

ISSN 2300-5149

# PRZEGLĄD TELEINFORMATYCZNY

Ł. Mielniczuk, Ł. Strzelecki Method of determining the longest simple chain in a graph with the use of a genetic algorithm .....	3
M. Melaniuk Badanie możliwości wykorzystania metody porównawczej do zapobiegania skutkom ataków typu DNS Injection .....	19
A. Arciuch, A.M. Donigiewicz Quality study of user activity using mobile device. Tap, double tap, flick gestures .....	37
Information for Authors – rules of papers preparation and reviewing for Teleinformatics Review .....	73
Informacje dla autorów – zasady przygotowania tekstu i recenzowania artykułów do Przeglądu Teleinformatycznego .....	75

PRZEGLĄD TELEINFORMATYCZNY  
TELEINFORMATICS REVIEW

Dawniej: BIULETYN INSTYTUTU AUTOMATYKI I ROBOTYKI WAT  
(ISSN 1427-3578)  
Ukazuje się od 1995 r.

RADA NAUKOWA

Lt. Col. Janos Balogh MSc  
dr hab. inż. Antoni M. Donigiewicz – przewodniczący  
Hacene Fouchal, PhD  
prof. Lech J. Janczewski, DEng  
prof. dr hab. inż. Włodzimierz Kwiatkowski  
prof. dr hab. inż. Bohdan Macukow  
Lt. Col. Lajos Mucha PhD  
prof. ing. Vladimír Olej, CSc.



Ministerstwo Nauki  
i Szkolnictwa Wyższego

UMOWA NR 630/P-DUN/2018 Z MNiSW (z dn. 21.06.2018 r.) na lata 2018-2019:

- wzrost liczby artykułów anglojęzycznych;
- obsługa strony internetowej czasopisma na platformie IC, nadawanie artykułom naukowym numerów DOI i wykorzystanie profesjonalnego panelu edycyjnego.

Wymienione zadania finansowane są w ramach umowy nr 630/P-DUN/2018 ze środków Ministra Nauki i Szkolnictwa Wyższego przeznaczonych na działalność upowszechniającą naukę.

ADRES REDAKCJI

Redakcja Przeglądu Teleinformatycznego  
00-908 Warszawa, ul. Gen. W. Urbanowicza 2  
tel. 261 83 95 52, fax. 261 83 71 44  
e-mail: pt [at] ita.wat.edu.pl  
WWW: <http://przeglad.ita.wat.edu.pl/>

Wersją pierwotną czasopisma jest wersja elektroniczna

REDAKTOR NACZELNY:

Antoni Donigiewicz

REDAKTOR WYDANIA

Antoni Donigiewicz

OPRACOWANIE STYLISTYCZNE

Renata Borkowska

PROJEKT OKŁADKI

Barbara Chruszczyk

WYDAWCA: Instytut Teleinformatyki i Automatyki WAT

**ISSN 2300-5149**

**ISSN 2353-9836 (on-line)**

# Method of determining the longest simple chain in a graph with the use of a genetic algorithm

**Łukasz MIELNICZUK, Łukasz STRZELECKI**

Institute of Teleinformatics and Automation, Faculty of Cybernetics,  
Military University of Technology  
ul. Gen. W. Urbanowicza 2, 00-908 Warszawa  
lukasz.mielniczuk@wat.edu.pl, lukasz.strzelecki@wat.edu.pl

**ABSTRACT:** This paper discusses the issue of determining the longest simple chain in a graph by using a heuristic algorithm – a genetic algorithm. A method enabling the effective determination of the longest chain in any connected, undirected graph without loops.

**KEYWORDS:** graph theory, Hamiltonian chain, heuristic algorithm, genetic algorithm

## 1. Introduction

According to the definition by B. Korzan [2], there may be multiple longest simple chains in a graph, although unlike maximum simple chains, they are all of the same length. This means that each longest chain is a maximum chain, while the reverse is false. If the given graph contains a Hamiltonian chain, it is at the same time the maximum and longest chain of the graph.

The problem of finding the longest simple chain in a graph can be reduced to the problem of finding a Hamiltonian chain or the longest simple chain that is not a Hamiltonian chain. Unlike the shortest chain problem, where algorithms exist that enable this problem to be solved in polynomial time, the problem of finding a Hamiltonian chain is an NP-complete problem.

It must be noted that there are multiple algorithms for finding the longest chain in a graph. They differ in the method used, and in their computational complexity. They are both analytical algorithms, such as the Roberts-Flores algorithm [3], and heuristic algorithms, an example being the ant colony algorithm.

The purpose of this paper is to present a method for heuristic determination of the longest chain in a graph using a genetic algorithm.

## 2. Graph properties

The graph definition applied in this paper follows the interpretation used by B. Korzan [2]. For the purposes of order and clarity, it is also given in this section. The definitions of *path*, *chain*, *simple chain* and *Hamiltonian chain* are given as well – they too follow the interpretation used by B. Korzan.

As this paper only refers to simple graphs, it is possible to adopt the definition of a graph as an ordered pair  $G = \langle W, U \rangle$ . Issues related to using the method described herein for directed graphs will be addressed in another paper.

From a practical standpoint, a graph can therefore be a model of a situation where we deal with two distinct sets,  $W$  and  $U$ , of elements of an object, with elements  $u \in U$  representing distinct bonds between certain elements  $w \in W$ , which means that element  $u$  binds two elements  $\{x, y\} \subseteq W$ . We therefore refer to set  $W$  as the graph node set, and to set  $U$  as the graph edge set.

*Node degree*  $s(x)$  is the number of graph edges that are incident to node  $x$ . Node degree is therefore equal to the number of all incoming and outgoing arcs, edges and loops that connect to the given node [2].

Path  $G = \langle W, U \rangle$  in a graph is any alternating sequence of nodes  $x_{i_s} \in W$  and edges  $u_{i_s} \in U$  leading to children  $x_{i_s}$ , taking the form of:

$$\{x_{i_0}, u_{i_1}, x_{i_1}, u_{i_2}, x_{i_2}, u_{i_3}, \dots, x_{i_{l-1}}, u_{i_l}, x_{i_l}\},$$

such that for every  $s \in \{1, \dots, l\}$  there exists edge connecting nodes  $x_s$  and  $x_{s+1}$ .

The number of edges  $l$  present in the sequence defining the path is called the path length.

A specific case of path is a *chain*, which is a path connecting two graph nodes in such a manner that all edges along this path are different. A chain  $\mathbb{L}_i(x_{i_0}, x_{i_1})$  with an initial node  $x_{i_0}$  and terminal node  $x_{i_1}$  is therefore any path

$\{x_{i_0}, u_{i_1}, x_{i_1}, u_{i_2}, x_{i_2}, u_{i_3}, \dots, x_{i_{l-1}}, u_{i_l}, x_{i_l}\}$  that meets this condition. Chain length is the number of the chain's edges, therefore the maximum chain length in a graph with  $m$  edges is  $m$ .

A *simple chain* is a chain where all nodes are different. This means that the maximum simple chain in a graph with  $n$  nodes can have a length no greater than  $n-1$ . A maximum simple chain is any simple chain that is not part of another simple chain in the graph<sup>1</sup>. This definition implies that there may exist multiple maximum simple chains of different lengths in a graph.

A *Hamiltonian chain* is a simple chain that includes all nodes of a graph, thus if in a simple graph there is a longest simple chain  $L(x_o, x_l) = \{x_o, u_1, x_1, \dots, x_{l-1}, u_l, x_l\}$  with a length of  $l \geq 2$  that meets the condition  $s(x_o) + s(x_l) \geq l + 1$ , then this chain is a Hamiltonian chain.

### 3. Principle of genetic algorithms

As issues related to the genetic algorithm will be used later in the paper, the mechanism of its functioning is shown below. This description has been prepared according to the classic interpretation of genetic algorithms, as presented by D. Goldberg [1].

Genetic algorithms are based on the mechanisms of natural selection and inheritance. They combine two principles that apply in nature - a greater survival chance of fitter organisms and their systematic development caused by information exchange (hybridisation) and individual mutation.

The algorithm assumes the existence of a certain initial population of individuals, which correspond to solutions to a problem. The initial population is generated randomly. The solutions are then given scores according to pre-defined criteria. During the next step, another population of individuals is generated, combining the characteristics of the best solutions from the preceding step. The analogy to the world of nature is clearly visible here.

The algorithm effectively utilises the experience of the previous populations to determine a new search area, for which the quality of the solutions is expected to be higher. The solutions in the given population that are scored higher than the others, are more likely to form a so-called parent pool for

---

<sup>1</sup> This means that a maximum simple chain in its entirety cannot be a fragment of another, longer chain.

the new generation. The next iteration of the algorithm will be performed on a dataset whose quality should be higher than in the preceding steps. The genetic algorithm searches for the best solution across a specific number of iterations, a pre-defined time, or until the optimum solution is found.

### **3.1. Basic operations used in a genetic algorithm**

When performing a task using a genetic algorithm, you should begin by generating a set of random solutions that form the initial population. These solutions may take various forms – for example, numbers or tables. Solutions are also frequently presented as binary vectors.

The population size can be chosen freely and needs to be specified when solving the task. A population that contains a small number of individuals may make finding the best solution impossible, while an excessive number of individuals will significantly extend the search duration.

The prepared datasets forming the population are subsequently subjected to operations corresponding to processes occurring to chromosomes. These are: reproduction, hybridisation and mutation.

During the reproduction process, individual datasets forming the solution are reproduced and hybridised with one another. The first stage of this process is solution evaluation. Function  $f$ , which is an objective function (in biology, it is referred to as the fitness function), is used for this purpose. This function is a certain measure of the utility of the solution whose value we wish to maximise – the better the solution, the higher the value of the objective function.

Once all solutions have been given scores, the so-called selection needs to be performed, where individuals to be reproduced and form the parent pool for the next solution population are selected. Selection may be performed according to one of many strategies. The basic strategy is the so-called roulette method, where the probability of an individual being selected for the reproduction pool depends on the evaluation function score - the higher the value of the objective function for the given solution, the greater the probability it will be selected for the next generation.

The first step is to select the number and locations of hybridisation points. This is done based on the strategy chosen. The basic strategy assumes that the number of points is determined first, after which they are arranged at random. It is also possible, as in the selection process, to use mathematical distributions, according to which the hybridisation points are selected.

Next, divide the solutions into groups within which hybridisation will be performed. The simplest method is to select pairs of individuals, then hybridise them with each other, although hybridising more solutions at the same time may be employed as well.

Using hybridisation exclusively may cause potentially good solutions to be eliminated by narrowing the search set. To avoid this effect, so-called mutation is used when searching for the optimum solution. This involves randomly changing values in certain solutions. For bit sequences, this can be a change of values at specific positions in the sequence from 0 to 1 or vice versa, for tables it may, for example, be changing the values of selected elements in the table.

Figure 1 below shows a flowchart of a genetic algorithm.

## **4. Developing a method for finding the longest chain in a graph using a genetic algorithm**

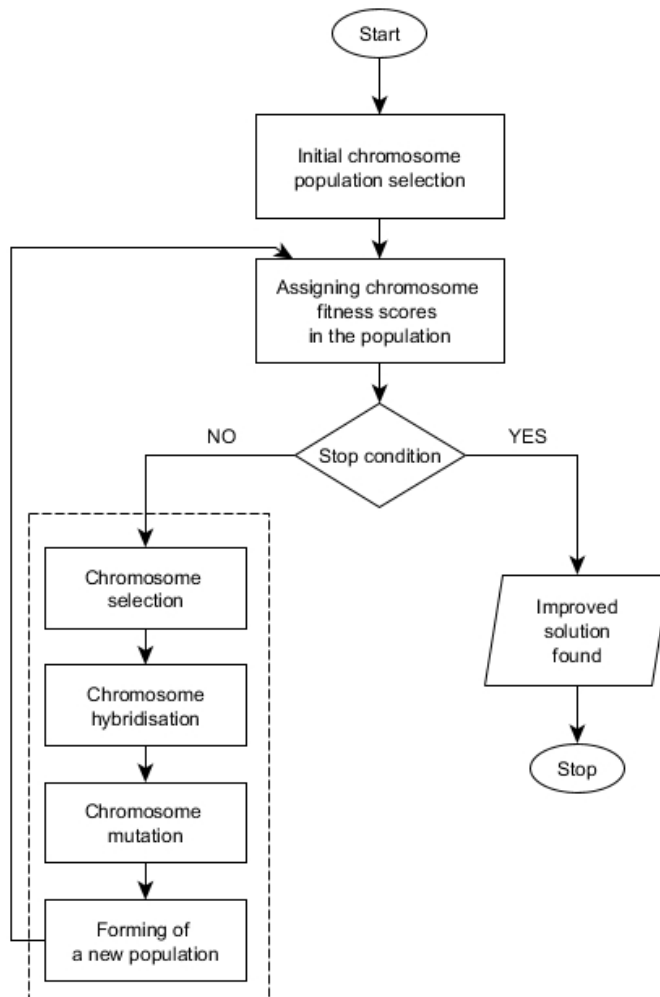
This section describes the principles adopted for the original method, and how the task was performed.

### **4.1. The principles**

It was assumed that the solution needs to enable finding the longest path in any connected, undirected and loopless graph. This subsection describes the method developed. It contains information on the principles adopted, the adopted graph presentation method, and the objective function developed. The method's functioning is also presented using a sample graph.

#### **4.1.1. The adopted graph presentation method**

Following an analysis of the available solutions, it was decided that a transition matrix-based graph presentation method [4] would be used to achieve the aim of this paper. It was decided to introduce several additional principles, which would enable the method to be simplified and better adapted to the problem to be solved.



**Fig. 1. Genetic algorithm flowchart**

One principle is to analyse only loopless graphs. The main diagonal  $A$  of the transition matrix determines the presence of edges whose source and closure is a single node. Assuming that no loops are present in the graph in question, analysing the values of elements lying on the main diagonal of the matrix is not necessary, which enables the algorithm's computational complexity to be reduced.

Another simplification stems from the properties of undirected graphs. An edge connecting two nodes in an undirected graph is interpreted as a pair of



antisymmetric arcs, which is reflected in the transition matrix. It can be observed that elements located above the main diagonal are symmetrical to elements located below the main diagonal, as shown in Figure 2.

Taking the above into account, a simplified matrix representation (derived directly from the transition matrix) can be adopted for undirected graphs, as shown in Figure 3.

$$A = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Fig. 2. Symmetry of transition matrix elements for undirected graphs

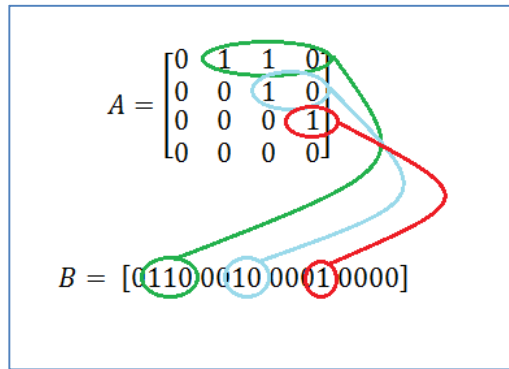
$$A = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\downarrow$$

$$B = \begin{bmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

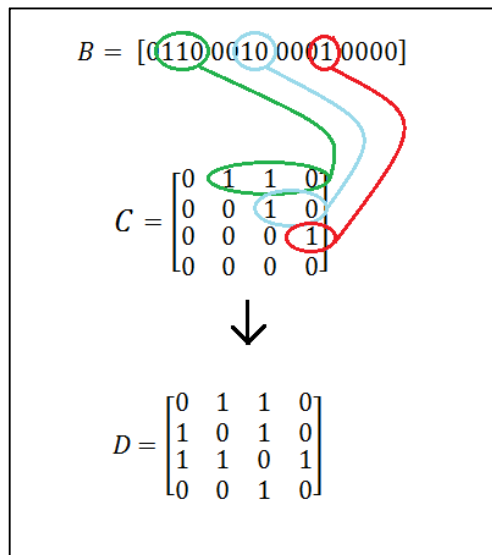
Fig. 3. Zeroing of elements located below the diagonal

Individuals that constitute the solutions for the genetic algorithm are commonly represented as bit vectors. It was therefore assumed that the graph in question will also be presented in this form. For this reason, matrix  $A$  was transformed to vector  $B$  in the manner described above. An example of transforming the transition matrix to a bit vector is shown in Figure 4.



**Fig. 4. Transformation of the transition matrix of an undirected graph to vector form**

It must be noted that the operation shown above is reversible, i.e. it enables building a simplified matrix, and then a transition matrix based on a bit vector (Figure 5). This property will be used in the original solution.



**Fig. 5. Transformation of an undirected graph vector to the matrix form**

### 4.1.2. Objective function

The objective function is a crucial element of the genetic algorithm. It is used to assign scores to the randomly selected solutions, and consequently the algorithm's effectiveness depends on its correct functioning. This function should assume the highest values for the most valuable solutions. For the

purposes of the method in question, an objective function with the following formula was developed:

$$f(x) = s_x - (k_x - s_x)$$

where:

$s_x$  – length of the longest simple path for solution  $x$ ,

$k_x$  – number of graph edges for solution  $x$ .

The values assumed by the objective function expressed with this formula are higher, the greater the  $s_x$  value is, or in other words, the greater the power of the set of nodes between which the path runs<sup>2</sup> is. Additionally, by subtracting the difference  $k_x - s_x$ , which is the difference between the number of edges in the graph and the length of the longest path, from the result, we promote those solutions for which the graph contains the lowest number of edges. Taking the above into account, it can be assumed that the function enables the correct evaluation of the selected solutions, assigning higher scores to those that contain the longest simple path with a greater number of nodes.

## 4.2. Practical application of the method

This subsection presents the way this method is used to determine the longest path for the undirected graph shown in Figure 6.

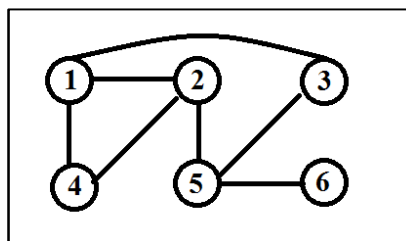


Fig. 6. Visualisation of a 6-node undirected graph

---

<sup>2</sup> Any algorithm capable of performing this task can be used to determine the set in question. It must be noted that the paper does not analyse the effects of computational complexity of the selected algorithm on the total complexity of the proposed solution.

- 1) Preparation of the node transition matrix

$$A = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

- 2) “Zeroing” of elements of matrix A, located below the diagonal

$$B = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- 3) Transformation of matrix B to the vector form

$$C = [011100 \ 000110 \ 000010 \ 000000 \ 000001 \ 000000]$$

- 4) Generating  $n$  vectors of length  $l = 36$ , constituting the first population. Population size  $n = 6$  was assumed.

$$x_1 = [011111 \ 000000 \ 010110 \ 010111 \ 010100 \ 001110]$$

$$x_2 = [000011 \ 010100 \ 011010 \ 010110 \ 000110 \ 010100]$$

$$x_3 = [001101 \ 011000 \ 010000 \ 010100 \ 100100 \ 001110]$$

$$x_4 = [000000 \ 010000 \ 011010 \ 000100 \ 110000 \ 011110]$$

$$x_5 = [011111 \ 011110 \ 110010 \ 010011 \ 011110 \ 111000]$$

$$x_6 = [010111 \ 011110 \ 000110 \ 000000 \ 010100 \ 000000]$$

- 5) Assigning scores to the resulting solutions

- a) Solution normalisation<sup>3</sup> based on vector  $B$ . The purpose of this operation is to obtain the value of 1 only at such positions in vectors  $x_1 - x_6$  at which they also occur in vector  $B$  – which ensures that an appropriate edge exists in the graph in question.

---

<sup>3</sup> Normalisation for the solution discussed means the preliminary preparation of solution vectors for further analysis by setting to zero those positions in the vectors for which no edges exist in the analysed graph.

$$\begin{aligned}
 x_{1n} &= [011100\ 000000\ 000010\ 000000\ 000000\ 000000] \\
 x_{2n} &= [000000\ 000100\ 000010\ 000000\ 000000\ 000000] \\
 x_{3n} &= [001100\ 000000\ 000000\ 000000\ 000000\ 000000] \\
 x_{4n} &= [000000\ 000000\ 000010\ 000000\ 000000\ 000000] \\
 x_{5n} &= [011100\ 000110\ 000010\ 000000\ 000000\ 000000] \\
 x_{6n} &= [010100\ 000110\ 000010\ 000000\ 000000\ 000000]
 \end{aligned}$$

b) Assigning scores to the solutions using objective function  $f_x$ .

**Tab. 1. Lengths of the longest graph paths for the randomly selected solutions**

Solution	Length of the longest path
$x_{1n}$	4
$x_{2n}$	2
$x_{3n}$	3
$x_{4n}$	2
$x_{5n}$	4
$x_{6n}$	5

Solutions  $x_{6n}$ ,  $x_{1n}$  and  $x_{5n}$ , which indicate the longest paths (among the selected solutions), received the highest scores.

- c) Selection of solutions to participate in reproduction, using one of the strategies. For the roulette strategy, solutions with the highest objective function values have the highest probability of becoming the parent pool of the next generation. In the case in question, solutions  $x_{6n}$  and  $x_{1n}$  were randomly selected to be used in reproduction twice, while  $x_{5n}$  and  $x_{3n}$  will be used once.
- d) Hybridisation of the selected solutions. A single hybridisation point was assumed, and the hybridisation point was selected using the roulette method and is  $r = 3$ . The solutions were divided into fragments at the 3rd position in the sequence.

$$\begin{aligned}
 x_{1n} &= [011100\ 000000\ 000010\ 000000\ 000000\ 000000] \\
 x_{2n} &= [011100\ 000000\ 000010\ 000000\ 000000\ 000000] \\
 x_{3n} &= [001100\ 000000\ 000000\ 000000\ 000000\ 000000] \\
 x_{4n} &= [010100\ 000110\ 000010\ 000000\ 000000\ 000000] \\
 x_{5n} &= [011100\ 000110\ 000010\ 000000\ 000000\ 000000] \\
 x_{6n} &= [010100\ 000110\ 000010\ 000000\ 000000\ 000000]
 \end{aligned}$$

Next, hybridisation pairs were randomly selected:

$$x_{1k} - x_{3k}$$

$$x_{2k} - x_{6k}$$

$$x_{4k} - x_{5k}$$

The following solutions were obtained:

$$y_1 = [011100\ 000000\ 000000\ 000000\ 000000\ 000000]$$

$$y_2 = [011100\ 000110\ 000010\ 000000\ 000000\ 000000]$$

$$y_3 = [001100\ 000000\ 000010\ 000000\ 000000\ 000000]$$

$$y_4 = [010100\ 000110\ 000010\ 000000\ 000000\ 000000]$$

$$y_5 = [011100\ 000110\ 000010\ 000000\ 000000\ 000000]$$

$$y_6 = [010100\ 000000\ 000010\ 000000\ 000000\ 000000]$$

- e) Solution mutation was performed. One solution  $y_3$  was randomly selected for mutation. Mutation was performed at three randomly selected positions:  $a = 4, b = 6$ , and  $c = 20$ .

After the mutation, vector  $y_3$  has the following form:

$$y_{3m} = [001001\ 000000\ 000010\ 010000\ 000000\ 000000]$$

- f)

$$y_1, y_2, y_{3m}, y_4, y_5, y_6$$

Repeat steps 4-7 for the resulting vectors. Repeat the steps for the assumed number of iterations or until the optimum solution is found.

- g) After the requisite number of iterations were performed, the optimum solution  $z_{opt}$  in bit vector form was found, based on which a simplified matrix, and subsequently a transition matrix were built (these operations were the reverse of steps 1-3).

$$z_{opt} = [011000 \ 000100 \ 000010 \ 000000 \ 000001 \ 000000]$$



$$C = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$



$$D = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

- h) Based on transition matrix  $D$ , the optimum solution was selected – the longest path in the analysed graph. It is shown in Figure 7.

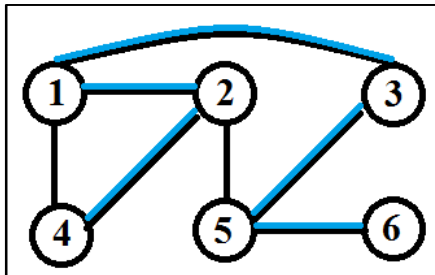


Fig. 7. Visualisation of the longest path in the analysed graph

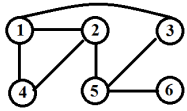
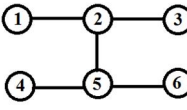
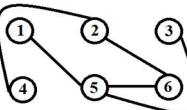
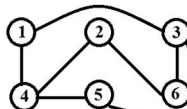
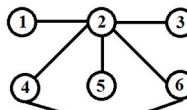
## 5. Acceptance tests

In order to confirm the effectiveness of the method developed, it was subjected to acceptance tests. The results obtained with its use were compared to results obtained by using an existing analytical algorithm – the Roberts-Flores algorithm [3].

The tests were performed on five 6-node graphs. The results are shown in Table 2.

The results confirm the effectiveness of the original method. For each test graph, the longest existing path was found. In some cases, it was not a path identical to that found by the Roberts-Flores algorithm. This stems from the fact that there may be multiple longest paths in a graph. One must remember that searching for a solution using a genetic algorithm is based on randomness, so different paths may be obtained after multiple attempts to use the method for a given graph.

**Tab. 2. Comparison of the results obtained using the original method with the results obtained using another algorithm**

No.	Analysed graph	Path found by the original method	Length of the path found by the original method	Path found by the R-F algorithm	Length of the path found by the R-F algorithm
1		4-2-1-3-5-6	6	3-1-4-2-5-6	6
2		1-2-5-4	4	1-2-5-4	4
3		4-2-6-5-1	5	1-5-6-2-4	5
4		1-3-6-2-4-5	6	1-3-6-2-4-5	6
5		1-2-4-6	4	1-2-4-6	4



## **6. Conclusion**

The method of finding the longest chain in a graph using a genetic algorithm, presented in the paper, is an alternative to existing analytical methods (such as the Roberts-Flores algorithm). Presenting the graph as a transition matrix enabled it (after performing the appropriate transformations) to be presented as a bit vector, which is the usual form of presenting specimens that constitute solutions for genetic algorithms. An objective function enabling the solutions generated to be properly assessed was also developed.

Acceptance tests performed on the original method, which involved comparing the results obtained with its use to results obtained from an analytical algorithm (Roberts-Flores), confirmed its effectiveness.

It must be emphasised that further development of the original method is possible, which will enable the algorithm's computational complexity and processing time to be reduced. This can be achieved, among other ways, by preparing a more effective objective function that enables better evaluation of the resulting solutions, which will be beneficial for the quality of subsequent populations and thus will reduce the algorithm's processing time. It is also possible to develop a different method of graph presentation, which may enable reducing the solution's computational complexity.

## **References**

- [1] GOLDBERG D., *Algorytmy genetyczne i ich zastosowania*, Warszawa 2003.
- [2] KORZAN B., *Elementy teorii grafów i sieci. Metody i zastosowania*, Warszawa 1978.

## **Online sources**

- [3] JAWORSKI J., *Algorytmy teorii grafów. Algorytm Roberta – Floresa*.  
<http://www.staff.amu.edu.pl/~jaworski/agrsyll6.pdf>.
- [4] WAŁASZEK J., *Reprezentacja grafów w komputerze*.  
[http://eduinf.waw.pl/inf/alg/001\\_search/0124.php](http://eduinf.waw.pl/inf/alg/001_search/0124.php).

## **Metoda wyznaczania najdłuższego łańcucha prostego w grafie wykorzystująca algorytm genetyczny**

STRESZCZENIE: W artykule rozpatrzono metodę wyznaczania najdłuższego łańcucha prostego w grafie wykorzystującą algorytm heurystyczny – algorytm genetyczny. Metoda umożliwia efektywne wyznaczanie najdłuższego łańcucha w dowolnym grafie spójnym, nieskierowanym, bez pętli.

SŁOWA KLUCZOWE: teoria grafów, łańcuch Hamiltona, algorytm heurystyczny, algorytm genetyczny

*Received by the editorial staff on: 26 June 2018*

# Badanie możliwości wykorzystania metody porównawczej do zapobiegania skutkom ataków typu DNS Injection

**Michał MELANIUK**

Instytut Teleinformatyki i Automatyki, WCY, WAT,  
ul. Gen. Witolda Urbanowicza 2, 00-908 Warszawa  
michal.melaniuk@wat.edu.pl

**STRESZCZENIE:** W artykule przedstawiono zagadnienie wykorzystania metody porównawczej do zapobiegania skutkom ataków typu DNS Injection. Zaproponowano rozwiązanie wykorzystujące przedmiotową metodę oraz opisano opracowane środowisko laboratoryjne do eksperymentalnego zbadania skuteczności działania opracowanego rozwiązania.

**SŁOWA KLUCZOWE:** metoda MM, metoda prób porównawczych, diagnostyka, serwer DNS, DNS Injection

## 1. Wprowadzenie

Jedną z najczęściej wykorzystywanych usług dostępnych w sieci Internet jest usługa Systemu Nazw Domenowych (DNS, ang. *Domain Name System*). Dzięki wspomnianej usłudze można przede wszystkim nawiązać połączenie z wybraną stroną internetową, wykorzystując jej nazwę mnemoniczną (zazwyczaj łatwiejszą do zapamiętania) zamiast jej adresu IP. W większości przypadków przekształcenia nazw na adresy IP (lub odwrotnie) dokonywane są przez osobny węzeł sieci nazywany serwerem DNS, jednakże zdarza się, że taki serwer pracuje również lokalnie na danym hoście. Użytkownik systemu DNS nie ma bezpośredniej kontroli nad serwerem, co wiąże się z ryzykiem uzyskania od serwera niepoprawnego odwzorowania nazwy. Niestety, praktyka pokazuje, że brakuje uniwersalnego sposobu, za pomocą którego użytkownik mógłby upewnić

się, że odpowiedzi otrzymane od serwerów DNS są wiarygodne. Brak wpisów w bazie rekordów DNS może spowodować brak dostępu do zasobów konkretnej sieci, natomiast błędne wpisy mogą przekierować ruch sieciowy do wskazanej przez atakującego, niepoprawnej lokalizacji będącej pod jego kontrolą. Można zatem stwierdzić, że poprawne działanie systemu DNS jest krytyczne dla sprawnego działania i bezpieczeństwa sieci Internet [4]. Postanowiono zbadać możliwość opracowania sposobu minimalizacji lub całkowitego zapobiegania skutkom ataków względem serwerów DNS. Jako podstawę wybrano metody diagnostyki systemowej. Niniejsze opracowanie stanowi materiał wstępny do porównania skuteczności zastosowania różnych metod diagnostyki systemowej w przedmiotowym obszarze. W pracy skupiono się na metodzie prób porównawczych w celu zbadania jakości jej zastosowania w wykrywaniu skutków ataków typu *DNS Injection*.

W niniejszym artykule skupiono się na ochronie użytkownika sieci Internet przed skutkami ataków typu *DNS Injection* – polegających na modyfikacji wpisów w tablicach odwzorowań serwera DNS [14]. Każdy użytkownik sieci Internet, który będzie w stanie znaleźć i wykorzystać luki bezpieczeństwa w oprogramowaniu serwera DNS, bądź wysłać do niego odpowiednio sfałszowane aktualizacje wpisów w tablicach odwzorowań, może być atakującym. Opisywana metoda ma na celu wykrywanie skutków ataku *DNS Injection*, a nie ochrony przed nim, jak to czynią tradycyjne systemy bezpieczeństwa (takie jak IDS/IPS lub zapory sieciowe). W obszarze zagrożeń związanych z atakami typu *DNS Injection* znaleźć można prace, m.in. [5], [20], [22], jednakże prezentują one zupełnie odmienne podejście do tematu ochrony użytkownika przed tego typu zagrożeniami niż omawiane w niniejszym opracowaniu. W pracy [5] autorzy przeprowadzili badania serwerów DNS z różnych obszarów sieci Internet pod kątem oceny ich podatności na ataki *DNS Cache Injection*. Na podstawie przeprowadzonych prac badacze stwierdzili, że ponad 90% z testowanych serwerów DNS było podatnych na realizowane ataki, co oznacza, że ich konfiguracja może być ukradkiem regularnie modyfikowana. Prezentowane metody nadużyć dodatkowo nie wymagają istotnych nakładów pracy podczas realizacji ataków. W pracy [20] opisano system REMeDy, który w sposób automatyczny ocenia spójność odpowiedzi serwerów DNS w danej sieci. Prezentowany system poprzez pasywne nasłuchiwanie ruchu sieciowego pozwala wykryć dodane do sieci węzły, które kierują ruch sieciowy do niepoprawnych lokalizacji lub zainfekowane serwery DNS. System bazuje na wskazaniach serwerów DNS dostawcy usług internetowych (ang. *Internet Service Provider – ISP*) i zakłada, że są one zawsze poprawne. Opracowanie [22] prezentuje rozwiązanie T-DNS – system DNS oparty na protokole TCP. Zastosowanie komunikacji połączeniowej pozwoliło na ograniczenie: możliwości podszywania się pod inne serwery DNS, realizacji ataków DoS typu

*Amplification*. Dodatkowo prezentowany system wykorzystuje szyfrowanie transmisji przy użyciu mechanizmu TLS.

## 2. Powiązane prace

Opracowana *metoda ochrony użytkownika sieci Internet przed skutkami ataków typu DNS Injection*<sup>1</sup> zakłada wykorzystanie metody prób porównawczych określanej mianem metody MM<sup>2</sup> [9]. W literaturze znaleźć można prace, w których wykorzystywano wspomnianą metodę najczęściej do diagnozowania sieci procesorów (o różnej strukturze logicznej). A. Arciuch w pracy [1] zaprezentował techniczne aspekty diagnozowania sieci procesorów o łagodnej degradacji metodą MM. Autor w swojej pracy zaimplementował wspomnianą metodę w fizycznej sieci zbudowanej z wykorzystaniem mikrokomputerów Micro2440. Opracowana sieć pracuje w określony sposób, przeplatając seanse diagnostyczne (w których następuje weryfikacja poprawnej pracy wszystkich węzłów sieci) okresami roboczymi (w których sieć wykonuje swoje zadania). R. Kulesza i Z. Zieliński w pracy [7] wykorzystali wspomnianą metodę do określania wnikliwości diagnostycznej sieci procesorów. Autorzy w swoim opracowaniu określili reguły, które pozwalają jednolite definiowanie klas sieci procesorów o określonych własnościach diagnostycznych. Na podstawie opracowanych reguł możliwe jest wytworzenie automatu programowego, który będzie określał własności diagnostyczne sieci, bazując na wzorcach syndromów. A. Sengupta i A.T. Dahbura w [13] zaproponowali wykorzystanie metody MM w samo-diagnostującym się systemie z wykorzystaniem modelu MM we wspomnianym systemie. W pracy podany został zestaw kryteriów określających, czy wadliwe procesory systemu mogą zostać wykryte na podstawie przeprowadzonych porównań wykonywanych przez nie zadań. Ponadto w pracy zaproponowany został algorytm wielomianowy realizujący zadanie identyfikacji uszkodzonych jednostek. G.Y. Chang, G.H. Chen i G.J. Chang w pracy [2] wykorzystali model MM<sup>\*3</sup> do opracowania sekwencyjnego diagnozowania sieci procesorów. W podejściu tym wymagane jest określenie odpowiedniej liczby procesorów niezdatnych i ich naprawienie w każdej iteracji algorytmu. W pracy opisano algorytm, który zastosowano do diagnostyki m.in. sieci typu

---

<sup>1</sup> W celu uproszczenia w dalszej części artykułu określana w skrócie jako *Metoda*.

<sup>2</sup> Nazwa metody pochodzi od nazwisk twórców: M. Malek oraz J. Maeng.

<sup>3</sup> Model MM\* charakteryzuje się wykorzystaniem struktur diagnostycznych składających się ze wszystkich możliwych prób porównawczych, podczas gdy model MM wykorzystuje minimalną liczbę prób porównawczych, która zapewnia detekcję  $t$  uszkodzonych węzłów sieci.

hipersześcian. Ponadto dostępnych jest wiele interesujących prac opisujących różne zastosowania metod diagnostyki systemowej, m.in.: [3], [9], [11], [12]. Jianxi Fan w [3] badał możliwość zastosowania metody prób porównawczych do wykrywania niezdatnych węzłów w sieci o strukturze pochodnej hipersześcianu określonej mianem *crossed cube*. Autor wykazał, że taka  $n$ -wymiarowa struktura jest  $n$ -diagnostowalna dla  $n \geq 4$ , podczas gdy dla hipersześcianu ta zależność zachodzi dla  $n \geq 5$ . W pracy [9] autorzy rozważali wykorzystanie diagnostyki adaptacyjnej, która dąży do zmniejszenia liczby rund testowych i samych testów, za pomocą modelu MM dla struktur typu hipersześcian  $n$ -wymiarowy. Wykazana została możliwość diagnozowania  $n$ -wymiarowych hipersześcianów w opisanym sposobie w ograniczonej liczbie testów. Przykładowo, dla  $n \geq 5$  przedstawiona metoda potrzebuje co najwyżej 6 rund testowych i  $2^n + 2n^3 + 8n^2$  testów do wykrycia  $n$  uszkodzonych węzłów. Jiarong Liang i Qian Zhang w pracy [11] opracowali metodę  $t/s$ -diagnozowania sieci o strukturze hipersześcianu z wykorzystaniem m.in. metody prób porównawczych. W pracy zaprezentowano teoretyczne zagadnienia związane z  $t/s$ -diagnozowalnością oraz przedstawiono algorytm izolowania uszkodzonych węzłów w podzbiory. Natomiast w pracy [12] autorzy rozważają zagadnienie warunkowej diagnostowalności systemu, w którym pojawiają się uszkodzone węzły. W pracy skupiono się na strukturze sieci określanej mianem *Split-Star*, która diagnozowana jest z wykorzystaniem modelu PMC<sup>4</sup>.

W niniejszej pracy postanowiono wykorzystać inne podejście i zastosować metodę prób porównawczych w diagnozowaniu poprawności działania serwerów DNS.

### 3. Proponowane rozwiązanie

Niniejszy punkt powstał w oparciu o [6] i [8]. Metoda MM wykorzystuje graf porównań jako jeden ze sposobów reprezentacji struktury logicznej węzłów wraz z odpowiadającym jej zbiorem prób porównawczych. Pojęcie grafu porównań (wraz z przykładami) zostało wyjaśnione w dalszej części niniejszego artykułu. W omawianym obszarze problemu, który stanowi wzajemne testowanie się serwerów DNS, jako elementarny test porównawczy rozumiane jest wysłanie przez komparator identycznego zapytania rozwinięcia nazwy domenowej do obu węzłów stanowiących parę porównawczą. Następnie zadaniem komparatora jest zweryfikowanie, że uzyskane adresy IP (odpowiedzi na wysłane zapytania) są identyczne. Opisany rodzaj sprawdzeń będzie wykonywany okresowo, co

---

<sup>4</sup> Dodatkowe referencje związane z modelem PMC zawarto w punkcie 5.

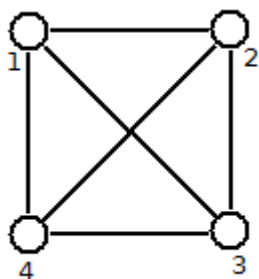
$k^5$  zapytań zapewniając ciągłą niezawodność serwerów DNS bez nadmiernego obciążania sieci komputerowej.

Jako węzeł *zdatny* rozumiany jest serwer DNS, który w sposób poprawny odwzorowuje nazwę domenową na adres IP (zwracany adres IP prowadzi do właściwej lokalizacji). Węzeł *niezdatny* jest to serwer DNS, którego odwzorowania nazw na adresy IP są niepoprawne (zwracany adres IP prowadzi do niewłaściwej lokalizacji, np. kontrolowanej przez agresora). *Uszkodzenie* jest to zainfekowanie serwera DNS, w wyniku którego zwraca on niewłaściwe adresy IP.

Zdatny komparator zaopiniuje, że para porównawcza jest zdatna (wynik testu porównawczego przyjmie wartość 0), jeżeli wyniki złożonych zapytań DNS są identyczne. Różne wyniki testu skutkują zaopiniowaniem o niezdatności pary porównawczej (wynik testu porównawczego przyjmie wartość 1), przy czym niezdatny jest przynajmniej jeden węzeł z pary porównawczej (nie jest wskazane który). Opinia wyrażona przez zdatny komparator jest więc zgodna ze stanem faktycznym. Niezdatny komparator (w założeniach metody MM) wyraża opinię, która jest przypadkowa i przyjmuje wartość 0 lub 1.

### 3.1. Istotne cechy struktury porównawczej typu MM

Rozpatrzmy przykładową strukturę logiczną sieci opisaną spójnym grafem zwykłym  $G = \langle E, U \rangle$ . Przykładowy graf przedstawiono na rysunku 1.



Rys. 1. Przykładowy graf reprezentujący strukturę logiczną

Strukturze logicznej  $G$  odpowiada zbiór wszystkich prób porównawczych oznaczony poprzez  $\Psi(G)$ , a pojedynczą próbę porównawczą oznaczono symbolem  $\psi \in \Psi'$ ,  $\Psi' \subseteq \Psi(G)$ . Dla próby porównawczej  $\psi$  istnieje zbiór komparatorów oznaczany jako  $K(\psi)$  oraz zbiór par porównawczych oznaczony

<sup>5</sup> Wartość parametru  $k$  jest dowolna.

jako  $P(\psi)$ . Zbiór węzłów biorących udział w próbie porównawczej  $\psi$  oznaczono jako  $E(\psi)$ .

W próbie porównawczej węzeł  $e_k \in E(G)$  będący komparatorem zleca parze porównawczej  $\{e_i, e_j\} \subset E(G)$  jednakowe zadanie, a następnie sprawdza, czy uzyskane wyniki są identyczne.

Próbie porównawczą oznacza się poprzez  $(e_k; e_i, e_j)$ . Wynik próby porównawczej  $d((e_k; e_i, e_j))$  może przyjąć jedną z trzech wartości (zależną od stanów niezawodnościowych komparatora i pary porównawczej):

$$d((e_k; e_i, e_j)) = \begin{cases} 0 \text{ dla } [e_k = 0 \wedge r(e_i|e_k) = r(e_j|e_k)] & \text{przyp. a)} \\ 1 \text{ dla } [e_k = 0 \wedge r(e_i|e_k) \neq r(e_j|e_k)] & \text{przyp. b)} \\ x \in \{0,1\} \text{ dla } e_k = 1 & \text{przyp. c)} \end{cases} \quad (1)$$

gdzie  $n(e_k)$  jest funkcjonalną niezawodnością węzła  $e_k$  oraz  $r(e|e_k)$  jest wynikiem zadania zleconego przez węzeł  $e_k$ , a wykonanego przez węzeł  $e$ .

W rozważanej dziedzinie problemu (wzajemne testowanie się serwerów DNS) w *przyp. c)* niezdatność serwera DNS będącego komparatorem nie ma wpływu na wynik realizowanego przez niego testu. Podczas porównania komparator weryfikuje wzajemną zgodność wyników otrzymanych od węzłów pary porównawczej. Wyniki te nie są zestawiane z wpisami posiadanymi przez dany serwer DNS, dlatego nawet jeśli posiadałby on skompromitowane (w wyniku ataku *DNS Injection*) odwzorowania nazw w swojej bazie, nie wpłyną one na poprawność opiniowania. Interpretacja wyników próby porównawczej została przedstawiona w tabeli 1. Natomiast *przyp. a)* i *przyp. b)* mają zastosowanie zgodnie z powyższym wzorem.

**Tab. 1. Interpretacja wyników próby porównawczej w opracowanej Metodzie**

$n(e_k)$	$n(e_i)$	$n(e_j)$	$d((e_k; e_i, e_j))$
1	0	0	0
	0	1	1
	1	0	1
	1	1	1



**Określenie 1.** [16] Sieć komputerów opisana strukturą  $G$  określa się jako *jednokroково  $t$ -diagnostowalną*<sup>6</sup> za pomocą zbioru prób porównawczych  $\Psi' \subseteq \Psi(G)$ , jeżeli każda para zbiorów  $E'$  i  $E''$  niezdatnych węzłów takich, że  $|E'| \leq t$  i  $|E''| \leq t$ , jest rozróżnialna za pomocą choć jednej próby porównawczej  $\psi \in \Psi'$ .

**Określenie 2.** [16] *Grafem porównań* sieci komputerów o strukturze  $G$  dla zbiorów prób porównawczych  $\Psi' \subseteq \Psi(G)$  nazywamy taki graf zwykły  $\hat{G}(G, \Psi') = \langle E(G), U(G, \Psi') \rangle$  o opisanych krawędziach, że  $[(e', e'') \in U(G, \Psi')] \leftrightarrow [\exists \psi \in \Psi': P(\psi) = \{e', e''\}]$ , gdzie etykietą krawędzi  $(e', e'')$  jest  $K(\psi)$ .

**Własność 1.** [7], [9] Warunkiem koniecznym, aby graf  $G$  był  $t$ -diagnostowalny za pomocą zbioru prób porównawczych  $\Psi' \subseteq \Psi(G)$  jest spełnienie zależności:

$$(|E(G)| \geq \max\{t + 3, 2 \cdot t + 1\}) \wedge (\forall_{e \in E(G)}: \mu(e) \geq t), \quad (2)$$

gdzie  $\mu(e)$  oznacza stopień wejściowy węzła  $e$ .

**Własność 2.** [7] Graf  $G$  jest  $t$ -diagnostowalny za pomocą prób porównawczych  $\Psi' \subseteq \Psi(G)$ , wtedy i tylko wtedy, gdy dla każdej pary podzbiorów węzłów  $E_1, E_2 \subseteq E(G)$  takich, że  $E_1 \neq E_2$  oraz  $|E_1| = |E_2| = t$ , spełniony jest jeden z poniższych warunków:

$$\begin{aligned} \text{a) } & \exists_{\psi', \psi'' \in \Psi(G)}: [(\{K(\psi'), K(\psi'')\} \cap \{E_1 \cup E_2\} = \emptyset) \wedge \\ & \wedge ( [|P(\psi') \cap \{E_1 \setminus E_2\}| = 1] \vee [|P(\psi'') \cap \{E_2 \setminus E_1\}| = 1] ) ] \end{aligned} \quad (3)$$

$$\text{b) } \exists_{\psi' \in \Psi(G)}: [|P(\psi') \cap \{E_1 \setminus E_2\}| = 2] \wedge [K(\psi') \cap \{E_1 \cup E_2\} = \emptyset] \quad (4)$$

$$\text{c) } \exists_{\psi' \in \Psi(G)}: [|P(\psi') \cap \{E_2 \setminus E_1\}| = 2] \wedge [K(\psi') \cap \{E_1 \cup E_2\} = \emptyset] \quad (5)$$

### 3.2. Metoda identyfikowania niezdatnych serwerów

Wyniki testów porównawczych przeprowadzonych w ramach jednej sesji diagnostycznej utworzą tzw. *syndrom globalny*. Każdy serwer posiada w swoich zasobach wartości wzorcowe określające niezawodność poszczególnych węzłów w wykorzystywanej strukturze. Wspomniane wartości wzorcowe są różne dla różnych struktur porównawczych i są określane mianem *wzorca syndromów*.

<sup>6</sup> Jako  $t$ -diagnostowalność rozumiana jest możliwość wskazania maksymalnie  $t$  uszkodzonych węzłów w sieci. Przykładowo struktura 3-diagnostowalna umożliwia wskazanie maksymalnie trzech uszkodzonych węzłów. Opisano to dokładnie w dalszej części artykułu.

Przykład wzorca syndromów dla struktury diagnostycznej zaprezentowanej w dalszej części artykułu na rysunku 3 przedstawiono w tabeli 2. Pojedyncza wartość (wiersz w tabeli 2) jest często określana mianem *syndromu wzorcowego*<sup>7</sup>.

Tab. 2. Przykładowy wzorzec syndromów

$i$					1	2	3	4	5	6	7	8
$K(\psi_i)$					1	1	2	2	3	3	4	4
$P(\psi_i)$					2	2	3	3	4	4	1	1
					4	3	1	4	2	1	3	2
$e$	1	2	3	4	$d(\psi_i)$							
$n(e)$												
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	1	1	1	0	0
0	0	1	0	0	0	1	1	1	0	0	1	0
0	1	0	0	0	1	1	0	0	1	0	0	1
1	0	0	0	0	0	0	1	0	0	1	1	1

Każdy serwer DNS po uzyskaniu wyników poszczególnych prób porównawczych od innych węzłów tworzy syndrom globalny (zawierający wyniki wszystkich przeprowadzonych prób porównawczych). Następnie próbuje dopasować otrzymany syndrom do jednego z syndromów wzorcowych w posiadanym wzorcu syndromów dla danej struktury diagnostycznej. Uzyskanie dopasowania pozwala określić stany niezawodnościowe serwerów DNS biorących udział w sesji diagnostycznej.

### 3.3. Wymagania względem opracowanej *Metody*

Na podstawie analizy wyników ataków względem serwerów DNS, które zrealizowano m.in w pracy [14] można stwierdzić, że istotne jest zapewnienie integralności zapisów w bazie odwzorowań serwera DNS. Nieautoryzowana zmiana choćby jednego rekordu we wspomnianej bazie odwzorowań serwera stanowi zagrożenie dla użytkowników, którzy się z nim komunikują i może

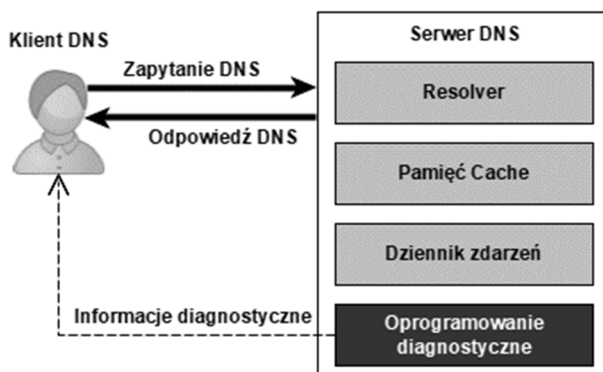
<sup>7</sup> Pojęcia: *syndrom globalny*, *syndrom wzorcowy* i *wzorzec syndromów* zostały dokładnie zdefiniowane w pracy [6].

skutkować przekierowaniem ruchu sieciowego do nieprawidłowej (często kontrolowanej przez agresora) lokalizacji.

Założono, że opracowana *Metoda* ma umożliwiać wykrycie określonej liczby skompromitowanych serwerów DNS w środowisku sieciowym (wspomniana liczba jest zdefiniowana jako  $t$ ). Mechanizm działania polega na wzajemnym testowaniu się serwerów DNS poprzez wysłanie odpowiedzi na zapytanie DNS. Liczba wymaganych porównań zależy od liczby uszkodzonych węzłów do wykrycia. Zebrane odpowiedzi zostaną porównane, co pozwoli określić, które z nich są błędne i pośrednio, aby umożliwić wskazanie niezdatnych serwerów DNS.

Artykuł koncentruje się na ochronie użytkownika przed omawianym typem ataków i zapobieganiu jego skutkom. Wyniki działania *Metody* wysyłane do komputera klienckiego pozwolą mu używać tylko tych serwerów DNS, które zostały zidentyfikowane jako zdatne. Zakłada się, że opracowana *Metoda* będzie w stanie wykryć serwery DNS zaatakowane przez agresora z wykorzystaniem DNS Injection.

Oprogramowanie diagnostyczne, które wykorzystuje opracowaną *Metodę*, rozszerza architekturę serwera DNS. Działając w tle, regularnie sprawdza zdatność serwerów DNS. Ponadto na żądanie informuje klienta DNS o wynikach testów. Uproszczony schemat oprogramowania do diagnostyki serwerów DNS przedstawiono na rysunku 2.



Rys. 2. Uproszczony schemat oprogramowania diagnostycznego

Oprogramowanie diagnostyczne wykonuje trzy główne zadania:

- wysyłanie zapytań DNS dla wskazanej nazwy domenowej i odbieranie odpowiedzi,
- grupowanie odpowiedzi i na ich podstawie określenie niezawodności serwerów DNS uczestniczących w teście,
- przekazanie klientowi informacji o zdatności serwerów DNS.

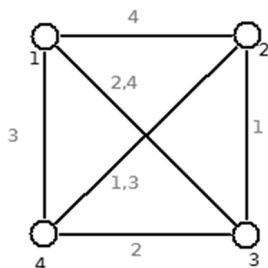
### 3.4. Opis opracowanej *Metody*

Proponowana w ramach niniejszego artykułu *Metoda* spełnia wymagania przytoczone w punktach 3.2-3.3. Ponadto spełnione są następujące założenia:

- porównanie odpowiedzi na zapytania DNS pochodzące z dwóch serwerów DNS rozumiane jest jako *test porównawczy*,
- w teście porównawczym biorą udział trzy serwery DNS: komparator (oznaczany jako  $K(\psi_i)$ ) oraz dwa serwery stanowiące parę porównawczą (oznaczane jako  $P(\psi_i)$ ).

Logiczną strukturę sieci testowanych węzłów opisać można spójnym grafem zwykłym  $G = \langle E, U \rangle$ . Opracowana *Metoda* bazuje na  $t$ -diagnozowalnym (za pomocą zbioru prób porównawczych  $\Psi' \subseteq \Psi(G)$ ) grafie porównań  $\hat{G}(G, \Psi')$ , który spełniać musi warunki konieczne i wystarczające dla metody MM (zależności (2)-(5) przedstawione w punkcie 3.1). Te zależności gwarantują, że graf porównań jest odpowiednią strukturą diagnostyczną. Poza odpowiednią liczbą węzłów biorących udział w porównaniu i odpowiednią liczbą porównań (wymuszoną przez właściwość 1 opisaną w punkcie 3.1), wspomniane porównania muszą angażować odpowiednie węzły do określenia zdolności serwerów DNS (co wymuszone przez właściwość 2 opisaną w punkcie 3.1).

Dla przykładowej struktury logicznej zaprezentowanej wcześniej na rysunku 1 można zaproponować strukturę porównawczą daną grafem porównań  $\hat{G}(G, \Psi')$  zaprezentowanym na rysunku 3.



Rys. 3. Graf porównań  $\hat{G}(G, \Psi')$  odpowiadający optymalnej strukturze diagnostowania  $\langle G, \Psi' \rangle$

Węzeł w grafie  $\hat{G}(G, \Psi')$  odpowiada serwerowi DNS. Ze zbioru prób porównawczych  $\Psi'$  wyznaczane są poszczególne próby porównawcze  $\psi_i$  ( $i \in \{1, 2, \dots, |\Psi'|\}$ ). Dla każdej wyznaczonej próby porównawczej realizowane są wymienione poniżej operacje.

1. Serwer DNS będący komparatorem próby  $i$  wysyła do pary porównawczej zapytanie DNS o rozwinięcie nazwy hosta, np. *host1.mm.pl*.

2. Serwery pary porównawczej odpowiadają adresem IP hosta, który posiadają zapisany w swoich bazach odwzorowań.
3. Komparator porównuje otrzymane adresy IP zgodnie z zależnością (1), a otrzymany wynik porównania zapisuje w syndromie globalnym i przekazuje do pozostałych serwerów DNS w strukturze.

Następnie na podstawie zebranych wyników ze wszystkich prób porównawczych w sesji diagnostycznej serwery DNS identyfikują niezdatne węzły zgodnie z metodą opisaną w punkcie 3.2. Użytkownik końcowy może pozyskać listę niezdatnych serwerów DNS, które tworzą tzw. *czarną listę serwerów DNS*. Użytkownik końcowy nie powinien korzystać z usługi DNS serwerów znajdujących się na *czarnej liście*. Wynikowo użytkownik korzysta jedynie z serwerów zdiagnozowanych jako zdatne, czyli takie, którym można zaufać.

## 4. Badanie skuteczności opracowanej metody

### 4.1. Opis środowiska laboratoryjnego

W celu praktycznej weryfikacji skuteczności opracowanej *Metody* zbudowano odpowiednie środowisko laboratoryjne. Wykorzystano rozwiązania wirtualizacji, które pozwalają na wygodne i szybkie zarządzanie serwerami DNS pracującymi w środowisku. Na chwilę obecną środowisku działa 7 serwerów DNS, co pozwala na wykorzystywanie maksymalnie 3-diagnozowalnej struktury porównawczej.

Wszystkie serwery zostały zainstalowane jako maszyny wirtualne w ramach wirtualizatora VMware ESXi 6.5.0 i pracują pod kontrolą systemu operacyjnego Linux Debian 4.9. Maszynom wirtualnym przydzielono po jednym wirtualnym procesorze, 400 MB pamięci operacyjnej oraz 2 GB przestrzeni dyskowej. Usługa serwera DNS realizowana jest za pomocą oprogramowania BIND9.

Odnośnie do konfiguracji DNS przyjęto, że każdy serwer będzie pełnił funkcję tzw. *mastera* i obsługiwał domenę *mm.pl*. Istotny z punktu widzenia badań przydatności opracowanej *Metody* będzie wpis w konfiguracji DNS dotyczący hosta *host1.mm.pl*, którego poprawny adres IP to *192.168.206.201*.

#### 4.1.1. Wykorzystywane struktury porównawcze

W literaturze przedstawione są prace opisujące algorytmy i sposoby generowania odpowiednich struktur  $t$ -diagnozowalnych zarówno dla wykorzystywanej metody porównawczej, jak i innych metod diagnostyki systemowej. Przykładowo Ł. Strzelecki w swoich publikacjach opisuje możliwości wykorzystywania do realizacji przedmiotowego zadania metodę adaptacyjną [15], [19] lub algorytm genetyczny [17], [18]. Na potrzeby opracowanego środowiska laboratoryjnego wykorzystano struktury 1-, 2- i 3-diagnozowalne wygenerowane z wykorzystaniem algorytmu opisanego w pracy [15].

Każdy serwer DNS w środowisku laboratoryjnym przechowuje w swoich zasobach wygenerowane struktury porównawcze w formie tabeli, analogicznej do przedstawionej tabeli 2. Struktury są wykorzystywane przy określaniu konkretnych testów do realizacji przez dany serwer.

#### 4.1.2. Opracowane oprogramowanie

Na potrzeby środowiska laboratoryjnego opracowano programy realizujące następujące operacje:

- infekowanie serwerów DNS,
- wielokrotne powtarzanie sesji diagnostycznych,
- testowanie zdatności środowiska laboratoryjnego opracowaną *Metodą*.

Oprogramowanie infekujące ma za zadanie podmienić konfigurację serwera DNS na niepoprawną. Zmianie podlega adres IP dotyczący hosta *host1.mm.pl*. Oprogramowanie generuje w sposób pseudolosowy błędny adres IP, który zostanie umieszczony w konfiguracji serwera w momencie infekcji. Dodatkowo oprogramowanie wykorzystywane jest do naprawiania serwerów DNS po skończonym teście poprzez podmianę niepoprawnego adresu IP na właściwy dla hosta.

Oprogramowanie testujące ma za zadanie wskazać niezdatne serwery DNS za pomocą opracowanej *Metody*. Następnie uzyskane wskazanie jest porównywane z rzeczywistym stanem sieci serwerów i na tej podstawie określana jest poprawność działania *Metody*. Dodatkowo oprogramowanie testujące pozwala na realizację badań w dwóch wariantach:

- infekowanie kilku serwerów DNS tym samym błędnym adresem IP,
- infekowanie kilku serwerów DNS różnymi błędnymi adresami IP.

## 4.2. Wyniki zrealizowanych badań

W środowisku laboratoryjnym opisanym w punkcie 4.1 przeprowadzono badania skuteczności opracowanej *Metody* w wykrywaniu skompromitowanych serwerów DNS. Procedura badań była następująca:

- 1) przeprowadzenie badań dla 1-, 2- i 3-diagnostycznych struktur porównawczych – serwery były infekowane różnymi niepoprawnymi adresami IP,
- 2) przeprowadzenie badań dla 1-, 2- i 3-diagnostycznych struktur porównawczych – serwery były infekowane identycznymi niepoprawnymi adresami IP.

Powyższa procedura została zrealizowana 1000 razy, by na podstawie uzyskanych wyników możliwe było wyciągnięcie miarodajnych wniosków. Podczas badań zbierano następujące statystyki: skuteczność wskazania niezdatnych serwerów oraz średni, minimalny i maksymalny czas trwania pojedynczej sesji diagnostycznej.

Poniżej w tabeli 3 przedstawiono zestawienie uzyskanych wyników. Rozpatrywano łącznie 5 różnych przypadków:

- P1 – struktura 1-diagnozowalna (w tym przypadku nie ma rozróżnienia na infekowanie tymi samymi lub różnymi adresami IP, bo infekowany jest tylko jeden węzeł),
- P2 – struktura 2-diagozowalna, każdy serwer infekowany innym adresem IP,
- P3 – struktura 2-diagozowalna, każdy serwer infekowany tym samym adresem IP,
- P4 – struktura 3-diagozowalna, każdy serwer infekowany innym adresem IP,
- P5 – struktura 3-diagozowalna, każdy serwer infekowany tym samym adresem IP.

Po każdej iteracji serwery DNS są przywracane do pierwotnego (zdatnego) stanu. Przy dużym obciążeniu środowiska (wykonywanie testów co kilka sekund) zdarzały się przypadki, że jeden z serwerów w teście  $n$  posiadał jeszcze ustawienia z testu  $n-1$  – serwer nie zdążył wczytać aktualnej konfiguracji zanim został odpytany o adres IP w teście  $n$ . W takich przypadkach otrzymywano błędne wyniki, które nie były brane pod uwagę w określaniu skuteczności *Metody*.

Tab. 3. Wyniki przeprowadzonych pomiarów

Przypadek	Skuteczność [%]	Czas średni [s]	Czas minimalny [s]	Czas maksymalny [s]
P1	100	0,817	0,724	7,522
P2	100	1,581	1,394	12,534
P3	0	1,531	1,394	12,534
P4	100	2,945	2,767	8,053
P5	0	4,235	2,665	16,916

### 4.3. Wnioski

Na podstawie wyników badań przedstawionych w punkcie 4.2 można stwierdzić, że zaproponowana metoda jest w stanie zidentyfikować niezdadne serwery w przypadku, gdy zostaną zainfekowane różnymi błędnymi adresami IP lub gdy mamy do czynienia z jednym infekowanym serwerem. W opisywanych przypadkach (P1, P2 i P4) skuteczność metody wyniosła 100%. Metoda bez problemu jest w stanie wskazać zaatakowane serwery, gdy te odpowiadają różnymi błędnymi adresami IP.

Jednocześnie można stwierdzić, że w przypadkach zainfekowania kilku serwerów tym samym błędnym adresem IP (przypadki P3 i P5) proponowana metoda nie jest w stanie wskazać niezdadnych serwerów. Zaistniała sytuacja wynika z faktu, że przy porównywaniu odpowiedzi od dwóch zainfekowanych serwerów, oba uzyskane adresy IP są takie same (ale oba przekierują użytkownika do niepoprawnej lokalizacji kontrolowanej przez agresora), przez co wynik porównania wskazuje brak różnic i metoda stwierdza, że para porównawcza jest zdadna (niezgodnie ze stanem faktycznym).

Bardzo ważną obserwacją jest fakt, że w przypadku infekcji kilku serwerów tym samym błędnym adresem IP, wynikiem metody jest informacja o braku dopasowania syndromu globalnego do wzorca. Oznacza to, że metoda nie jest w stanie wskazać uszkodzonych serwerów i nie zwraca żadnych wyników. W związku z tym nie występuje w proponowanej metodzie błąd polegający na wskazaniu w rzeczywistości zdadnych serwerów jako niezdadnych.

Dodatkowo można zaobserwować, że skuteczność metody nie zależy od *t*-diagnozowalności struktury. W związku z tym zwiększanie jej złożoności jest zasadne tylko w przypadku, gdy chcemy mieć możliwość wykrycia większej liczby serwerów. Opisywana sytuacja może mieć miejsce w przypadku, gdy jakaś organizacja uzna (np. na podstawie analizy ryzyka), że wymagane jest, aby mieć



możliwość wykrycia większej liczby serwerów – ryzyko infekcji np. 5 serwerów jest nieakceptowalne i fakt takiego nadużycia powinien być możliwy do wykrycia.

Ponieważ metoda porównawcza nie sprawdziła się w każdym z analizowanych przypadków, dobrym pomysłem wydaje się sprawdzenie przydatności innych metod diagnostyki systemowej. W ramach dalszych prac planuje się rozbudowanie środowiska laboratoryjnego o możliwość wykorzystania np. metody PMC [6], [9], [10], [21] lub BGM [6], [10]. W podanej literaturze zostały szczegółowo opisane własności metod PMC i BGM oraz występujące między nimi zależności i sposoby zastosowania w praktyce.

Odnosnie do czasów realizacji sesji diagnostycznych, zaobserwować można ich liniowy wzrost wraz ze wzrostem  $t$ -diagnozowalności struktury, co jest zachowaniem poprawnym. Obserwowana sytuacja pozwala wnioskować, że dla wyższych  $t$  czas realizacji pojedynczego testu będzie rósł proporcjonalnie do złożoności struktury. Zasadne jest natomiast podjęcie działań, których celem będzie optymalizacja pracy algorytmu i w efekcie skrócenie czasu wykonania pojedynczej sesji diagnostycznej.

## **5. Podsumowanie**

Celem artykułu było przedstawienie środowiska laboratoryjnego do eksperymentalnego badania możliwości zastosowania metody porównawczej do wykrywania zainfekowanych serwerów DNS. Realizacja przedmiotowego zadania wymagała zbudowania środowiska komputerowego, w którym działać będzie określona liczba serwerów DNS oraz opracowania oprogramowania, którego zadaniem było odpowiednie infekowanie i naprawianie wspomnianych serwerów. Ponadto wymagane było opracowanie aplikacji do realizacji wielokrotnych badań środowiska z wykorzystaniem zaproponowanej *Metody*. Opis środowiska laboratoryjnego przedstawiono w punkcie 4.1, natomiast zaproponowane rozwiązanie opisano w punkcie 3.

Przeprowadzone testy empiryczne (dla dużej liczby przypadków) wykazują jedynie połowiczną skuteczność proponowanego rozwiązania. Metoda porównawcza znakomicie radzi sobie z detekcją serwerów DNS zainfekowanych różnymi błędnymi adresami IP. Niestety, jednocześnie wykrywanie serwerów zainfekowanych tymi samymi adresami IP nie jest możliwe. Z tego względu wymagane są dalsze badania w kierunku zwiększenia skuteczności proponowanego rozwiązania, np. poprzez weryfikację możliwości zastosowania w przedmiotowym obszarze metody PMC lub BGM.

## Literatura

- [1] ARCIUCH A., *Techniczne aspekty diagnozowania sieci procesorów o łagodnej degradacji typu sześcian 4-wymiarowy metodą prób porównawczych*. Przegląd Teleinformatyczny 2013, nr 2, s. 3-11.
- [2] CHANG G.Y., CHEN G.H., CHANG G.J., *(t,k) – Diagnosis for Matching Composition Networks under the MM\* Model*. IEEE Transactions Computers, 2007, 56, 1, pp. 73-79.
- [3] FAN J., *Diagnosability of crossed cubes under the comparison diagnosis model*. IEEE Transactions on Parallel and Distributed Systems, 2002, vol. 13, 10, pp. 687-692.
- [4] GRABOWSKI T., *DNS spoofing, czyli podszywanie się pod serwer DNS*. Hakin9, nr 1, [http://www.centrum.bezpieczenstwa.pl/artykuly/h9\\_dns.pdf](http://www.centrum.bezpieczenstwa.pl/artykuly/h9_dns.pdf) (dostęp: 22.04.2015 r.).
- [5] KLEIN A. SHULMAN H. WAIDNER M., *Internet-Wide Study of DNS Cache Injections*. IEEE INFOCOM 2017 – IEEE Conference on Computer Communications, 2017, pp. 1-9.
- [6] KULESZA R., *Podstawy diagnostyki sieci logicznych i komputerowych*. Wyd. II, Warszawa, ITA, WAT, 2000.
- [7] KULESZA R., ZIELIŃSKI Z., *Wnikliwość diagnozowania sieci procesorów metodą porównawczą*. Systemy czasu rzeczywistego. Postępy badań i zastosowania. Red. Z. Zieliński, WKŁ, Warszawa, 2009, s. 211-225.
- [8] KULESZA R., ZIELIŃSKI Z., *Diagnosis resolution of processors' network using the comparison method*. Przegląd Elektrotechniczny (Electrical Review), 2010, vol. 89, Nr 9, s. 157-162.
- [9] LAI PL., *Adaptive system-level diagnosis for hypercube multiprocessors using a comparison model*. Journal Information Sciences—Informatics and Computer Science, Intelligent Systems, Applications: An International Journal, 2013, vol. 252, pp. 118-131.
- [10] LIANG JR., FENG H., DU X., *Intermittent Fault Diagnosability of Interconnection Networks*. Journal of Computer Science and Technology, 2017, vol. 32, no 6, pp. 1279-1287.
- [11] LIANG J., ZHANG Q., *The t/s – Diagnosability of Hypercube Networks Under the PMC and Comparison Models*. IEEE Access, 2017, vol. 5, pp. 5340 - 5346.
- [12] LIN L.M., XU L., ZHOU S.M., *Conditional diagnosability and strong diagnosability of Split-Star Networks under the PMC model*. Journal Theoretical Computer Science, 2015, vol. 562, issue C, pp. 565-580.
- [13] SENGUPTA A., DAHBURA A.T., *On self-diagnosable multiprocessor systems: Diagnosis by the comparison approach*. IEEE Transactions on Computers, 1992, vol. 41, 11, pp. 1386-1396.

- [14] SPARKS, NEO, TANK, SMITH, DOZER, *The Collateral Damage of Internet Censorship by DNS Injection*. Newsletter ACM SIGCOMM Computer Communication Review, 2012, vol. 42, issue 3, pp. 21-27.
- [15] STRZELECKI Ł., RENCZEWSKI K., *Adaptacyjna metoda wyznaczania ekonomicznych m-diagnozowalnych struktur opiniowania diagnostycznego typu PMC*. Biuletyn ITA, 2008, nr 25, s. 139-153.
- [16] STRZELECKI Ł., *Metody projektowania ekonomicznych t-diagnozowalnych struktur diagnostyki systemowej dla sieci procesorów typu binarnego sześcianu 4-wymiarowego*. Rozprawa doktorska, WAT, WCY, Warszawa, 2012.
- [17] STRZELECKI Ł., ZIELIŃSKI Z., *Projektowanie struktur opiniowania diagnostycznego z wykorzystaniem algorytmu genetycznego*. Biuletyn ITA, 2009, nr 27, s. 19-31.
- [18] STRZELECKI Ł., *Wyznaczanie struktur diagnozowania porównawczego przy wykorzystaniu algorytmu genetycznego*. Przegląd Teleinformatyczny, 2016, nr 3-4, s. 19-30.
- [19] STRZELECKI Ł., *Wyznaczanie struktur diagnozowania porównawczego z użyciem metody PSO*. Przegląd Teleinformatyczny, 2017, nr 1-2, s. 3-12.
- [20] TREVISAN M., DRAGO I., MELLIA M., MUNAFO M.M., *Automatic Detection of DNS Manipulations*. IEEE International Conference on Big Data, 2017, pp. 4010-4015.
- [21] WANG R., ZHU Q., *The h-extra conditional diagnosability of burnt pancake networks under the PMC model*. 2017 First International Conference on Electronics Instrumentation & Information Systems (EIIS), 2017, pp. 1-6.
- [22] ZHU L., HU Z., HEIDEMANN J., WESSELS D., MANKIN A., SOMAIYA N., *Connection-Oriented DNS to Improve Privacy and Security*. Proceedings of the 2015 IEEE Symposium on Security and Privacy, 2015, pp. 171-186.

## **Exploring the possibility of using a comparative method to prevent effects of DNS Injection attacks**

ABSTRACT: The article presents the problem of using the comparative method to prevent effects of DNS Injection attacks. A solution based on the presented method was proposed, and a developed laboratory environment was described for experimental examination of the effectiveness of the developed solution.

KEYWORDS: MM method, Malek Maeng model, comparative method, diagnostics, DNS server, DNS Injection

*Praca wpłynęła do redakcji: 2.07.2018 r.*

# Quality study of user activity using mobile device

## Tap, double tap, flick gestures

**Artur ARCIUCH, Antoni M. DONIGIEWICZ**

Institute of Teleinformatics and Automation, Faculty of Cybernetics,  
Military University of Technology,  
ul. Gen. W. Urbanowicza 2, 00-908 Warszawa  
{artur.arciuch}, antoni.donigiewicz@wat.edu.pl

**ABSTRACT:** The article presents the results of research on the quality of gestures performed by users using a mobile device. As a mobile device, the Nokia Lumia 800 smartphone was used. The results of the research concern the basic gestures of tap, double tap and flick, and include the execution time and the precision of the gesture. The results take into account the division of users into age groups and groups using and not using a smartphone every day. A comparison of the designated characteristics between groups is presented. Research on other gestures will be presented in the next paper.

**KEYWORDS:** tap, double tap and flick gestures, gestures entered (input) with a finger, mobile device, gesture execution time, gesture precision

### 1. Introduction

Among the quality studies conducted in the field of user activity, particular attention has recently been paid to quality studies of smartphone (and/or tablet) user activity. The reason for this is the common use of such devices, and the multitude of tasks performed with them. More and more frequently, this refers both to professional and personal tasks.

Knowing the characteristics of the quality of the gestures made by smartphone or table users constitutes the basis for the assessment of the performance of tasks comprising several consecutive, similar or different gestures. The most common interesting gesture quality indicators include gesture time, and the precision of indicating objects on a screen. In the context of design, it is necessary to learn whether the quality of gestures made by users belonging to various age groups differs significantly or if the users' ages are irrelevant in this scope. Another factor worth investigating is whether the quality of the gestures made by people using mobile devices on an everyday basis differs from the quality of gestures made by people who use such devices less frequently. The time during which the user's attention is concentrated on making a given gesture is also a relevant factor. At that time, the user is looking at the screen, so they are not focussing on other external objects. Such external objects may include field obstacles, pedestrian crossings, or roads while driving a car (or riding a bicycle). Irrational behaviour of users holding smartphones can often be observed.

This paper presents the results of studies aimed at quantifying the quality of tap, double tap and flick gestures made by users on a mobile device screen, i.e. a Nokia Lumia 800 smartphone. Taking into account ongoing studies and analyses [11], [13], the basic gesture quality characteristics used in this paper are: gesture duration and precision (distance between screen touching point and touched object centre) and gesture error probability. The gestures were made with fingers, and 60 people in the 16-66 age bracket were involved in the studies.

The gestures included in the studies are typical for smartphone users.

The "tap" gesture involves short, single touch of a screen (tapping its surface) and is generally used to start applications, select options or input data. This gesture ends when the finger or stylus is removed from the screen.

When making a "double tap" gesture, a user quickly touches the screen twice (taps the surface twice). This gesture is typically used to toggle between normal and zoomed-in views, e.g. of websites in a browser. This gesture is used relatively rarely. Similarly to the tap gesture, it ends when the finger or stylus is removed from the screen after the second tap.

The "flick" usually involves swiping (sliding) a finger across a screen in a selected direction. This gesture actually ends when the finger is removed from the screen, but the screen objects remain in motion. This gesture has multiple uses. Most commonly, it is used to move content or scroll lists, and can occur after the "pan" gesture.

The results of studies concerning other typical gestures (e.g. pan, pinch and stretch, etc.) will be presented in an upcoming paper.

## **2. Selected studies documented in literature**

While discussing studies related to mobile devices and documented in the applicable literature, special attention has been paid to study results concerning gestures made on the screens of mobile devices, or directly on smartphones.

The majority of studies presented in the literature have been conducted using a stylus on a touch screen.

Studies concerning the selection and control of objects on a Pocket PC mobile computer screen are presented in publication [3]. These studies were aimed at determining the efficiency of younger and older adults while performing simple tasks using a stylus. The mobile device used for the purposes of these studies was a medium-sized Compaq iPAQ PocketPC 3950 computer operated by an Intel PXA250 processor. This device comes with a 57.6 mm × 76.8 mm screen with a resolution of 240 × 320 pixels. During the studies, the mobile computer was connected to a desktop using a USB cable.

Sixty people participated in the studies. The participants were presented with four types of tasks: tapping, touching, direct (straight) and circular swiping. The “tapping” tasks involved touching a green circle with a stylus, which resulted in the display of a red circle (target) to be touched. The target circle diameter was 16, 24 or 32 pixels. Circles (targets) were displayed within a distance three, four or five times larger in relation to the target circle diameter and at one of the eight angles in relation to the green circle’s location.

All studies concerning the above-mentioned tasks were preceded with training tests. The study concerning each individual comprised two blocks with 72 randomly arranged tasks each. Such factors as task performance time and precision were recorded. Based on the recorded results, it was possible to draw the following conclusions:

For the purpose of the “tap” task, such factors as the target dimensions and the user's age had a significant impact on task completion time and precision. The distance from the target had a significant impact on the completion time, which did not apply to precision. The task sequence had a significant influence on precision, which did not apply to completion time.

For the purpose of the “touch” task, such factors as target dimensions, the distance from the target and the location angle had a significant impact on task completion time, which did not apply to precision. However, such factors as the

age group and task sequence did not have any significant impact on task completion precision and time.

Details regarding straight and circular operation as well as the average completion time and precision for each task for each age group are presented in paper [3].

The assessment of the effectiveness of selecting objects using a stylus on a small touch screen is presented in paper [4]. The users' actions were assessed on the basis of experiments involving three different devices, i.e. a mobile phone, a palmtop and a tablet. The phone's screen diameter was 2.9", the palmtop's was 3.6" and tablet's was 10.2". All devices were placed in an upright position. The smaller devices (the phone and palmtop) were used in two positions, i.e. held in the hand and placed on a desk. Due to its size, the tablet was used only in a flat position, i.e. laid on a desk.

The test consisted of indicating two rectangular targets in a predefined sequence, i.e. the first target, the second target. After the second target was indicated, the next set of targets was displayed. The tests were run in two scenarios, i.e. errors are permissible and errors are impermissible.

The size of the first target (a black square) was constantly 6.0 mm, while the size of the second target (a white square) was randomly selected by the phone from the 2-3 mm range. The size of the second target on the palmtop or tablet screen was selected randomly from a range ensuring the same difficulty factor as for the phone.

The test results made it possible to formulate the following conclusions. The throughput of hand-held devices was significantly lower than for devices laid on the desk. The error rate (percentage) for a hand-held palmtop was much lower than for a palmtop laid on the desk. No significant difference in the error rate for a phone in the same conditions was determined. Details regarding device throughput and error rates for such testing conditions are provided in paper [4].

Studies concerning the three methods of data input (with a finger, stylus and mouse, as the performance reference point) are presented in paper [5]. The tests involved various types of basic user actions aimed at selecting targets by tapping, swiping and radial swiping.

A HP Compaq 2710p (screen diameter 12.1", screen resolution 1280 × 800 pixels) was used for test purposes. When a Logitech MX610 laser mouse was used, the screen was set in an upright position, however, with a stylus, the



screen was in a flat position. Data was input with a touch on an HP TouchSmart 600 PC (23" screen, resolution 1920 × 1080 pixels).

Eighteen people (including 6 women) participated in the tests (age bracket: 21-59). The participants were trained in the tasks involved in the test.

The efficiency of tapping and swiping was studied on the basis of a typical single-direction test [14].

The following test results are interesting. During the "tap" gesture test, the input method (finger, stylus, mouse) had a significant impact on the error rate. The significant impact of the difficulty factor on the error rate was determined. A low precision of finger indication resulted from the lack of feedback on the finger position right before touching the screen, and the finger size in relation to the object.

While analysing the test results for object indication time, it was determined that data input was fastest using a finger, and slowest using a mouse. Thus a significant impact of the input method and difficulty factor on the object indication time was detected.

As far as swiping is concerned, the error rate was lower than for tapping. No significant impact of the input method (finger, stylus, mouse) on the error rate was determined.

Details concerning input test results and test results for radial swiping are presented in paper [5].

Three experiments concerning the selection of objects are presented in paper [6]. Two experiments (one-dimensional and two-dimensional) are more interesting from the point of view of this topic, as the third one concerns typing using a touch keyboard. The paper itself concerns another issue, i.e. using the sum of two independent normal distributions to interpret the distribution of the final points input with a finger, on a screen.

A HTC NEXUS smartphone, running Android, was used in these tests. It came with a capacitive touch screen (size 48 × 80 mm, resolution 480 × 800 pixels). 12 people (including 3 women) participated in the tests (age bracket: 18-45).

The one-dimensional experiment consisted of generating a sound, displaying a grey, 6 mm wide, horizontal bar (the initial bar) and displaying a red, horizontal target bar (of varied width). Successful selection of the initial bar resulted in its colour changing to green along with the simultaneous

generation of an audio signal. Raising a finger after touching the target bar ended each task performed by a given person.

The test results demonstrated that the error rate significantly exceeded 4%, i.e. indicated very low test performance precision. The effective object width, i.e. the actual spread of touch points, was more than twice as large as the nominal target width. See Table 1 for average task completion times and error rates for the  $A$  and  $W$  value combinations.

**Tab. 1. Average task completion times and error rates for one-dimensional experiment [6]**

$A$ [mm]	$W$ [mm]	Average time [ms]	Error rate [%]
20	2.4	432	29
30	2.4	483	38
20	4.8	383	14
30	4.8	433	11
20	7.2	367	3
30	7.2	394	6

The two-dimensional experiment was similar. The difference consisted of the fact that circles of diameter  $W$  were the targets.

Similarly to the one-dimensional experiment, it was determined that the error rate significantly exceeded 4%. Average task performance times in the two-dimensional experiment were similar to the values determined during the one-dimensional experiment, however, the error rates were higher. See paper [6] for details.

An examination of the impact of the device size on performance, and the impact of the screen size on input quality for various tasks is presented in article [7].

Thirty people (including 15 men) participated in the tests (age bracket: 19-43). All participants had experience in using notebooks, and the majority of them had experience in using touch pads.

ASUS Eee PC 701SD, 900HA, 1000HG and 1101HA models were used during the tests (screen diameters: 7", 8.9", 10.1" and 11.6", respectively). The screen resolution was set to  $800 \times 480$ ,  $1024 \times 600$ ,  $1024 \times 600$  and  $1366 \times 768$  pixels, respectively. A comparable RAS (resistive) touch screen was installed for each model to compare the operation of a touch pad and touch screen with the same model.

The tests comprised five tasks. Task 1 (omnidirectional clicking) and task 2 (omnidirectional selection with “tap” gestures) were easy tasks. Task 3 (consecutive selection and clicking) and task 4 (consecutive selection by tapping) and task 5 (typing in Mandarin using a keyboard) were complex tasks.

The tests within tasks 1 and 2 were performed as per Appendix B of ISO 9241-9 [14]. During task 3 (consecutive selection and clicking), users selected icons by means of a touch pad. An icon (button) 1 was selected. Next, buttons from 2 to 5 were selected consecutively. Similarly, during task 4 (consecutive selection by tapping), the user selected icons (buttons) consecutively, using a touch screen. The total distance (from button 1 to button 5) was maintained at 1000 pixels.

The average operation time measurement results in this test showed that the tasks were performed more slowly on models fitted with smaller screens. Taking into account all tasks, a significant relation between a device size and task performance time was demonstrated.

Significant differences in the performance of simple tasks with a touch pad (task 1: multi-directional clicking) and tasks performed with a touch screen (task 2: multi-directional tapping test) were also demonstrated.

Taking into account the fact that the error rate was below 5%, the error percentage values were not significantly different across the various device models.

The details of the input test results, including average task performance times and error rates for all device models and all tasks, are presented in paper [7].

The tests presented in paper [8] are considered to be relevant studies in the field of the touch-input of data. The tests consisted of selecting objects. The input device was a mouse connected to a laptop and a smartphone touch screen (inputting data with a finger).

Sixteen people (including 6 women) participated in the tests. The laptop tests were performed in a seated position, and the touch screen tests were conducted in a standing position. The tests were performed with an LG Nexus 4 smartphone (touch screen and Android 4.2.2 OS). Screen size 61 × 102 mm, resolution 768 × 1184 pixels.

The study included typical, one-dimensional tests (a one-directional test and indication of objects) and two-dimensional tests (a multi-directional test and indication of objects) as per ISO 9241-9 [14].

Each participant performed tests on each device for each block of conditions and each task type (one- and two-dimensional), taking into account the distances between objects and object sizes.

The test results included throughput, movement time and error rate. It was demonstrated that efficiency is higher by 42-88% with touch operation (direct input), in relation to the efficiency obtained when a mouse was used (indirect input). The throughput for the two-dimensional tasks was significantly lower than for the one-dimensional tasks.

No significant differences between movement time and device placement (smartphone held in hand, smartphone laid on a desk) were determined. However, differences between movement duration in relation to given task type (shorter times during one-dimensional tasks) were detected.

No statistically relevant differences in error rates in relation to task types and device placement were determined. Detailed information concerning these test results are published in paper [8].

Apart from the studies presented in the available literature that have been presented in detail, numerous interesting studies can be mentioned whose detailed descriptions are impossible to present within a single paper. For example, they include studies concerning the time and precision of object indication on a smartphone screen while driving a car (in a simulator) presented in paper [9]. Participants were asked to take their right hand from the steering wheel and indicate and touch a target on a smartphone screen, defined by the person running the experiment. Movement times and touch coordinates were recorded.

Paper [10] presents studies concerning the impact of device feedback after the selection of very small (2-4 mm) targets on touch screens. The feedback comprised the contact point on a touch screen after a user raised their finger. The tests showed that application of device feedback resulted in increasing the object indication time, but also in significantly decreasing the error rate value.

The studies presented in article [11] were aimed at such gestures as swiping, pinching (spreading) and complex gestures such as double touch with swiping and swiping with double tapping, as well as swiping down the screen. Children, adults and seniors participated in the tests. The tests confirmed that children and adults achieve better results than seniors.

Attention should also be paid to studies in which precision and speed of two zooming methods are compared, i.e. the tap-and-drag and traditional pinch-to-zoom method [12]. The results showed that the tap-and drag method was 47% more effective than pinch-to-zoom, and the effectiveness was measured on the basis of the number of gestures necessary to complete a test.

### 3. Test conditions

#### 3.1. Device used in tests

A mobile device, i.e. a Nokia Lumia 800 smartphone (hereafter referred to as “the smartphone”) was used during the tests. Basic smartphone parameters: MS Windows Phone 7.5 Mango operating system, single-core Qualcomm MSM8255T 1.40 GHz processor. Screen type: capacitive display with the multi-touch function, 3.7” in size. Display technology: AMOLED with ClearBlack, facilitating operation in WVGA (480 × 800 px, 252 ppi ~54.7% screen-to-body ratio) resolution [19], [20]. A capacitive display precludes using a stylus.

#### 3.2. Application facilitating test completion

The application facilitating test completion was of modular design (Fig. 1):

- *Mobile Application* module run on a phone operated by the Windows Phone system;
- *Analysis Program* module run on an IBM PC with Windows 7 installed;
- *Server Program* module, i.e. a webservice run on an IBM PC.

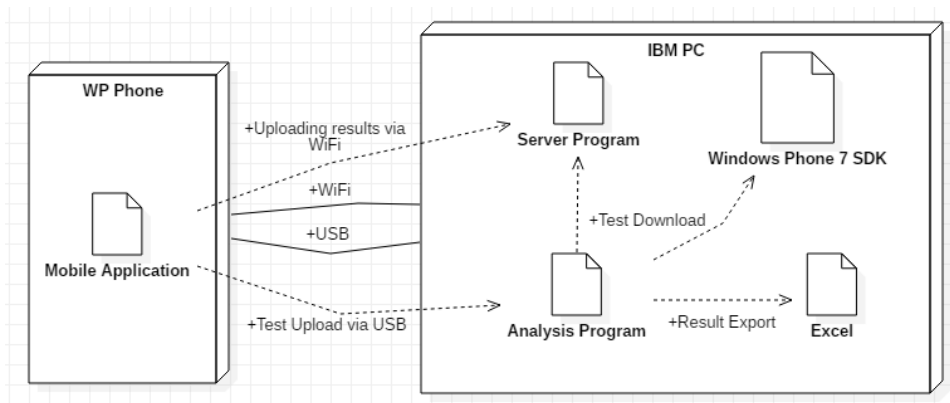
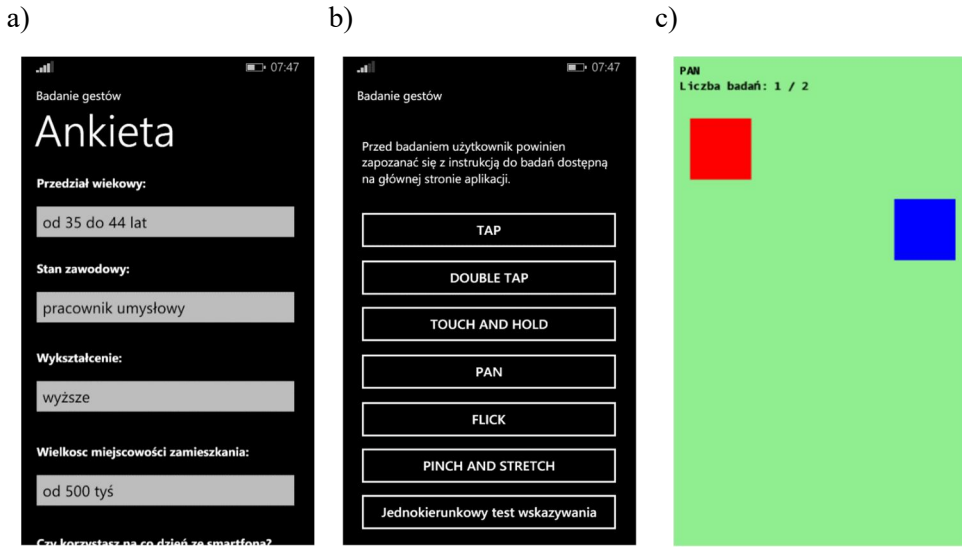


Fig. 1. Architecture of application facilitating test completion [13]

After starting the *Mobile Application* module, a test participant filled in a short survey (Fig. 2a) and made certain gestures, repeating them a preset number of times (Fig. 2b). See Fig. 2c for an example screen showing the status after objects were displayed and before a gesture was made.



**Fig. 2. Mobile Application a) survey, b) measurement types, c) a view during one “pan” gesture**

Test results were sent from the *Mobile Application* module to the *Analysis Program* module using the *Server Program* webservice via WiFi or directly from the *Mobile Application* module to the *Analysis Program* module via a USB interface. The *Analysis Program* (Fig. 3) could present total results (all test participants) or single test results (gesture or person).

Selected measurement results could be exported from the *Analysis Program* module to an Excel file. In the studies presented, the results were exported to Excel files for further processing. The basic data included in these Excel files comprised:

- indication of whether a target was hit or missed;
- distance between an object and finger at the moment of touching the screen (for the “flick” gesture, this parameter was unavailable);
- gesture completion time.

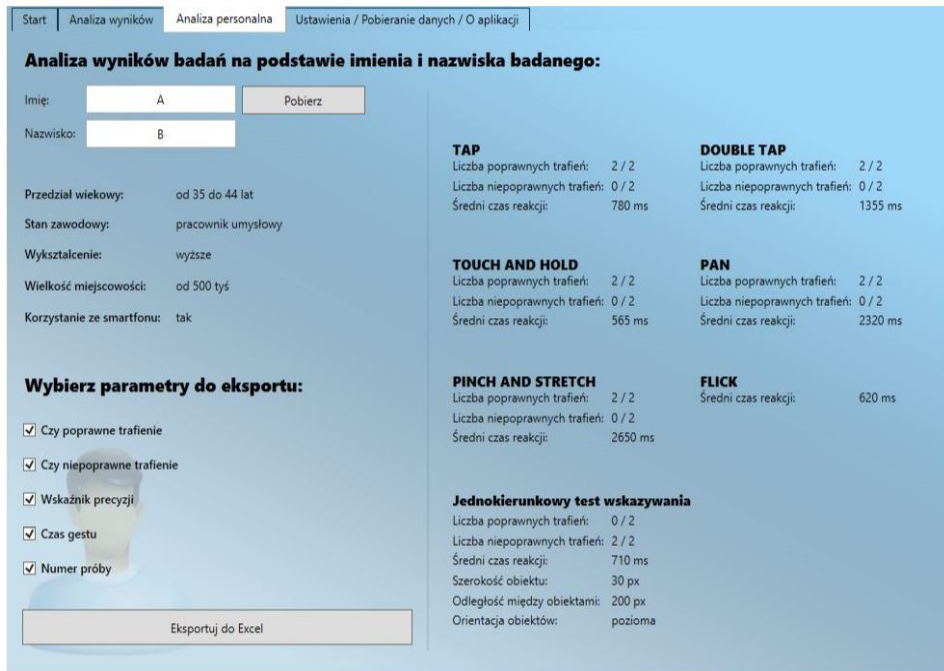


Fig. 3. Example of the display in the *Analysis Program* module

### 3.3. Number of test participants and test conditions

See Table 2 for the number of test participants. 60 people participated in the tests (age bracket: 16-66, with the majority of young and middle-aged people). Five age groups (1-5) were distinguished. The test participants included 2 women.

Tab. 2. Number of test participants

Age brackets [years of age]	16-24	25-34	35-44	45-54	≥55	Total
Group number	1	2	3	4	5	
Number of participants	14	5	23	13	5	60

The test participants included mostly men, white-collar workers with higher education degrees as well as several students, i.e. secondary school graduates. Some participants used smartphones on an everyday basis, and some did not use them on an everyday basis (Tab. 3). All participants repeated the gestures 30 times.

**Tab. 3. Participants using and not using smartphones on an everyday basis**

Feature	Uses a smartphone	Does not use a smartphone
Number of participants	48	12

The tests were conducted from 8.00 a.m. to 4 p.m., in one room. A smartphone was held in an upright position, in a left hand (none of the participants were left-handed). The participants did not move (walk). Most of them were sitting. The next object (target) requiring a participant's action was displayed immediately after finishing the previous gesture. No training or familiarising activities were conducted prior to the tests. The participants were informed orally about the following tasks to complete.

The tests presented in this paper involved tapping, double tapping, flicking and panning.

The main parameters determined on the basis of measurements taken were:

- average time of making a gesture by a user;
- average gesture performance precision (average distance between an object and finger).

Statistical tests adequate to the study conditions were used for calculations presented in this paper [15], [16], [18]. Direct calculations and diagrams were elaborated using the MATLAB software for scientific and technical calculations [17].



#### 4. “Tap” test results

During the “tap” test, a red square ( $100 \times 100$  pixels) was subjected to manipulations (Fig. 4). Thus, the square was slightly larger than a typical finger touching a screen surface. The square was displayed in randomly selected positions, on the smartphone screen.

Values measured for a participant (Fig. 5):

$x_{so}, y_{so}$  – coordinates of the object (square) centre;

$x_p, y_p$  – coordinates of the screen point touched by the participant;

$t_a$  – object (square) displaying time;

$t_b$  – time in which the participant removed their finger from the screen.

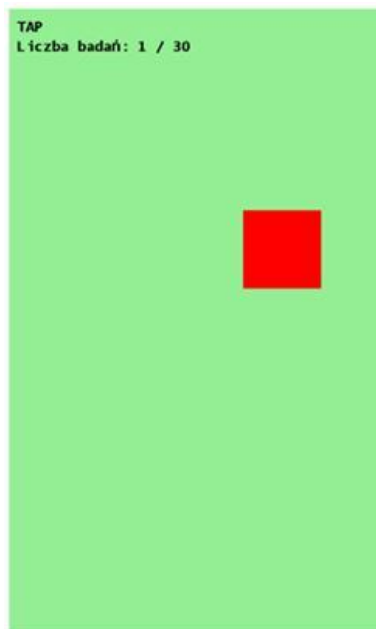
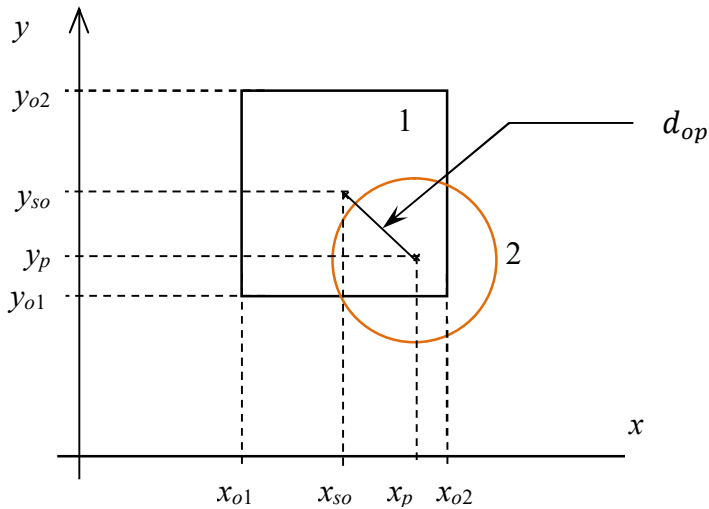


Fig. 4. Nokia Lumia 800 smartphone view while making the “tap” gesture



**Fig. 5.** Example of object 1 (square) and finger 2 locations on the screen, where:  
 $(x_{o1}, y_{o1}), (x_{o2}, y_{o2})$  – coordinates of the object 1 (square) centre;

“Tap” gesture correctness condition:

$$(x_{o1} \leq x_p \leq x_{o2}) \wedge (y_{o1} \leq y_p \leq y_{o2}) \quad (9)$$

where:  $x_{o1}, x_{o2}, y_{o1}, y_{o2}$  - as in Fig. 5.

The following values were determined during the test.

Distance  $d_{op}$  between the object and finger (hereinafter referred to as “precision”):

$$d_{op} = \sqrt{(x_{so} - x_p)^2 + (y_{so} - y_p)^2}. \quad (10)$$

Length of time in which the participant made the gesture:

$$t_1 = t_b - t_a,$$

where:  $t_a$  – object displaying time;

$t_b$  – time in which the participant removed their finger from the screen.

Following the measurements, such aspects as average gesture precision, average gesture completion time and the probability of an error while making a “tap” gesture were determined.

Average gesture precision:

$$\bar{d}_{op} = \frac{1}{n} \sum_{i=1}^n d_{op i},$$

where:  $d_{op i}$  – gesture precision following the  $i$ -th measurement;

$n$  – number of gesture repetitions ( $n = 30$ ).

Average gesture completion time:

$$\bar{t}_1 = \frac{1}{n} \sum_{i=1}^n t_{1 i}, \quad (11)$$

where:  $t_{1 i}$  - gesture completion time following the  $i$ -th measurement;

$n$  - number of gesture repetitions ( $n = 30$ ).

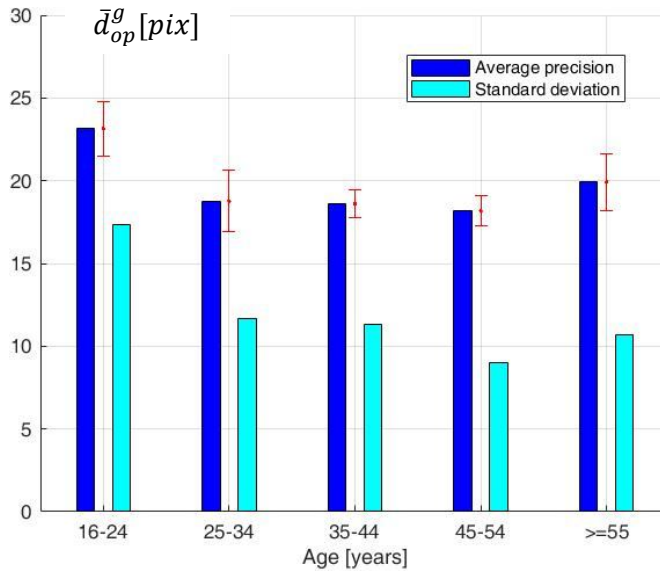
The probability of an error while making a “tap” gesture was determined by means of the gesture completion correctness condition (compare dependence [9]).

In order to make the necessary comparisons, the average “tap” gesture precision of  $\bar{d}_{op}^g$  was determined in age group  $g$ ,  $g \in \{1, 2, 3, 4, 5\}$ . The calculation results are presented in Table 4.

See Figure 6 for the average precision, standard deviation of precision and precision confidence ranges for “tap” gesture completion in the given age groups.

**Tab. 4. Average precision  $\bar{d}_{op}^g$  for “tap” gesture completion in given age group  $g$**

Age bracket [years of age]	16-24	25-34	35-44	45-54	$\geq 55$
Group $g$	1	2	3	4	5
Average tap gesture precision $\bar{d}_{op}^g$ [pix]	23.15	18.77	18.62	18.18	19.92



**Fig. 6. Average precision  $\bar{d}_{op}^g$ , standard deviation of precision and precision confidence ranges for “tap” gesture completion in given age groups**

Similarly, the average time for “tap” gesture completion  $\bar{t}_1^g$  in a given age group  $g$  was determined,  $g \in \{1, 2, 3, 4, 5\}$ . Apart from the gesture completion time in an age group, gesture completion time standard deviation and confidence ranges for gesture completion time in an age group were determined. See Table 5 for calculation results for gesture completion average time for all gestures and for correctly completed gestures (condition [10] fulfilled).

See Figure 7 for the average time, standard deviation of time and time confidence ranges for “tap” gesture completion in given age groups.

**Tab. 5. Average time  $\bar{t}_1^g$  for “tap” gesture completion in given age group  $g$**

Age bracket [years of age]	16-24	25-34	35-44	45-54	$\geq 55$
Group $g$	1	2	3	4	5
Average time $\bar{t}_1^g$ for “tap” gesture completion (all gestures) [ms]	584.2	639.5	751.7	931.0	952.5
Average time $\bar{t}_1^g$ for “tap” gesture completion (correct gestures) [ms]	588.2	641.6	753.1	931.7	952.5

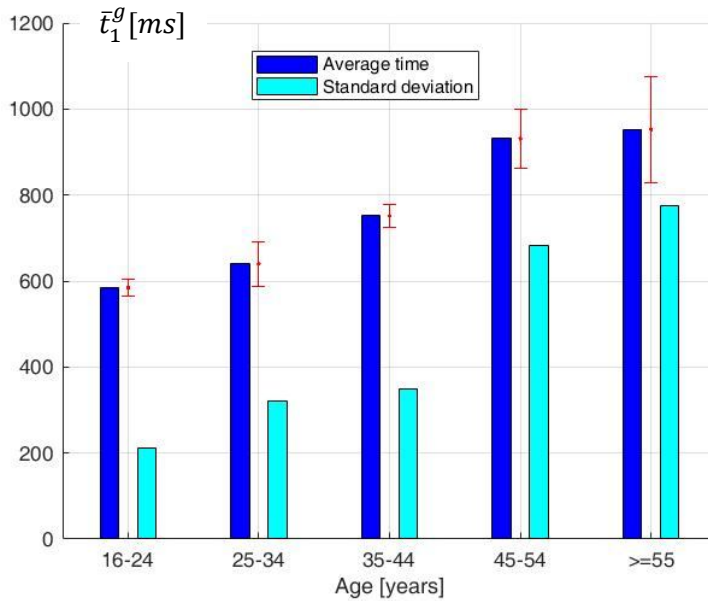


Fig. 7. Average gesture completion time  $\bar{t}_1^g$ , standard deviation and confidence ranges for “tap” gesture completion in given age groups (all gestures)

Taking into account the “tap” gesture correctness requirement (condition [9]), the probability  $Pb_{tap}$  of gesture completion errors in the given age groups was determined. The results are shown in Figure 8.

Test results among age groups were compared, in particular between the first group and the other groups.

The following hypotheses regarding gesture completion average precision were formulated:

H0: equal average precision for age groups 1 and  $j$  ( $\bar{d}_{op}^1 = \bar{d}_{op}^j$ );

H1: different average precision for age groups 1 and  $j$  ( $\bar{d}_{op}^1 \neq \bar{d}_{op}^j$ ).

Similarly, the following hypotheses regarding gesture completion average time were formulated:

H0: equal average time for age groups 1 and  $j$  ( $\bar{t}_1^1 = \bar{t}_1^j$ );

H1: different average time for age groups 1 and  $j$  ( $\bar{t}_1^1 \neq \bar{t}_1^j$ ).

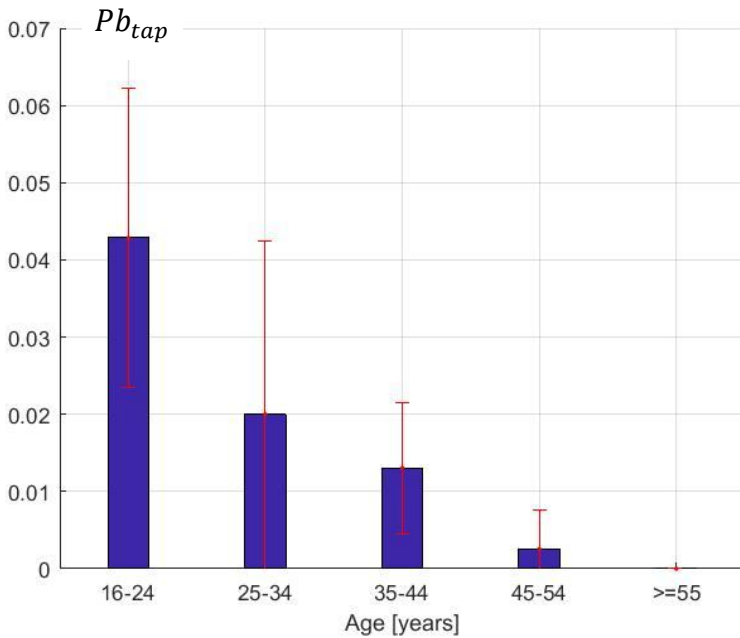


Fig. 8. Probability of errors  $Pb_{tap}$  and confidence ranges for “tap” gesture in age groups

Due to the fact that the tests were numerous, a typical test for comparing average values was used. The hypotheses were tested at the significance level of  $\alpha = 0.05$ , and the hypothesis verification results are given in Table 6.

Tab. 6. Results of comparison between average precision and average time of “tap” gesture completion, between group 1 and the other age groups

Groups subject to comparison	1-2	1-3	1-4	1-5
Decision on average precision comparison result	Reject H0	Reject H0	Reject H0	Reject H0
Decision on average time comparison result	No grounds to reject H0	Reject H0	Reject H0	Reject H0

The following designations were assumed.

$\bar{d}_{op}^u$  – average precision of gesture completion by users using smartphones on an everyday basis;

$\bar{d}_{op}^n$  – average precision of gesture completion by users not using smartphones on an everyday basis;

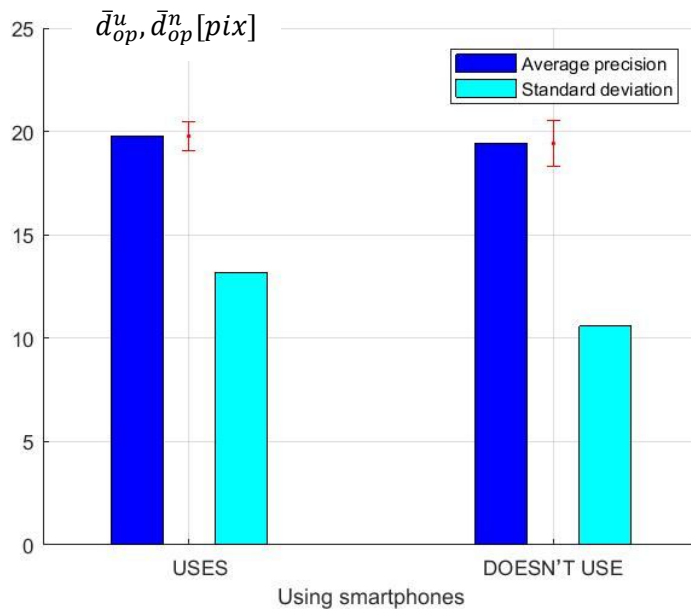
$\bar{t}_1^u$  – average time of gesture completion by users using smartphones on an everyday basis;

$\bar{t}_1^n$  – average time of gesture completion by users not using smartphones on an everyday basis;

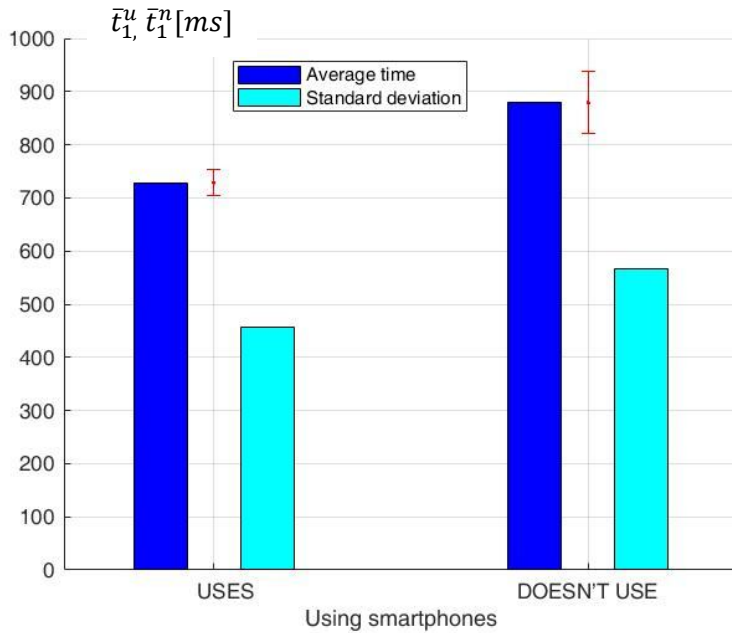
$Pb_{tap}^u$  – probability of errors in gesture completion by users using smartphones on an everyday basis;

$Pb_{tap}^n$  – probability of errors in gesture completion by users not using smartphones on an everyday basis.

On the basis of the results obtained, the average gesture precision, average gesture completion time and probability of a “tap” gesture error were determined within the group of participants using smartphones on an everyday basis and the group of participants not using smartphones on an everyday basis. The results are shown in Figures 9-11.



**Fig. 9. Average precision, standard deviation of precision and precision confidence ranges for “tap” gesture completion for participants who use and do not use smartphones on an everyday basis**



**Fig. 10.** Average time, standard deviation and precision confidence ranges for “tap” gesture completion time for participants who use and do not use smartphones on an everyday basis

It was also checked whether the fact that participants use smartphones on an everyday basis influenced the quality of their operation. The following hypotheses were formulated:

H0: equal average precision (participants using; participants not using smartphones on an everyday basis) ( $\bar{d}_{op}^u = \bar{d}_{op}^n$ ),

H1: different average precision ( $\bar{d}_{op}^u \neq \bar{d}_{op}^n$ ),

H0: equal average time ( $\bar{t}_1^u = \bar{t}_1^n$ ),

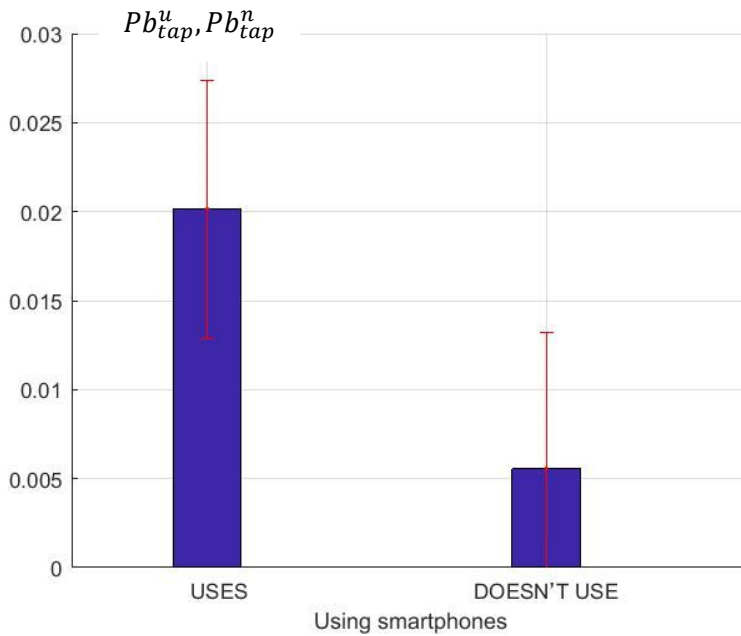
H1: different average time ( $\bar{t}_1^u \neq \bar{t}_1^n$ ),

H0: equal error probability ( $Pb_{tap}^u = Pb_{tap}^n$ ),

H1: different error probability ( $Pb_{tap}^u \neq Pb_{tap}^n$ ).

The hypotheses were tested at the significance level of  $\alpha = 0.05$ , and verification results are given in Table 7.





**Fig. 11. Error probability and confidence ranges for “tap” gesture completion for participants who use and do not use smartphones on an everyday basis**

**Tab. 7. Results of comparison between average time, average precision and error probability for “tap” gesture completion for participants who use and do not use smartphones on an everyday basis**

Compared parameter	Average time	Average precision	Error probability
Decision on H0	Reject H0	No grounds to reject H0	No grounds to reject H0

### Comments and conclusions regarding test results

On the basis of the results obtained, one can observe fairly significant standard deviation values, both for the average “tap” gesture completion precision and average “tap” gesture completion time, in various age groups. Practically, taking into account the average time  $\bar{t}_1^g$  of “tap” gesture completion

in an age group, no differences between all gestures and correct gestures (compare Tab. 5) can be observed.

Characteristic values include a decrease in standard deviation of precision (compare Fig. 6), an increase in standard deviation of gesture completion time (compare Fig. 7), and a decrease in gesture completion error probability (compare Fig. 8) in relation to the participants' age. Thus, it can be concluded that the older the users are, the more significantly slow and less precise gesture completion is (compare Tab. 6).

In all cases, the hypotheses regarding equality of average precision between the first age group and other groups were rejected. In the majority of cases (apart from the comparison with the second group), hypotheses regarding the equality of "tap" gesture average completion time between the first age group and other groups were rejected.

Taking into account participants using and not using a smartphone on an everyday basis, the hypothesis regarding the equality of the "tap" gesture average completion time between these groups was rejected. For average precision and error probability in "tap" gesture completion, there were no grounds for rejecting the hypotheses regarding the equality of these parameters among participants using and not using smartphones on an everyday basis.

## 5. "Double tap" test results

In the "double tap" gesture studies, the same object as for the "tap" gesture studies was used (compare Fig. 4). The square was displayed in randomly selected positions, on a smartphone screen.

The values measured for the test participant and the gesture completion correctness requirement were the same as for the "tap" gesture, however, the coordinates of the screen point touched by the participant, and the time of removing the finger from the screen referred to the second tap.

During those tests, values similar to the "tap" test were determined, i.e. the distance  $d_{op}$  between the object and finger (second tap on the object – precision) and gesture completion time, and time  $t_b$  was the time of removing the participant's finger from the screen for the second time.

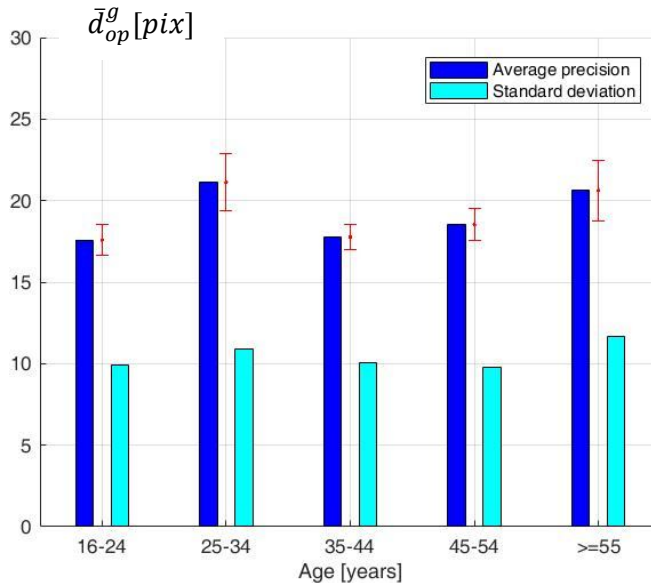
Following the measurements, such aspects as average gesture precision, average gesture completion time and probability of an error while making a "double tap" gesture were determined.

In order to make the necessary comparisons, average “double tap” gesture precision  $\bar{d}_{op}^g$  was determined in age group  $g$ ,  $g \in \{1, 2, 3, 4, 5\}$ . The calculation results are presented in Table 8.

**Tab. 8.** Average precision  $\bar{d}_{op}^g$  for “double tap” gesture completion in given age group  $g$

Age bracket [years of age]	16-24	25-34	35-44	45-54	$\geq 55$
Group $g$	1	2	3	4	5
Average “double tap” gesture precision $\bar{d}_{op}^g$ [pix]	17.58	21.11	17.76	18.53	20.61

See Figure 12 for the average precision, standard deviation of precision and precision confidence ranges for “double tap” gesture completion in the given age groups.



**Fig. 12.** Average precision  $\bar{d}_{op}^g$ , standard deviation of precision and precision confidence ranges for “double tap” gesture completion in the given age groups

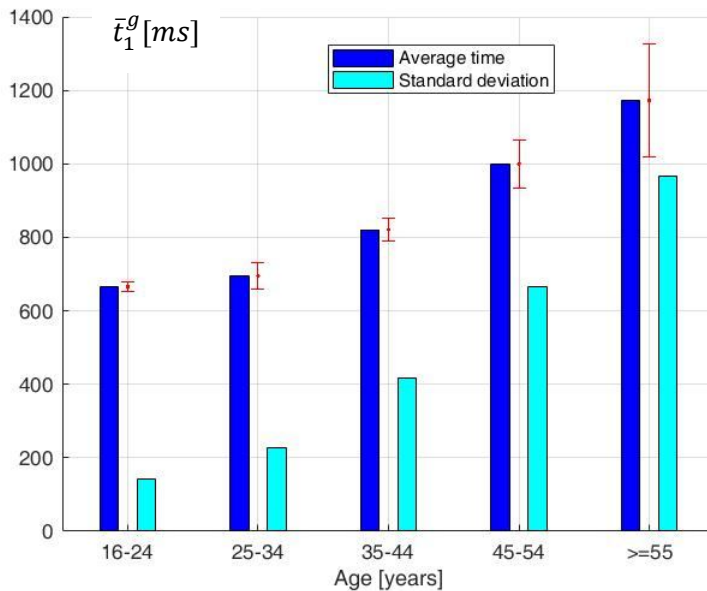
Similarly, the average time for “double tap” gesture completion  $\bar{t}_1^g$  was determined in a given age group  $g$ ,  $g \in \{1, 2, 3, 4, 5\}$ . Apart from the gesture completion time in the age group, gesture completion time standard deviation and confidence ranges for gesture completion time in the age group were

determined. See Table 9 for calculation results for gesture completion average times for all gestures and for correctly completed gestures (condition [10] fulfilled).

**Tab. 9. Average time  $\bar{t}_1^g$  for “double tap” gesture completion in given age group  $g$**

Age brackets [years of age]	16-24	25-34	35-44	45-54	$\geq 55$
Group $g$	1	2	3	4	5
Average time $\bar{t}_1^g$ for “double tap” gesture completion (all gestures) [ms]	665.7	695.3	821.0	999.8	1172.4
Average time $\bar{t}_1^g$ for “double tap” gesture completion (correct gestures) [ms]	665.7	696.0	821.2	999.8	1177.2

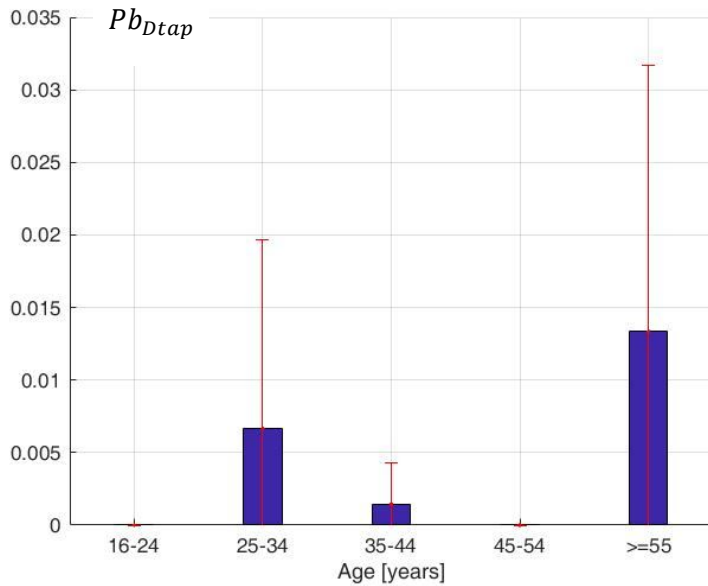
See Figure 13 for the average time, standard deviation of time and time confidence ranges for “double tap” gesture completion in given age groups.



**Fig. 13. Average gesture completion time  $\bar{t}_1^g$ , standard deviation and confidence ranges for “double tap” gesture completion in the given age groups (all gestures)**

Similarly, to the “tap” gesture, taking into account the gesture completion correctness requirement (condition [9], when the object was tapped for the

second time) the probability  $Pb_{Dtap}$  of the “double tap” gesture completion error, in age groups, was determined. The results are shown in Figure 14.



**Fig. 14. Probability of errors  $Pb_{Dtap}$  and confidence ranges for “double tap” gesture in age groups**

The test results among the age groups were compared, in particular between the first group and the other groups.

The following hypotheses regarding “double tap” gesture completion average precision were formulated:

H0: equal average precision for age groups 1 and  $j$  ( $\bar{d}_{op}^1 = \bar{d}_{op}^j$ );

H1: different average precision for age groups 1 and  $j$  ( $\bar{d}_{op}^1 \neq \bar{d}_{op}^j$ ).

Similarly, the following hypotheses regarding “double tap” gesture completion average time were formulated:

H0: equal average time for age groups 1 and  $j$  ( $\bar{t}_1^1 = \bar{t}_1^j$ );

H1: different average time for age groups 1 and  $j$  ( $\bar{t}_1^1 \neq \bar{t}_1^j$ ).

The hypotheses were tested at the significance level of  $\alpha = 0.05$ , and verification results are given in Table 10.

The following designations were assumed:

$\bar{d}_{op}^u$  – average precision of “double tap” gesture completion by users using smartphones on an everyday basis;

$\bar{d}_{op}^n$  – average precision of gesture completion by users not using smartphones on an everyday basis;

$\bar{t}_1^u$  – average time of “double tap” gesture completion by users using smartphones on an everyday basis;

$\bar{t}_1^n$  – average time of gesture completion by users not using smartphones on an everyday basis;

$Pb_{Dtap}^u$  – probability of errors in “double tap” gesture completion by users using smartphones on an everyday basis;

$Pb_{Dtap}^n$  – probability of errors in gesture completion by users not using smartphones on an everyday basis.

**Tab. 10. Results of comparison between average precision and average time of “double tap” gesture completion, between group 1 and the other age groups**

Groups subject to comparison	1-2	1-3	1-4	1-5
Decision on average precision comparison	Reject H0	No grounds to reject H0	No grounds to reject H0	Reject H0
Decision on average time comparison	No grounds to reject H0	Reject H0	Reject H0	Reject H0

On the basis of the results obtained, the average gesture precision, average gesture completion time and probability of a “double tap” gesture error were determined within the group of participants using smartphones on an everyday basis and the group of participants not using smartphones on an everyday basis. The results are shown in Figures 15-17.

It was also checked whether the fact that participants use smartphones on an everyday basis influences the quality of “double tap” gesture completion. The following hypotheses were formulated:

H0: equal average precision (participants using; participants not using smartphones on an everyday basis) ( $\bar{d}_{op}^u = \bar{d}_{op}^n$ ),

H1: different average precision ( $\bar{d}_{op}^u \neq \bar{d}_{op}^n$ ),

H0: equal average time ( $\bar{t}_1^u = \bar{t}_1^n$ ),

H1: different average time ( $\bar{t}_1^u \neq \bar{t}_1^n$ ),

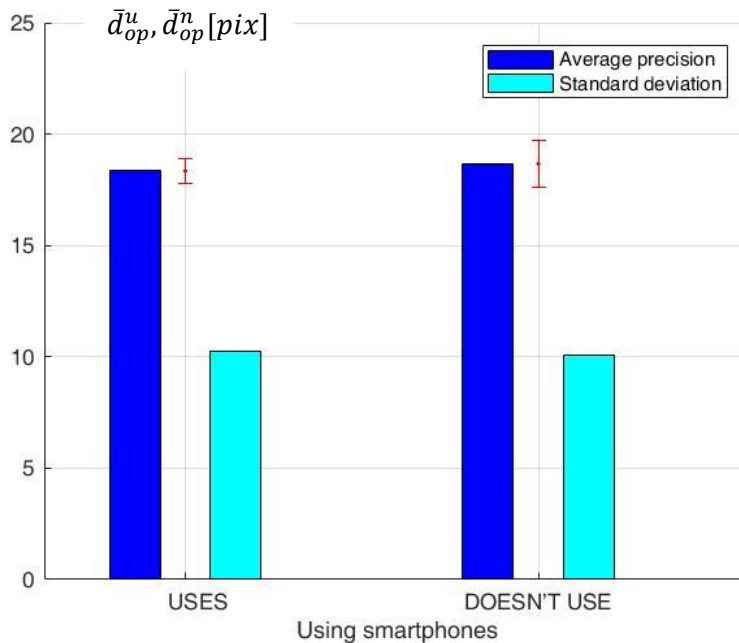
H0: equal error probability ( $Pb_{Dtap}^u = Pb_{Dtap}^n$ ),

H1: different error probability ( $Pb_{Dtap}^u \neq Pb_{Dtap}^n$ ).

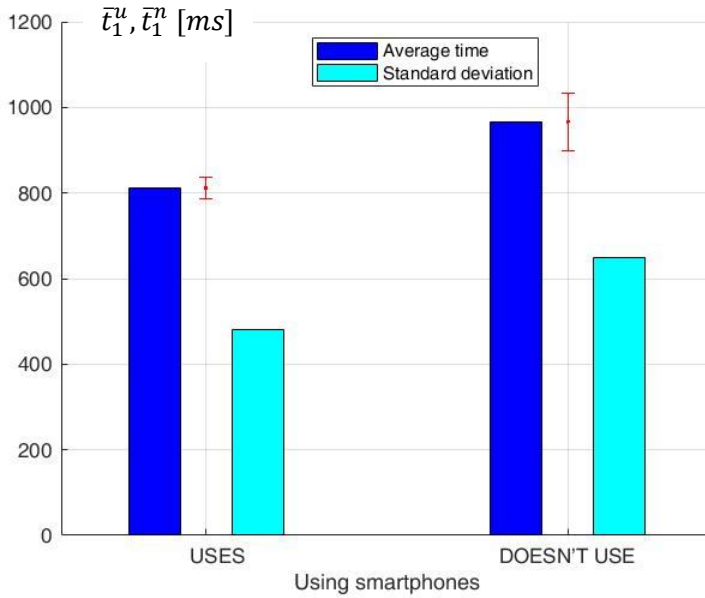
The hypotheses were tested at the significance level of  $\alpha = 0.05$ , and verification results are given in Table 11.

**Tab. 11. Results of comparison between average time, average precision and error probability for “double tap” gesture completion for participants who use and do not use smartphones on an everyday basis**

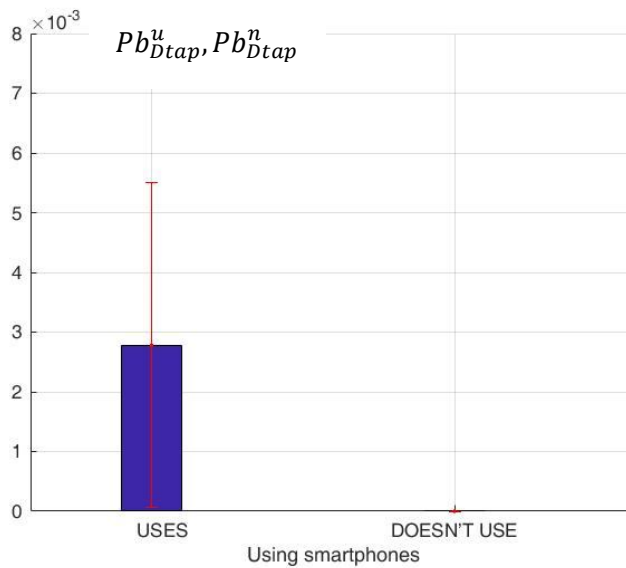
Compared parameter	Average time	Average precision	Error probability
Decision on H0	Reject H0	No grounds to reject H0	No grounds to reject H0



**Fig. 15. Average precision, standard deviation of precision and precision confidence ranges for “double tap” gesture completion for participants who use and do not use smartphones on an everyday basis**



**Fig. 16. Average time, standard deviation of time and precision confidence ranges for “double tap” gesture completion time for participants who use and do not use smartphones on an everyday basis**



**Fig. 17. Error probability and confidence ranges for “double tap” gesture completion error probability for participants who use and do not use smartphones on an everyday basis**



## **Comments and conclusions regarding test results**

On the basis of the results obtained, one can observe fairly significant standard deviation values for the average “double tap” gesture completion precision in each age group. A significant increase in the standard deviation of “double tap” gesture completion average time with the increase in the participants’ ages (compare Fig. 13) can be observed.

The result of the comparison of the “double tap” gesture completion average time among age groups is the same as for the “tap” gesture. The hypotheses regarding the equality of the “double tap” completion average time between the first age group and other groups (apart from the comparison with the second group) (compare Tab. 10) must be rejected.

Taking into account participants using and not using smartphones on an everyday basis, the results of the comparison of the average gesture precision, average gesture completion time and probability of an error while making the “double tap” gesture are the same as for the “tap” gesture - the hypothesis regarding the equality of the gesture completion average time was rejected. Nevertheless, for average precision and error probability in “double tap” gesture completion, there were no grounds for rejecting the hypotheses regarding the equality of these parameters among participants using and not using smartphones on an everyday basis.

## **6. “Flick” gesture test results**

During the “flick” gesture tests, the object subjected to manipulations was the same as the one used for the “tap” gesture tests (compare Fig. 4). The square was displayed in randomly selected positions on the smartphone screen.

“Flick” gesture measurements concerned only time, so the values measured for a tested participant were:

$t_a$  – object (square) displaying time;

$t_b$  – time in which the participant removed their finger from the screen.

After the measurements were taken, the average time for “flick” gesture completion was determined (compare formula [11]). Gesture completion time standard deviation and confidence ranges for gesture completion time in an age group were determined. See Table 12 for the calculation results regarding the “flick” gesture completion average time.

**Tab. 12.** Average time  $\bar{t}_1^g$  for “flick” gesture completion in given age group  $g$

Age bracket [years of age]	16-24	25-34	35-44	45-54	$\geq 55$
Group $g$	1	2	3	4	5
Average “flick” gesture completion time $\bar{t}_1^g$ [ms]	714.0	706.0	741.8	1017.5	1093.3

See Figure 18 for the average time, standard deviation of time and time confidence ranges for “flick” gesture completion the in given age groups.

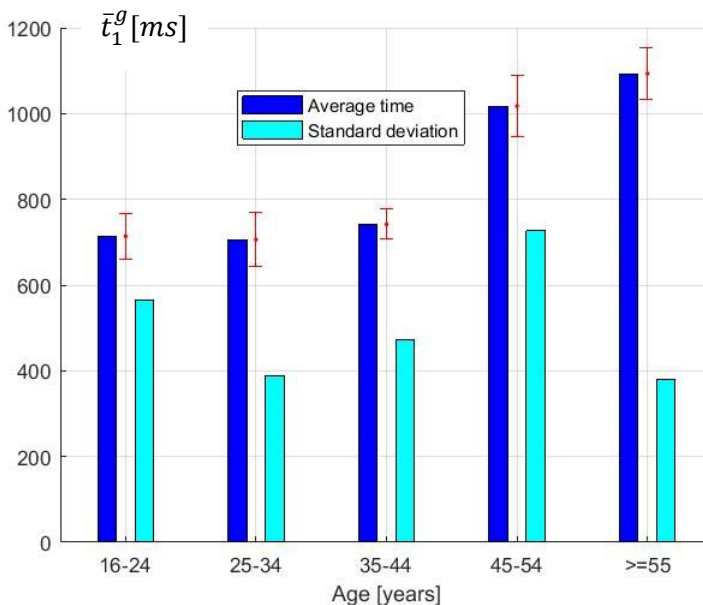
Test results (“flick” gesture completion average time) among age groups were compared, in particular between the first group and the other groups.

The following hypotheses regarding “flick” gesture completion average time were formulated:

H0: equal average time for age groups 1 and  $j$  ( $\bar{t}_1^1 = \bar{t}_1^j$ );

H1: different average time for age groups 1 and  $j$  ( $\bar{t}_1^1 \neq \bar{t}_1^j$ ).

The hypotheses were tested at the significance level of  $\alpha = 0.05$ , and verification results are given in Table 13.



**Fig. 18.** Average “flick” gesture completion time  $\bar{t}_1^g$ , standard deviation and confidence ranges for gesture completion in the given age groups

**Tab. 13. Results of comparison between average time of “flick” gesture completion between group 1 and the other age groups**

Groups subject to comparison	1-2	1-3	1-4	1-5
Decision on H0	No grounds to reject H0	No grounds to reject H0	Reject H0	Reject H0

The following designations were assumed:

$\bar{t}_1^u$  – average time of “flick” gesture completion by users using smartphones on an everyday basis;

$\bar{t}_1^n$  – average time of gesture completion by users not using smartphones on an everyday basis;

On the basis of the results obtained, the average “flick” gesture completion time within the group of participants using smartphones on an everyday basis and the group of participants not using smartphones on an everyday basis was determined. The results are shown in Figure 19.

It was also checked whether the fact that participants use smartphones on an everyday basis influenced the quality of “flick” gesture completion. The following hypotheses were formulated:

H0: equal average time ( $\bar{t}_1^u = \bar{t}_1^n$ ),

H1: different average time ( $\bar{t}_1^u \neq \bar{t}_1^n$ ).

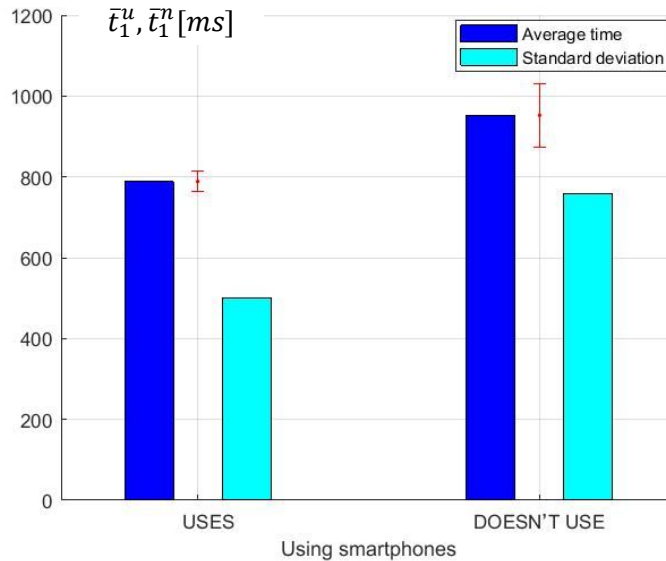
The hypothesis was tested at the probability level  $\alpha = 0.05$ , and the verification result indicated that hypothesis H0 must be rejected, to the advantage of an alternative hypothesis.

### Comments and conclusions regarding test results

On the basis of the results obtained, one can observe fairly significant standard deviation values for the average “flick” gesture completion time, regardless of the age group (compare Fig. 18).

While comparing the average time for “flick” gesture completion among age groups, it can be concluded that differences between the first and fourth as well as first and fifth age group (for larger age difference among the participants) are significant.

It was also concluded that differences concerning the average time for “flick” gesture completion are significant, taking into account persons using and not using smartphones on an everyday basis.



**Fig. 19.** Average time, standard deviation of time and precision confidence ranges for “flick” gesture completion time for participants who use and do not use smartphones on an everyday basis

## 7. Summary of conclusions

This paper constitutes the first part of a broader study concerning the results of tests examining the quality of test participants’ operation while using a smartphone (and/or tablet) and indicating objects on the screen using certain gestures. The tests were performed using an application based on recommendations provided in the ISO 9241-9 standard [14]. The test results presented in the paper concern “tap”, “double tap” and “flick” gestures. Gesture completion time and gesture completion precision (distance between the screen point being touched with a finger and the centre of the object being touched) were used as the basic quality characteristics.

While comparing the test results with data available in the literature, it can be observed that, as far as the “tap” gesture is concerned, the age of the test participants had a significant impact on precision (compare Tab. 6 and the conclusions in paper [3]). The partial confirmation regards the gesture completion average time, and one must remember that in paper [3], the “tap” gesture tests were performed differently from the tests performed by us.

The impact of the test participants' ages on the input characteristics subject to comparison is clearly visible. The older the users are, the more significantly slow and less precise gesture completion is (compare Tab. 6). Taking into account test participants using and not using smartphones on an everyday basis, this characteristic has a significant impact only on the average time of "tap" gesture completion, however, it has no significant influence on the average gesture precision or error probability.

As for the "double tap" gesture, the participants' ages exerted no unequivocal impact on the characteristics. A significant impact is only visible for the gesture completion average time; however, it does not concern the comparison between the first and the second age group. Thus the conclusion here is the same as for the "tap" gesture. The results of the comparison of the gesture precision, gesture completion average time and gesture completion error probability for the "double tap" gesture in the case of test participants using and not using smartphones on an everyday basis are the same as for the "tap" gesture - the hypothesis regarding the equality of the gesture completion average time was rejected.

While comparing the average precision of "tap" and "double tap" gesture completion in age groups, slight differences that do not exceed 2.5 pixels, apart from the first age group, can be observed. While comparing the significance of the average precision for "tap" and "double tap" gestures, it was determined that, for the first age group, the hypothesis regarding the equality of average precision of "tap" and "double tap" gestures must be rejected. As for the other groups, there are no grounds for rejecting the hypothesis regarding the equality of average precision of "tap" and "double tap" gestures.

Taking into account the average time of "tap" and "double tap" gesture completion in various age groups, these differences do not exceed 90 ms, apart from the oldest age group, where this difference exceeds 200 ms. While studying the significance of average completion time for (all) "tap" and "double tap" gestures, it was determined that, for the first, third and fifth age group, the hypothesis regarding the equality of average time of "tap" and "double tap" gesture completion must be rejected. As for the other groups, there are no grounds for rejecting the hypothesis regarding the equality of average time of "tap" and "double tap" gesture completion. Exactly the same results of the tests regarding the significance of average time for "tap" and "double tap" gesture completion were obtained, taking into account only correct gestures.

Taking into account the "flick" gesture and its average completion time, significant differences between the first and fourth, and the first and fifth age groups were determined (significant differences at a higher age difference between the tested groups). No significant differences for the other age groups were identified.

The next paper will present the results of tests examining the quality of test participants' operation using a smartphone, taking into account gestures different from the ones presented in this paper.

## Literature

- [1] DONIGIEWICZ A.M., *Modelowanie interakcji człowiek-komputer. Problemy oceny jakości i niezawodności*. Wyd. WAT, Warszawa, 2005.
- [2] DONIGIEWICZ A.M., *Badanie czasu wskazywania obiektu przez użytkownika testem wielokierunkowym. Część II – wyniki z lat 2015-2016*. Przegląd Teleinformatyczny, nr 1-2, 2017, s. 13-39.
- [3] HOURCADE J.P., BERKEL T.R., *Simple pen interaction performance of young and older adults using handheld computers*. Interacting with Computers, 20, 2008, pp. 166-183.
- [4] FUJIOKA R., AKIBA T., OKADA H., *Evaluation of Pointing Efficiency on Small Screen Touch User Interfaces*. In: Human Interface and the Management of Information. Information and Interaction. Human Interface, Part II, HCII 2009, LNCS 5618, 2009, pp. 375-384.
- [5] COCKBURN A., AHLSTRÖM D., GUTWIN C., *Understanding performance in touch selections: Tap, drag and radial pointing drag with finger, stylus and mouse*. Int. J. Human-Computer Studies, 70, 2012, pp. 218-233.
- [6] BI X., LI Y., ZHAI S., *FFitts Law: Modeling Finger Touch with Fitts' Law*. CHI'13, Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2013, pp. 1363-1372.
- [7] LAI C.C., WU C.F., *Display and device size effects on the usability of mini-notebooks (netbooks)/ultraportables as small form-factor Mobile PCs*. Applied Ergonomics, 45, 2014, pp. 1106-1115.
- [8] MACKENZIE I.S., *Fitts' throughput and the remarkable case of touch-based target selection*. Human-Computer Interaction, Part II, HCII 2015, LNCS 9170, 2015, pp. 238-249.
- [9] CONTI A.S., KREMSER F., KRAUSE M., AN D., BENGLER K., *The effect of varying target sizes and spaces between target and non-target elements on goal-directed hand movement times while driving*. Procedia Manufacturing, 3, 2015, pp. 3168-3175.
- [10] YU C., WEN H., XIONG W., BI X., SHI Y., *Investigating effects of post-selection feedback for acquiring ultra-small targets on touchscreen*. Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, 2016, pp. 4699-4710.

- [11] TSAI T.H., TSENG K.C., CHANG Y.S., *Testing the usability of smartphone surface gestures on different sizes of smartphones by different age groups of users*. Computers in Human Behavior, 75, 2017, pp. 103-116.
- [12] FARHAD M., MACKENZIE I.S., *Evaluating tap-and-drag: A single-handed zooming method*. In: Human-Computer Interaction. Interaction Technologies, HCI 2018, LNCS 10903, 2018, pp. 233-246.
- [13] WAWRYNIUK R., *Metodyka oceny jakości działania użytkownika urządzenia mobilnego*. Praca dyplomowa, WAT, Warszawa, 2013.
- [14] *Ergonomic requirements for Office work with visual display terminals (VDTs). Part 9: Requirements for non-keyboard input devices*. ISO 9241-9:2000(E), International Organization for Standardization.
- [15] KOWALSKI L., *Statystyka*. Wyd. Wydział Cybernetyki WAT, BelStudio, Warszawa, 2005.
- [16] CIECIURA M., ZACHARSKI J., *Metody probabilistyczne w ujęciu praktycznym*. Wizja Press & IT, Warszawa, 2007.

### **Online sources**

- [17] <http://www.mathworks.com/>, <http://www.ont.com.pl/>
- [18] <http://statystyka.rezolwenta.eu.org/materialy.html>
- [19] <http://gsmowo.pl/nokia-lumia-800/>
- [20] <https://tech.wp.pl/nokia-lumia-800-6039436541907585c>

## **Badanie jakości działania użytkownika wykorzystującego urządzenie mobilne**

### **Gesty tap, double tap i flick**

**STRESZCZENIE:** W artykule przedstawiono wyniki badań jakości wykonywania gestów przez użytkowników wykorzystujących urządzenie mobilne. Jako urządzenie mobilne wykorzystano smartfon Nokia Lumia 800. Wyniki badań dotyczą podstawowych gestów tap, double tap i flick i obejmują czas wykonania gestu oraz precyzję wykonania gestu. Wyniki uwzględniają podział użytkowników na grupy wiekowe oraz grupy używające i nie używające smartfona na co dzień. Przedstawiono porównanie wyznaczonych charakterystyk pomiędzy grupami. Badania dotyczące innych gestów zostaną przedstawione w kolejnym artykule.

**SŁOWA KLUCZOWE:** gesty tap, double tap i flick, wprowadzanie gestów palcem, urządzenie mobilne, czas wykonania gestu, precyzja wykonania gestu

*Received by the editorial staff on: 1.12.2018*



**Information for Authors**  
**– rules of papers preparation and reviewing for**  
**TELEINFORMATICS REVIEW**

The *Teleinformatics Review* is devoted to the publication of original research results in fields of science including, but not limited to: computer science, telecommunication, signal processing, network systems, automation and robotics, etc., which have not been published elsewhere in their entirety or considerable part. If a submitted paper is a part of another published work, e.g. a doctoral dissertation, a postdoctoral thesis, etc., the source work should be included in the list of literature and the editorial office must be informed about it.

In order to publish a paper in the Teleinformatics Review it is necessary to submit it to the editorial office in an electronic form (and possibly its printed copy, one-sided, legible, on white A4 sheets) according to the given template. Only original works in English or Polish will be accepted. The text of the paper should be prepared in the format of Microsoft Word editor (versions 2003 or 2010 are suggested). Appropriate templates can be downloaded from website [review.ita.wat.edu.pl](http://review.ita.wat.edu.pl) (or [przeglad.ita.wat.edu.pl](http://przeglad.ita.wat.edu.pl)). The electronic version submitted to the editorial office should contain a source file of the paper in DOC or DOCX format, with all figures and tables being inserted. The editorial office does not rewrite the text neither make drawings. In addition to the mentioned source file, all figures should be delivered in commonly used image formats (preferably as EPS, JPG, TIFF, or others).

Papers to be published in the Teleinformatics Review are subject to initial acceptance by the editorial office and then are subject to review by two external reviewers. Reviewers and authors do not know each other personal data. The content of the review will be available at the editorial office. If one review is negative (or imprecise) then a third reviewer may be appointed. If both reviews are negative the paper is rejected. If the review indicates a necessity of some corrections, the author must consider all of them and resubmit the improved paper by the determined deadline.

The volume of a submitted paper generally not exceed 20 pages of typescript A4. A deviation from this rule requires agreement of the editorial office. Except the last page, no more than 10% of any page within the paper can be left empty. Figures must be numbered and described below them as well as tables must be numbered and described at the top of them. The literature should hold the form given in the template.

The authors are obliged to submit a statement to the editorial office on the percentage contribution to the creation of the accepted paper, confirming the lack of prior publication of such a work, or a public speech on the subject at a conference or symposium.

The editorial board reserves rights to introduce minor editorial changes to the content of paper without consulting the author. The editorial office insists that no special formatting should be used, which would be inconsistent with the template.

Papers printed in the Teleinformatics Review and their abstracts are placed in the national database of Polish technical journals BazTech as well as on the INDEX COPERNICUS website. Additionally, the papers will be available in the electronic PDF form on website [review.ita.wat.edu.pl](http://review.ita.wat.edu.pl).

Publication in the Teleinformatics Review does not involve any costs for authors. The editorial office does not charge for submitting, reviewing, preparing for publication and publishing the work. The publication of a paper in the Teleinformatics Review is tantamount to transfer of authors' property rights for publication to the publisher, i.e. the Military University of Technology. By submitting a paper for publication in the Teleinformatics Review, the author agrees, for publication purposes, to the processing by the editorial office the author's name, email address, affiliation, and other contact details.



All papers published in the journal **TELEINFORMATICS REVIEW** are made available under the Creative Commons Attribution-NonCommercial-NoDerivatives 3.0 (CC BY-NC-ND 3.0) license. Thus, licensees may copy, distribute, display, and perform the work and make derivative works and remixes based on it only for non-commercial purposes; licensees may copy, distribute, display and perform only verbatim copies of the work, not derivative works and remixes based on it.

The editorial office does not return received materials.

The editorial office does not pay fees for papers publishing.

The editor-in-chief may refuse to publish a paper in the following cases:

- if the content of the paper violates the law (principles of secrecy protection, press law, copyright law, etc.) or good manners;
- the author does not agree to introduce all necessary corrections proposed by the editorial board or reviewers;
- the text and illustrative material submitted by the author does not meet the technical requirements given in this document or the template.

**Informacje dla autorów**  
**– zasady przygotowania tekstu i recenzowania artykułów do**  
**PRZEGLĄDU TELEINFORMATYCZNEGO**

W Przeglądzie Teleinformatycznym zamieszczane są oryginalne artykuły z dziedzin: *informatyka, telekomunikacja, przetwarzanie sygnałów, systemy sieciowe, automatyka i robotyka* oraz pokrewnych, niepublikowane dotychczas w całości lub w znaczącej części. Jeśli nadesłana praca stanowi część innej opublikowanej pracy, np. pracy doktorskiej, habilitacji, etc., to źródło powinno być umieszczone w spisie literatury, a redakcja powinna być o tym poinformowana.

W celu opublikowania artykułu w *Przeglądzie* niezbędne jest dostarczenie do redakcji treści artykułu w postaci **elektronicznej** według podanego szablonu i ewentualnie jednego egzemplarza wydrukowanego (jednostronnie, czytelnie, na białym papierze formatu A4). Przyjmowane są tylko oryginalne prace w języku angielskim lub polskim. Tekst artykułu powinien być przygotowany w formacie edytora Microsoft Word (wersja 2003 lub 2010 jest zalecana). Szablony dla artykułów są dostępne w pliku na stronie [przeglad.ita.wat.edu.pl](http://przeglad.ita.wat.edu.pl) (lub [review.ita.wat.edu.pl](http://review.ita.wat.edu.pl)). Przekazane do redakcji materiały powinny zawierać plik źródłowy w formacie DOC lub DOCX, ze wstawionymi rysunkami. Redakcja nie przepisuje tekstów i nie wykonuje rysunków. Dodatkowo należy dostarczyć pliki źródłowe rysunków (najlepiej w formacie EPS, JPG, TIFF lub innym powszechnie używanym).

Artykuły przeznaczone do opublikowania w *Przeglądzie* podlegają wstępnej ocenie przez redaktora działu, a następnie recenzji dwóch zewnętrznych recenzentów. Recenzenci i autorzy nie znają swoich danych personalnych. Z treścią recenzji można zapoznać się w redakcji. Jeśli jedna z recenzji jest negatywna (lub nieprecyzyjna), może być powołany trzeci recenzent. Jeśli dwie recenzje są negatywne, artykuł jest odrzucany. Jeśli z recenzji wynika konieczność dokonania poprawek w treści artykułu, to autor jest zobowiązany do ich rozpatrzenia i dostarczenia do redakcji poprawionej wersji artykułu, w terminie ustalonym przez redakcję.

Objętość artykułu zasadniczo nie powinna przekroczyć 20 stron maszynopisu A4. Odstąpienie od tej zasady wymaga uzgodnień z redakcją *Przeglądu*. Na stronach tekstu artykułu nie może być pozostawione więcej niż 10% pustego miejsca, za wyjątkiem ostatniej strony. Rysunki należy numerować i opatrzyć (pod spodem) wyczerpującym podpisem. Tabele również muszą być numerowane (tytuł nad tabelą). Literatura może być uszeregowana alfabetycznie oraz powinna mieć postać jak w szablonie.

Autorzy są zobligowani do złożenia w redakcji oświadczenia autorskiego o wkładzie procentowym w powstanie artykułu, braku wcześniejszej publikacji artykułu w przedstawionej formie lub wystąpieniu publicznym na ten temat na konferencji lub sympozjum.

Redakcja zastrzega sobie prawo wprowadzenia niewielkich redakcyjnych zmian w treści artykułu bez konsultacji z autorem. Redakcja nalega, aby **nie stosować** żadnego specjalnego formatowania i **trzymać się ściśle** ustaleń zawartych w szablonie.

Streszczenia i pełne teksty artykułów drukowanych w *Przeglądzie* zamieszczane są w krajowej bazie danych o zawartości polskich czasopism technicznych BazTech oraz na platformie INDEX COPERNICUS. Opublikowane w *Przeglądzie* artykuły będą także w całości udostępnione w internetowej wersji (format PDF) czasopisma, pod adresem [przeglad.ita.wat.edu.pl](http://przeglad.ita.wat.edu.pl) (lub [review.ita.wat.edu.pl](http://review.ita.wat.edu.pl)).

Publikacja w *Przeglądzie* nie wiąże się z żadnymi kosztami dla autorów. Redakcja nie pobiera opłat za zgłoszenie, przygotowanie do druku, recenzję czy publikację pracy. Przekazanie artykułu do publikacji w *Przeglądzie* jest równoznaczne z przekazaniem autorskich praw majątkowych do publikacji na rzecz wydawcy, tj. Wojskowej Akademii Technicznej. Przekazując artykuł do publikacji w *Przeglądzie*, autor zgadza się na przechowywanie i przetwarzanie przez redakcję, w celach publikacyjnych, imienia, nazwiska, adresu e-mail i afiliacji.



Wszystkie artykuły opublikowane w czasopiśmie **PRZEGLĄD TELEINFORMATYCZNY (TELEINFORMATICS REVIEW)** są udostępniane na licencji Creative Commons Uznanie autorstwa – Użycie niekomercyjne – Bez utworów zależnych 3.0 (CC BY-NC-ND 3.0), która zezwala na kopiowanie, przedstawianie i rozpowszechnianie utworu jedynie w celach niekomercyjnych oraz pod warunkiem zachowania go w oryginalnej postaci (czyli nietworzenia utworów zależnych), przy jednoczesnym odpowiednim oznaczeniu autorstwa utworu.

Redakcja nie zwraca materiałów dostarczonych do redakcji.

Redakcja nie przewiduje honorariów za opublikowanie artykułu.

Redaktor naczelny może odmówić opublikowania artykułu w przypadku, gdy:

- treści zawarte w materiałach naruszają prawo (zasady ochrony tajemnicy, prawo prasowe, prawo autorskie itp.) lub dobre obyczaje;
- autor nie zgadza się na wprowadzenie wszystkich koniecznych poprawek zaproponowanych przez redakcję lub recenzentów;
- tekst i materiał ilustracyjny złożony przez autora nie spełnia wymagań technicznych podanych w niniejszym dokumencie i szablonie.