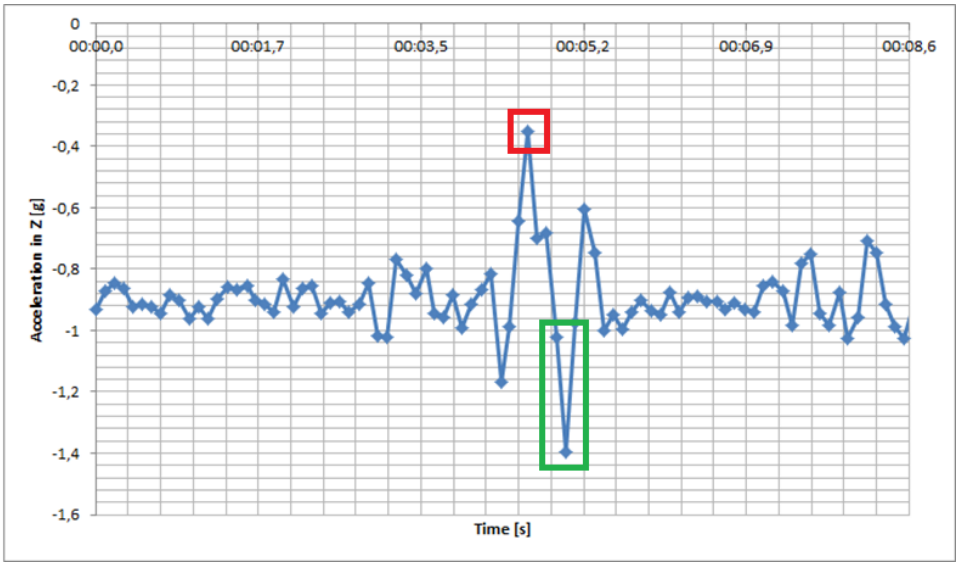


Fig. 1. Screenshot of the application used in the experiments.

1.1.3 STDEV

The STDEV [12] is based on the standard deviation in Z-axis, which is also being compared to the defined static threshold. The possible road artefact is based on the first datapoint from data window being analyzed. As it is one of the algorithms based on data window technique, it is necessary to set up the proper data window size.

1.1.4 G-ZERO

G-ZERO is the last of the methods described in [12], based on detection of free fall movement. The free fall is a moment where the body is being only accelerated by the Earth's gravitational pull and is being detected where all three values of

acceleration are equal to zero. Because of very short time of driving over the pothole, it is heavily dependent on sampling frequency, so the better way than detection of pure zero may be detection of all three values below a defined threshold.

1.1.5 MOD-Z-THRESH

The MOD-Z-THRESH is a modified version of Z-THRESH [13], in which instead of comparing the pure values of Z acceleration, the value marked Z_s is used, calculated as such:

$$Z_s = |Z - avg(Z)| \tag{1}$$

Where Z is an acceleration value in the Z (perpendicular to the Earth's surface) axis, and avg(Z) is an average value of the Z acceleration in the analyzed data window. In addition, the threshold is not static, but is calculated as a multiplication of standard deviation of Z-axis acceleration by a static factor, which was experimentally decided to be of a value of 4.3, as this value was performing the best in the described implementation.

1.1.6 F-THRESH

The last one is the F-THRESH algorithm [3], the most complex one of threshold-based algorithms described in this paper, being the derivation of MOD-Z-THRESH. It is being divided into two parts – in the first is dividing the Z-axis acceleration data into non-overlapping data windows (tumbling windows), and for each of these windows, the standard deviation of Z-axis acceleration as well as road quality measurement (in the scale from 0 to 7) is being calculated. The second part is to calculate threshold value for every of datapoints of the analyzed windows. The threshold value is calculated by usage of fuzzy system, combining the calculated road quality and Z_s value (1) and compared to every datapoint, which – when exceeded – is marked as a road artefact.

2 Comparison metrics and methodology

The metrics used to compare the algorithms were based on [6], and are mostly based on a confusion matrix, as all the algorithms presented in the previous sections are performing binary-classification. That means the result of the classification for the single data point is binary – there is a road artefact in this data point or there is not. If there is a road artefact in the current location and the analyzed algorithm marked the current data point as such, it is classified as a True Positive (TP). True Negative (TN) is if there is no road artefact at a given location and the algorithm decided as such, False Positive (FP) is where there is no artefact but the algorithm classified there is, and False Negative (FN) – where there is a road artefact but it was not detected by the algorithm.

In the performed meta-analysis the most important metric was number of False Negatives, which is showing the number of wrong classifications. True Positives number and False Positives number were used in additional metrics, described later, while TN were mostly unused.

2.1 Sensitivity and precision

Sensitivity is describing the proportion of positive (good) classifications to the sum of True Positives and False Negatives.

In the context of the presented paper, sensitivity is a proportion of a number of points correctly classified to a number of really existing road artefacts in the analyzed road fragment, showing the percentage of correctly detected road artefacts for a given road fragment.

Precision on the other hand, a proportion of True Positives to sum of True Positives and False Positives is defined in the context of this paper, as what is a proportion of correctly classified points to the number of all points which are classified as road artefacts, thus showing the percentage of correct classifications performed by analyzed algorithm.

2.2 False Positive Rate

False Positive Rate (FPR) is an important metric to consider if the classification should be performed locally, e.g. on the Internet of Things (IoT) device itself – in the proposed scenario, for each positive classification data must be sent to the cloud processing system for further analysis, that means the huge percentage of false positives may introduce high data transmission rate. In the case of local alerting system, high percentage of false positives may lead to ignoring alerts by the user, which is also not desirable. FPR is calculated as such:

2.3 Computation time

Computation time (CW) is a metric important in the context of streaming processing, if the algorithms were to be used on-the-fly data classification. The authors decided to divide the overall computation time by number of data points in the analyzed road fragment, and then to calculate the value, dubbed CW. CW is a measurement how many milliseconds is needed to classify each 100 datapoints, in average. Thanks to this approach, the CW will be independent of the analyzed road fragment. Of course, time of computation is highly dependent on the final implementation, so in the case in the presented research, all algorithms have been implemented by the authors from scratch (black box), only based on the algorithms' description in the literature.

3 Experiments and results

The application used for experiments have been implemented by the authors from scratch and is based on the data source called CRADIA, which is a project the authors have been working since 2013 and is publicly available [3]. The application is being fed with road fragment acceleration data in the CSV format, performing classification using the implemented algorithms, and presenting results as a map as well as statistics and metrics, as presented in the Figure 2.

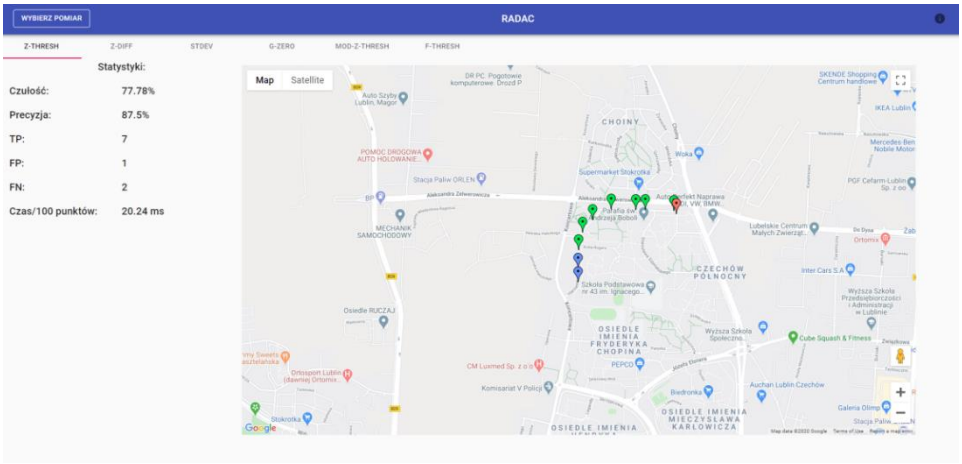


Fig. 2. Screenshot of the application used in the experiments.

The application was built as a web-based implementation of algorithms, in the Python Language, with a front-end using the Google Maps platform to present the results of road artefacts detection using visual pushpins placed on a map in different colors (green for true positive, red for false positive, blue for false negative); as well as statistical information about the number of found artefacts, and calculated sensitivity, precision and computation time.

3.1 Testing dataset

CRADIA dataset is being divided into 12 road fragments, each of them divided into two sections – the first being the raw values of acceleration in device-local X, Y, Z axis, reoriented global N, E, Z2 axis [11], current geographical location and timestamp, as presented in the Table 1. The second part is a raw list of geographical locations of confirmed road artefacts by the human labeling process. The accelerometer and gyroscope acquisition frequency are 10 Hz, but such GNSS (Global Satellite Navigation System) resolution in the acquisition device was not available and is about 1 Hz. Thus, for each geographical point there may be multiple acceleration values – to cope with that, the authors decided if any of acceleration points in the current geographical location is classified positive, that

whole geographical location is classified positive. That means that TP/TN/FP/FN classification is based on the geographical locations, not only acceleration values.

Table 1. CRADIA example dataset fragment, not including Speed, Time and Accuracy.

X	Y	Z	N	E	Z2	Lat	Lon
0,057	0,044	-0,060	0,058	0,023	-0,070	51,223	22,493
0,096	-0,062	0,124	0,096	-0,044	0,132	51,223	22,493
0,020	-0,030	0,002	0,017	-0,031	0,008	51,223	22,493
0,063	0,030	0,106	0,072	0,046	0,094	51,223	22,493
0,057	0,044	-0,060	0,058	0,023	-0,070	51,223	22,493
0,096	-0,062	0,124	0,096	-0,044	0,132	51,223	22,493

Source: own work

3.2 Aggregated results

For each road fragment available in the CRADIA dataset, the application has been run 3 times. As all of the algorithms are strictly deterministic, sensitivity, precision and FPR were unchanged, however time of computation (CW) has been averaged. In the Table 2, the aggregated results for all analyzed metrics are presented.

Table 2. Calculated statistic parameters of the algorithms on the whole dataset.

Algorithm	Sensitivity	Precision	FPR
Z-THRESH	80.24%	36.75%	11.75%
Z-DIFF	89.40%	31.99%	16.5%
STDEV	87.07%	51.53%	7.75%
G-ZERO	93.01%	38.54%	13.17%
MOD-Z-THRESH	68.33%	48.4%	5.67%
F-THRESH	45.10%	35.06%	5.83%

Source: own work

As one may notice, the sensitivity for the four first algorithms is between 80 and 95%, where the highest value is for G-ZERO, exceeding 90%, while two algorithms with a dynamically calculated threshold placed in the last positions.

In case of precision, the only algorithm with the result over 50% was STDEV while MOD-Z-THRESH was a close second and Z-DIFF ranked last with the result of only 31.99%.

When comparing the false positive rate, the differences between compared approaches are clearly visible: the dynamically-calculated threshold algorithms, both MOD-Z-THRESH and F-THRESH are ranking the lowest, where Z-DIFF is over 16.5%, that means that more than one in every 10 alerts is actually an unnecessary alert.

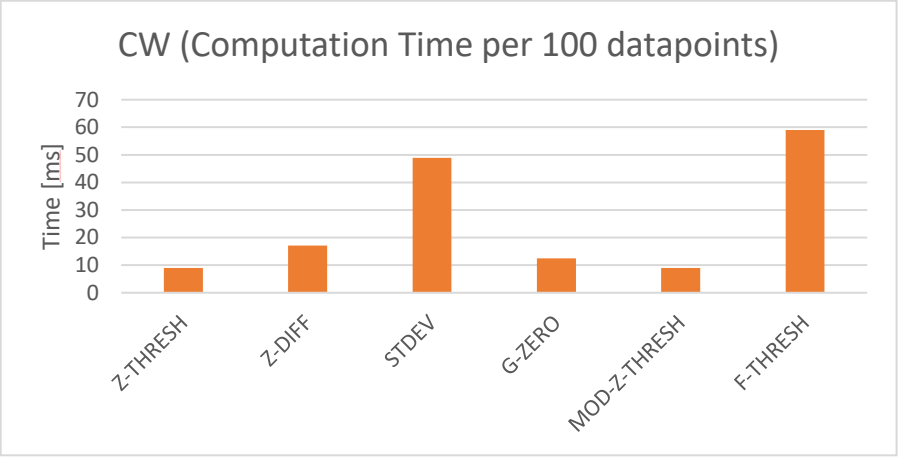


Fig. 3. Computation time per 100 datapoints (CW) for the analyzed algorithms.

Finally, when comparing the classification time, the MOD-Z-THRESH and Z-THRESH were ranked first, where STDEV and F-THRESH were drastically slower from the rest, which is not surprising taking into consideration the complex mathematical operations performed in both of the lowest ranking algorithms in that manner.

4 Conclusion and discussion

One very important difference that may be noticed is the availability of two groups of algorithms – with dynamically calculated and static threshold.

In the Table 6, the differences in average sensitivity and false positive rate may be clearly seen. Static threshold algorithms are performing well in the percentage of correctly classified data points, or – properly found road artefacts, where sensitivity is in 80–95% range compared to 68% for MOD-Z-THRESH and only 45% for F-THRESH. However, when taking into account the number of false positives, the statistics are completely inverted.

Based on the statistics presented in the table 6, the conclusion is that static-threshold algorithms are less elastic – while they are very good, very sensitive, they are giving really high number of false alerts, and their performance is highly

dependent on the defined threshold. Dynamic-threshold algorithms are very rigoristic, taking into consideration more than pure acceleration values, that means they are marking overall less datapoints as possible alerts – but also with a drastically lower number of false alerts.

When implementing solutions based on the presented algorithms, STDEV is very interesting proposition to be taken into consideration – it has a similar sensitivity as Z-DIFF and G-ZERO, but with better precision (over 50%) and with lower number of false positives. However, its classification time is exceptionally low compared to other static-threshold algorithms, being 2 to 3 times slower. The MOD-Z-THRESH is faster and with slightly better values of all provided metrics except sensitivity.

The final decision on which algorithm should be used in the road artefacts detection problem is an open question. The authors are not able to provide final decision, however several points are important:

- In general, static-threshold algorithms are slightly better in terms of sensitivity, however the cost is the higher number of false positives,
- Static-threshold algorithms are very dependent on defined threshold,
- Complex calculations are very slow and may not be implementable in the IoT Edge scenarios, like in the case of STDEV and F-THRESH,
- Dynamic-threshold algorithms are better in the false positives rate, however at the cost of lower sensitivity,

5 Future work

Based on these conclusions, the authors believe that the best algorithm for threshold-based detection of road artefact may be some kind of hybrid approach, like the MOD-Z-THRESH, a modification of Z-THRESH. However, more and more analysis, especially on different data sources, must be performed and will be a goal of the authors in the near future.

The presented paper is not including any other method for road artefacts classification other than thresholding – there is no comparison of Support Vector Machines (SVM) or isolation forest techniques. The authors have already started working on their own implementations of some of the earlier described methods to include them in the second meta-analysis.

References

[1] P. Mohan, V. N. Padmanabhan, and R. Ramjee, “TrafficSense : Rich Monitoring of Road and Traffic Conditions using Mobile Smartphones,” *The 6th ACM Conference on Embedded Networked Sensor Systems*. pp. 1–29, 2008

[2] P. Mohan, V. N. Padmanabhan, and R. Ramjee, “Nericell: Rich Monitoring of Road and Traffic Conditions using Mobile Smartphones,” *Proceedings of the 6th ACM conference on Embedded network sensor systems – SenSys ’08*. p. 323, 2008

[3] M. Badurowicz, J. Montusiewicz, and P. Karczmarek, “Detection of Road Artefacts

Using Fuzzy Adaptive Thresholding," in *2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1–8, 2020

[4] V. S. K. P. Varma, S. Adarsh, K. I. Ramachandran, and B. B. Nair, "Real Time Detection of Speed Hump/Bump and Distance Estimation with Deep Learning using GPU and ZED Stereo Camera," *Procedia Comput. Sci.*, vol. 143, pp. 988–997, 2018

[5] T. Nguyen, B. Lechner, and Y. D. Wong, "Response-based methods to measure road surface irregularity: a state-of-the-art review," *Eur. Transp. Res. Rev.*, vol. 11, no. 1, 2019

[6] G. Singh, D. Bansal, S. Sofat, and N. Aggarwal, "Smart patrolling: An efficient road surface monitoring using smartphone sensors and crowdsourcing," *Pervasive Mob. Comput.*, vol. 40, pp. 71–88, 2017

[7] A. Allouch, A. Koubaa, T. Abbes, and A. Ammar, "RoadSense: Smartphone Application to Estimate Road Conditions Using Accelerometer and Gyroscope," *IEEE Sens. J.*, vol. 17, no. 13, pp. 4231–4238, 2017

[8] M. Y. Alam *et al.*, "Crowdsourcing from the True crowd: Device, vehicle, road-surface and driving independent road profiling from smartphone sensors," *Pervasive Mob. Comput.*, vol. 61, 2020

[9] V. K. Nguyen, É. Renault, and V. H. Ha, "Road Anomaly Detection Using Smartphone: A Brief Analysis," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11005 LNCS, pp. 86–97, 2019

[10] N. Van Khang and E. Renault, "Cooperative sensing and analysis for a smart pothole detection," *2019 15th Int. Wirel. Commun. Mob. Comput. Conf. IWCMC 2019*, pp. 1785–1790, 2019

[11] M. Badurowicz and J. Montusiewicz, "Identifying Road Artefacts with Mobile Devices," in *Information and Software Technologies, ICIST 2015 [WOS]*, R. Damasevicius and G. Dregvaite, Eds. Berlin, Germany: Springer, pp. 503–514, 2015

[12] A. Mednis, G. Strazdins, R. Zviedris, G. Kanonirs, and L. Selavo, "Real time pothole detection using Android smartphones with accelerometers," *2011 Int. Conf. Distrib. Comput. Sens. Syst. Work. DCOSS'11*, 2011

[13] M. Badurowicz, T. Cieplak, and J. Montusiewicz, "The Cloud Computing Stream Analysis System for Road Artefacts Detection," in *Computer Networks: 23rd International Conference, CN 2016, Brun{ó}w, Poland, June 14-17, 2016, Proceedings*, P. Gaj, A. Kwiecień, and P. Stera, Eds. Cham: Springer International Publishing, pp. 360–369, 2016

Bernadetta Stachura-Terlecka[1], Antoni Ligeza[2]

EPIDEMIC MODELLING WITH CELLULAR AUTOMATA

Keywords: epidemic, cellular automata, epidemic modelling, simulation.

Abstract

This paper focuses on a novel approach to modelling epidemics, putting them into a framework of a kind of spatial dynamic conflict system. An approach based on Layered Competitive Cellular Automata (LCCA) is used for that purpose. Epidemic modelling is an extremely complex process requiring taking into consideration multiple factors, measures taken to prevent the disease from spreading, methods of treatment, as well as infectivity time, are also significant factors. In the considered case study, there influenza viruses (type A, B, and type C) were analysed. The fourth type of influenza virus (type D) is not considered in the simulation or in the study as it causes attacks cattle. Cellular Automata are a highly suitable method for models incorporating so many different factors influencing at least one aspect of the model or, in some cases, on all its aspects. A simple yet expressive computer model has been developed and implemented in Python. The model covers the environment layer, the epidemic development layer, and the medical treatment layer. The visualisation of example results is presented on the website on a contour map.

1. Introduction

Due to the increasing influence of the anti-vaccine movements, some countries have been observing the reappearance of new centres of measles and tuberculosis. One can hear about this comeback not only in the so-called Third World countries, but outbreaks tend to occur more and more also in developing countries, for example in Poland and Ukraine. However, there are few widely available tools allowing to model the development of epidemics and presenting graphically the efforts made in order to cope with those diseases. Such tools should allow not only to present the outbreak development and to assess the probability of its occurrence but also to demonstrate how essential it is to reach the vaccination threshold.

This article focuses on a novel approach to modelling epidemics, putting them into a framework of a kind of spatial conflict, and uses Layered Competitive Cellular Automata (LCCA) for that purpose.

[1] AGH University of Science and Technology, Department of Applied Computer Science, Al. A.Mickiewicza 30, 30-059 Kraków, Poland bstachur@agh.edu.pl
[2] AGH University of Science and Technology, Department of Applied Computer Science, Al. A.Mickiewicza 30, 30-059 Kraków, Poland, ligeza@agh.edu.pl

2. State of the art

2.1. Cellular Automata

2.1.1. Theory

Cellular Automata are considered as alternative mathematics, the great advantage of which is that the calculations performed by the CA are free of rounding errors. A standard cellular automaton consists of a network of single cells i and is defined by a set of states and rules defining the change of state at time t depending on the cell possessed by the cell at time t-1 and the states of its neighbours. In the case of 2D cellular machines, several types of neighbourhoods can be distinguished (Fig. 1).

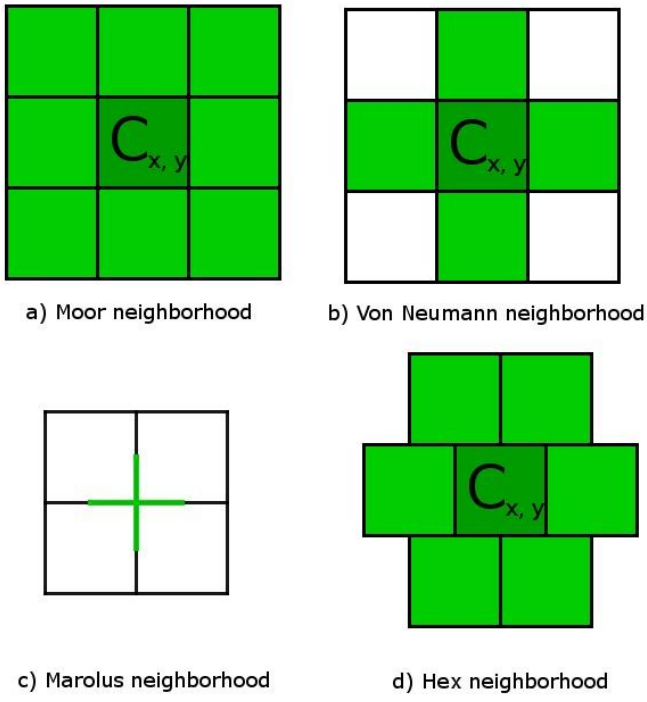


Fig. 1. Types of the neighbourhood [1].

Cellular Automata (CA) are usually described with the help of two numbers, k – which is the number of states that a cell can reach, and r – the surroundings, which specifies the distance from the cell to its nearest neighbours. The best-known cellular automaton is Conway's Game of Life (k = 2, r = 1), thus CA Life can be represented with the following tuple: (2, 1) [6].

2.1.2. Practical application of CA

Nowadays, cellular automata have changed from the role of entertainment for mathematicians into tools for simulation of various types of phenomena, including distributed physical and biological systems, cryptography, etc. Examples of CA applications in modelling are:

- Epidemic modelling (implementation of SIR, SIS, SEIR models) [9],
- Game of Life,
- Model for the evolution of DNA sequences [7],
- Lattice gas automaton (HPP model) [8],
- downhill skiing [11],
- oil spill spreading disasters [13], Building evacuation [12],
- Scheduling [10].

2.2. Epidemic theory

Epidemics are the occurrences of diseases caused by viruses or bacteria and other health-related events. However, to consider a given phenomenon as an epidemic, the occurrence of a phenomenon or disease in a given territory must be higher than before [4].

A virus is a microorganism that has no cell structure, parasites in the living organism, and causes disease. Let us define some important terms concerning viruses and epidemics which will be used in the following sections:

- Virulence, microbial virulence – the ability to penetrate, multiply, and damage tissues of the infected organism by a given type of pathogen. Individual strains of a given species may differ in their virulence,
- Contagiousness – the ability of a pathogen to move between carriers,
- Lifetime – is the time the pathogen can survive outside the host organism,
- Grim resistance – a situation in which vaccination of some members of the population provides measurable protection against people who are susceptible to infectious disease and who have not been vaccinated,
- Immunisation – the minimum number of people vaccinated necessary to obtain a group immunity.

2.3. Influenza virus

The influenza virus [2, 3] has four varieties: A, B, C, D, of which D attacks cattle, while A, B, and C – people. Of these three types of viruses, type C is the mildest and usually goes asymptomatically or with mild symptoms. In our research, we consider the influenza epidemic of two types with the highest virulence, i.e. type A and B. The following factors should be considered when modelling:

- for viruses:
  - temperature
  - humidity
  - the statistical distribution of virus types (probability of encounter virus type A or B)
  - the level of virulence
  - the virus's lifetime outside the host organism depending on the type of substrate
- for treatment and prevention:
  - the total number of people vaccinated during the period considered
  - the length of the latent period for influenza type A and B
  - the duration of the disease in the absence of treatment and its use
  - the time for which the body gains resistance to infection after recovery

The basic formulas used in the influenza model according to NIZP-PZH [2] and CDC [3] data for 2016:

- Vaccination effectiveness – ~40% (depending on the type)
- Number of cases in 2016 – ~4,300,000
- Number of people vaccinated in 2016 – 857029
- Percent of vaccinated – ~2.26%
- Probability of meeting a vaccinated person – 0.0226
- Probability of meeting an unvaccinated person – 0.9774
- A chance for meeting influenza A virus – 0.6 – 0.7
- A chance for meeting influenza B virus – 0.3 – 0.4
- Lifetime influenza type A and B – about 24 hours depending on the ground
- Virulence, microbial virulence – Influenza type A – large, type B – medium.
- For over a dozen years, Cellular Automata have been used to model the development of influenza and other epidemics, e.g.:
- a two-layer cellular automaton modelling
- influenza in an organism [14],
- 2D cellular automaton used in the modelling of bird flu in Indonesia [16],
- modelling of spreading influenza using cellular and agent-based automata [15].

2.4. Simulation software

We currently have several systems available to simulate the spread of the epidemic. One such system is GLEaMviz. The system is designed to model the course of an epidemic without considering the outbreak of diseases transmitted by vectors dependent on local contact patterns. GLEaMviz uses a stochastic and discrete calculation scheme to model the spread of an epidemic called "GLEaM". – the global epidemic and mobility model [18].

A stochastic SEIR model to model the spread of the Sars-cov2 virus epidemic and the lung disease caused by it, Covid-19. In the study, the researchers assumed that one person enters an enclave of 1000 people [19].

3. Solution of the problem

Layered Competitive Cellular Automata (LCCA), thanks to its structure and principle of operation, allows modelling of various types of spatial conflicts, and not only epidemics themselves, which makes them a universal modelling tool. The types of conflicts we can model by using LCCA are, for example:

- natural disasters (floods, fires),
- epidemics,
- terrorism,
- damage to critical infrastructure,
- technical conflicts (e.g. removing oil stains in the sea),
- conflicts in computer games (as one of the parties).

The use of a variable number of layers depending on the number of conflicting parties allows modelling conflicts of various sizes. The use of pLCCA allows a user to visualise the course of the conflict on contour maps.

Epidemic modelling is a complex process requiring taking into consideration multiple factors [5] and the type of influenza virus. These factors determine the virulence, infectivity, and the virus lifetime outside the host's organism. Measures taken in order to prevent the disease from spreading, namely vaccinations and their efficacy, the number of vaccinated people, methods of treatment as well as infectivity time are also significant factors. It is essential to consider all those components before modelling an epidemic. In the considered case study, two influenza viruses – hereinafter called virus A and B – were analysed. In our solution, an epidemic is considered as a spatial dynamical conflict. The side of the conflicts are:

- Virus (in these problems is Influenza virus type A and B – else type they are not considered).
- Medicine (drugs and vaccines), and immunity.

Epidemic modelling (as a specific type of conflict) was based on the Layered Competitive Cellular Automata (LCCA).

3.1. Describe of implementation

The Layered Competitive Cellular Automata (LCCA) is a cellular automaton with $n+1$ layer, where n is the number of conflict sides, and the size of the layer hinges on the size of the simulation field. The standard LCCA includes a map layer (GRID), which usually retrieves selected information from the PostGIS databases or is loaded from a text file of a map or raster maps. For simulations of large areas, a variation of LCCA – pLCCA (pixel Layered Competitive Cellular

Automata) can be used, in which a single cell corresponds to a single pixel on the screen.

The first and further layers contain information about the algorithms and models necessary to perform the simulation. Each layer can consider a different set of external factors. The influence of the factors may apply only to one of the layers, to several selected layers, or all layers representing the parties of the conflict. A cell in LCCA is multi-layered. The layer system and the idea of a multi-layer cell are presented in Figure 2, and an exemplary implementation of Lst. 1.

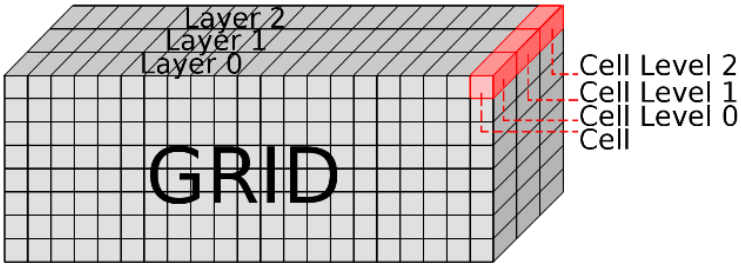


Fig. 2. Graphic presentation for grid use in LCCA and Multilayer Cell [1].

```
class Cell:   def __init__(self):
   self.level = [CellLevel(), CellLevel(), CellLevel()]   def replace_cell(self):
return self
```

Lst. 1. Example implementation for Multi-layer Cell [1].

A single cell in the grid (red markings in Fig. 2) consists of three cells (three layers). Thanks to these layers, we can model the operation of different conditions in one space and time for individual sides, as well as model their impact on each other (in-depth) without losing necessary data for modelling the operation of a specific page within one layer (in a wider).

The standard neighbourhood in LCCA is a combination of Moor's neighbourhood within the same layer and Von Neumann between the layers. The neighbourhood looks like in Figure 3.

The LCCA is defined as Eq 1.

$$\text{LCCAutomaton} = (Nx, Ny, D, \delta, \{M\,Cij : i = 1, 2, \ldots, M\,Cij : i = 1, 2, \ldots, Nx, j = 1, 2, \ldots, Ny\})) \quad (1)$$

where N x and N y are positive integers denying the size of the grid (identical to all levels), k is the number of active levels (recall that the level 0 is environment layer), D is a domain of some basic accessible elements, δ is a mapping: δ: {1, 2, . . . , N x } × {1, 2, . . . , N y } → D denes the zero-level environment, and M Cx,y multi-level cells assigned to positions over the whole grid.

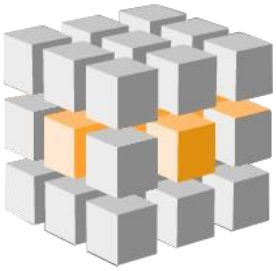


Fig. 3. LCCA neighbourhood.

3.2. Model formulation

During the simulation of the spread of the influenza virus epidemic, we have to consider separately both types A (Eq. 3) and B (Eq. 4). They differ in many factors, the most important of which are:

- length of virus survival outside the host organism,
- virulence,
- the probability of a virus of a given type,
- the effectiveness of vaccination.

$$Z(C1(n))=1/x * y *ZarA *W \quad (3)$$

$$Z(C1(n))=1/x * y *ZarB *W \quad (4)$$

Where:
Z(C1(n)) – the virus infectivity coefficient in a given cell,
1/x – coefficients depending on temperature for the type
A and B virus y – humidity dependent coefficients for type A and B virus
Zar B – infectiousness of influenza B virus
Zar A – infectiousness of influenza A virus
W – virulence depending on the type of influenza A – high, B – average
The x and y coefficients for a given virus or its type are selected based on the analysis of the number of cases in previous years, the average temperature, and humidity during the same period.

3.3. Model study

The data collections used to create patterns of influenza virus spread come from the Polish National Institute for Hygiene (NIZP-PZH) [2] for the number of influenza cases and the Institute of Meteorology and Water Management (IMGW) [17] where data on average monthly temperatures and average humidity coming from. For modelling spatial conflicts, we developed a dedicated environment implementing the LCCA concept. The software is written in the Python

programming language, and the visualisation is carried out using the Flask framework, which task is to present the results on the website.

By running the simulation, a user enters the values of humidity, temperature, the number of people vaccinated in the area covered by the modelling and then runs the simulation. During modelling, the following steps are performed, which can be divided into two stages.

First of them is preparation for starting the simulation:

- Cells in which there is a virus such as influenza are randomly chosen.
- Cells in which people are found are randomly chosen.

The second of them is the main simulation: For cells in which influenza viruses are found, the maximum virus lifetime is determined depending on the type of ground, information about which is stored in the Map layer. At each iteration, the time variable is reduced by 1 until the time reaches 0. After this event, the information in the cell is set to the lack of a virus. It is checked if the higher layer cell, in which the human was placed, contains a virus. In such a case, we randomly determine if the unit was vaccinated and what type of virus is in the upper layer. After these activities, the infectivity coefficient is calculated. Based on this coefficient, it is determined whether the person has flu or not. In the case of a person who does not get sick, the time_vir parameter is set to 4 (hours). In the opposite case (an infected person), it is set to 216. The time_vir parameter tells the time in which an individual having contact with the virus will spread it.

3.4. Model formulation

The simulation results are presented on the website on the contour map. From this page, it is also possible to manage the main modelling parameters. An exemplary result of flu epidemic simulation is presented in Figure 4 and Figure 5.

Green colour represented resistant persons (e.g. vaccinated and those who have already been ill). Red colour represented infected or sick persons. The intensity of the colours depends on the time the unit will be infected (red) or will be resistant to infection (green). Other people are not presented on the map. The graph below the map shows the number of cases. The shape of the graph demonstrates that the number of cases bottom out in the summer months.

The shape of the graph was changed when the analyst changed humidity or temperature. The largest increase in the incidence of influenza in Poland is observed when the air temperature ranges from about 0 degrees Celsius to about 10 degrees Celsius with humidity in the range of about 75% to 90%, a similarly large increase in the incidence of influenza is observed at temperatures from about -10 degrees Celsius to 0 and air humidity above 80%.

The main result of the modelling is the infected units marked on the map and the recovered units which are immune or resistant due to previous vaccination.

All simulations were performed under the assumption that the percentage of people vaccinated is constant.

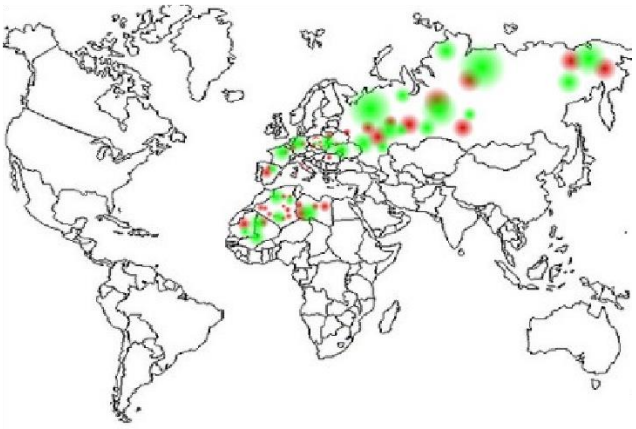


Fig. 4. Results of the sample simulation 1/x * y = 30.

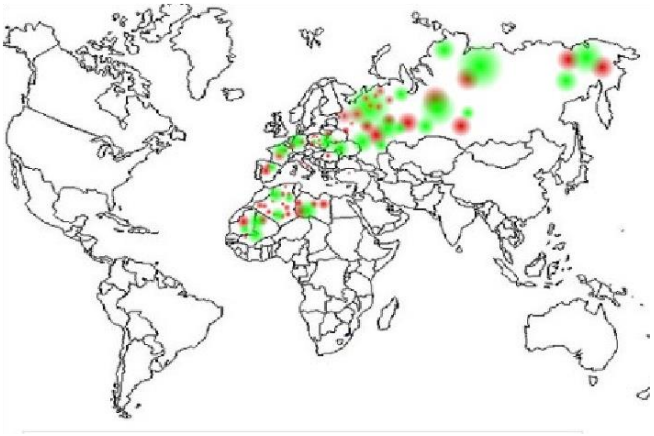


Fig. 5. Results of the sample simulation 1/x * y = 40.

4. Conclusions and future work

Epidemic modelling is an example of conflict modelling. In this case, it is a conflict between resistance and the achievements of medicine, and viruses. By modelling the epidemic, we have a chance to find the so-called cut-off threshold that should be reached to prevent outbreaks of epidemy.

In the future, we plan to create models based on LCCA for outbreaks of other types of viruses. It seems that the proposed approach can be used as a generic tool for modelling, analysis, and visualisation of dynamic, spatial, conflicting phenomena. As such, it can become a valuable tool in the fight against natural disasters or military applications.

References

[1] B. Stachura-Terlecka, A. Ligęza, Modelling and analysis of spatial conflicts with Layered Competitive Cellular Automata, *Multimedia Communications, Services and Security* 2015

[2] Polish National Institute for Hygiene – NIZP-PZH: http://wwwold.pzh.gov.pl/oldpage/epimeld/grypa/index.htm

[3] CDC https://www.cdc.gov/flu/

[4] A. Zieliński, What do we mean by developing an epidemiological focus?, Polish National Institute for Hygiene – NIZP-PZH

[5] J. D. Murray, *Introduction to Biomathematics*, Wydawnictwo Naukowe PWN, 2006

[6] K. Kułakowski, *Cellular Automata*, OEN AGH, 2000

[7] P. Chaudhuri, D. Chowdhury, S. Nandi, S. Chattopadhyay, *Additive Cellular Automata: Theory and Applications, Volume 1*

[8] D. Wolf-Gladrow, *Lattice-Gas Cellular Automata and Lattice Boltzmann Models: An Introduction,* 2000

[9] S. Whitea, A. Martín del Rey, G. Rodríguez Sánchez, Modelling epidemics using cellular automata, *Applied Mathematics and Computation Volume 186,* 193–202, 2007

[10] E. Kucharska, K. Grobler-Dębska, K. Rączka, L. Dutkiewicz, Cellular Automata approach for parallel machine scheduling problem, 2016

[11] T. Korecki, D. Pałka, J. Wąs, Adaptation of Social Force Model for simulation of downhill skiing, 2016

[12] I. Koumis, I. Georgoudas, G. Trunfio, J. Wąs, G. Sirakoulis, A GPU Implemented 3F Cellular Automata-Based Model for a 2D Evacuation Simulation Pattern, 2017

[13] M. Gług, J. Wąs, Modelling of oil spill spreading disasters using a combination of Langrangian discrete particle algorithm with Cellular Automata approach, 2018

[14] C. Beauchemin, J. Samuel, J. Tuszynskia, A simple cellular automaton model for influenza A viral infections, *Journal of Theoretical Biology Volume 232, Issue 2*, 223–234, 2005

[15] S. Emrich, F. Breitenecker, G. Zauner, N. Popper, Simulation of influenza epidemics with a hybrid model – combining cellular automata and agent-based features, ITI 2008 – 30th International Conference on Information Technology Interfaces, 2008

[16] H. Situngkir, EPIDEMIOLOGY THROUGH CELLULAR AUTOMATA Case of Study: Avian Influenza in Indonesia, (not publishing), 2004

[17] Institute of Meteorology and Water Management (IMGW) – www.imgw.pl

[18] he GLEaMviz computational tool, a publicly available software to explore realistic epidemic spreading scenarios at the global scale Wouter Van den Broeck, Corrado Gioannini, Bruno Gonçalves, Marco Quaggiotto, Vittoria Colizza & Alessandro Vespignani BMC Infectious Diseases

[19] A Simulation on Potential Secondary Spread of Novel Coronavirus in an Exported Country Using a Stochastic Epidemic SEIR Model, Kentaro Iwata, Chisato Miyakoshi, Special Issue Real-Time Clinical and Epidemiological Investigations on Novel Coronavirus