МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Одеська політехніка»

# ІНФОРМАТИКА ТА МАТЕМАТИЧНІ МЕТОДИ В МОДЕЛЮВАННІ

# INFORMATICS AND MATHEMATICAL METHODS IN SIMULATION

Том 13, № 1-2

Volume 13, No. 1-2

Одеса – 2023
Odesa – 2023

Журнал внесений до переліку наукових фахових видань України (технічні науки) згідно наказу Міністерства освіти і науки України № 463 від 25.04.2013 р. Перереєстровано на категорію «Б» за фахами 121, 122, 125, 151 згідно наказу МОН України № 1473 від 26.11.2020 р.

**Виходить** 4 рази на рік

**Заснований** Одеським національним політехнічним університетом у 2011 році

**Свідоцтво** про державну реєстрацію КВ № 17610 - 6460Р від 04.04.2011р.



**Published** 4 times a year

**Founded** by Odesa National Polytechnic University in 2011

**Certificate** of State Registration KB № 17610 - 6460P of 04.04.2011



**Друкується** за рішенням редакційної колегії та Вченої ради Національного університету «Одеська політехніка»

**Оригінал-макет** виготовлено редакцією журналу

**Адреса редакції:** просп. Шевченка, 1, Одеса, 65044, Україна
Телефон: +38 048 705 8506
Web: www.immm.op.edu.ua (immm.opu.ua) E-mail: immm.ukraine@gmail.com

**Editorial address:** 1 Shevchenko Ave., Odesa, 65044, Ukraine
Tel.: +38 048 705 8506
Web: www.immm.op.edu.ua (immm.opu.ua) E-mail: immm.ukraine@gmail.com

## ЗМІСТ/CONTENTS

DOI 10.15276/imms.v13.no1-2.5
UDC 004.043

Informatics and Mathematical Methods in Simulation
Vol. 13 (2023), No. 1-2, pp. 5-15

# HIERARCHICAL CLUSTERING ALGORITHM FOR DENDROGRAM CONSTRUCTION AND CLUSTER COUNTING

N.I. Boyko, O.A. Tkachyk

Lviv Polytechnic National University, Knyaz Roman Str. 5 Lviv, 79013; Ukraine; nataliya.i.boyko@lpnu.ua; oleksandr.a.tkachyk@lpnu.ua

The article provides a comprehensive overview of hierarchical clustering and dendrogram construction, with a focus on the methods used for determining the optimal number of clusters. The article discusses the theoretical foundations of hierarchical clustering and the process of constructing dendrograms, and goes on to describe several popular methods for determining the number of clusters. The article focused on both divisive and agglomerative clustering methods and the dendrogram, the advantages and disadvantages of each method, and how dendrograms are used to visualize the results of hierarchical clustering. It also provides comparison of hierarchical clustering with non-hierarchical clustering, particularly the K-means algorithm, and discusses their respective advantages and disadvantages. One of the key advantages of hierarchical clustering is that it does not require the user to specify the number of clusters in advance, as is the case with non-hierarchical clustering. Instead, a dendrogram can be used to determine the appropriate number of clusters. The article concludes by noting the usefulness of hierarchical clustering for a range of applications, particularly in exploratory data analysis. The article also covers the main methods to identify which objects and clusters are most similar. Additionally, the article provides an overview of the K-means clustering method and compares it to hierarchical clustering.

**Keywords:** agglomerative clustering, divisive clustering, hierarchical clustering, k-means.

**Introduction.** Cluster analysis has proven to be an invaluable tool in the context of multidimensional datasets and its utility in research and uncontrolled analysis. It highlights the hierarchical approach to clustering as a popular method in genomics and other fields due to its ability to reveal multiple layers of clustering structure simultaneously. This allows researchers to gain a deeper understanding of the data, identify patterns, and develop hypotheses or inform decision-making. Many applied problems, measuring the degree of similarity between objects is often much simpler than forming descriptive features. For example, taking two photos and immediately identifying that they both depict the same person is much easier than understanding the specific features that make them similar. The task of object classification based on their similarity, without any predefined classes to which the objects can be assigned, is called clustering [2, 15].

Hierarchical clustering is a popular technique used for data analysis and pattern recognition. It is a method that groups data objects based on their similarities and differences. This technique can be used to explore the underlying structure of a dataset, identify relationships between variables, and discover patterns in data [12, 20].

Hierarchical clustering algorithms can be divided into two main types: divisive and agglomerative. Divisive clustering starts with a single cluster containing all data objects and divides it into smaller clusters until each object is in its own cluster [1,3]. On the other hand, agglomerative clustering starts with each object in its own cluster and merges them together until all objects belong to a single cluster [7]. One of the most important outputs of hierarchical clustering is the dendrogram. A dendrogram is a tree-like diagram that illustrates the hierarchical relationships between clusters. It displays the

order in which clusters are merged and the distances between them. This diagram helps to visualize the hierarchical structure of the data and provides insights into the relationships between clusters [5, 11].

**Content statement of the problem.** The purpose of study is to apply different algorithms for constructing dendrograms and determining the optimal number of clusters in hierarchical clustering. The study aims to provide a theoretical background of hierarchical clustering, explain the process of dendrogram construction, and discuss the different methods for determining the number of clusters.

The object of this study is to analyze different clustering algorithms for constructing dendrograms and determining the number of clusters in hierarchical clustering. The study focuses on the methodology and practical aspects of hierarchical clustering, including different types of linkage methods, distance metrics, initialization of clusters, and scalability issues. However, the study also compares several non-hierarchical clustering algorithms such as k-means. Therefore, the object of the study includes both hierarchical and non-hierarchical clustering algorithms, and aims to provide a comprehensive and practical guide to different clustering methods for researchers and practitioners in the field. The study aims to provide guidance and insights into the process of dendrogram construction and the determination of the optimal number of clusters. The object of the study is the algorithmic approach to hierarchical clustering and its applications in data analysis and research.

**Analysis of recent resources.** The paper [1] by M. Kuchaki Rafsanjani is a research paper that provides an extensive overview of the various hierarchical clustering algorithms. The paper discusses the different types of hierarchical clustering, such as agglomerative and divisive clustering, and the pros and cons of each type. The paper includes a comparative analysis of several hierarchical clustering algorithms, including Ward's method, k-means clustering, and spectral clustering, and provides insights into the strengths and weaknesses of each algorithm. The author concludes the paper by discussing some of the current research directions in hierarchical clustering, such as semi-supervised clustering and graph-based clustering, and highlighting the potential applications of hierarchical clustering in various fields, including bioinformatics and image analysis [10, 13].

In the research [2, 17] Luben M. C. Cabezas discusses different approaches to dendrogram construction and visualization. Also, the author explores the use of dendrograms for interpreting machine learning models and extracting insights from large datasets.

Research [3, 19] is interesting by its discussion of different methods for determining the number of clusters in hierarchical clustering, including the silhouette method and the elbow method. The paper also covers other important aspects of k-means clustering, such as the choice of distance metrics and initialization of clusters, and provides insights into the strengths and weaknesses of different methods.

The article [4, 20] provides a survey of agglomerative clustering algorithms and their applications in high-dimensional data. Authors describe and present a comprehensive classification of different clustering techniques for high dimensional data. The article covers other important aspects of clustering validation, such as the choice of distance metrics and the interpretation of validation results. Furthermore, the paper presents experimental results and case studies to illustrate the effectiveness and limitations of different clustering validation measures.

**Methods and tools of research.** The approach of creating a dendrogram, which is a type of tree diagram that shows the arrangement of clusters produced by a clustering algorithm, using specific algorithms that are intended for this purpose. These algorithms take the data input and produce a hierarchical structure of clusters that can be visualized in the form of a dendrogram [6, 8].

Hierarchical clustering is a type of cluster analysis. One of the great advantages of hierarchical algorithms over non-hierarchical ones is their hierarchical structure, which is created during the algorithm's operation. The operation of such an algorithm can be represented as a dendrogram (Fig. 1) or, as they are also called, a tree diagram. In the dendrogram, each level corresponds to one iteration of the algorithm [9, 14].

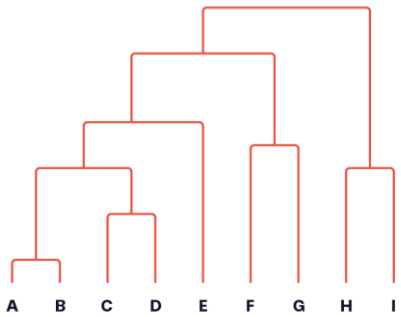



**Fig. 1**. Dendrogram example

The number of clusters can either decrease or increase with each iteration, depending on the type of hierarchical clustering algorithm used.

**Types of hierarchical clustering.** As already mentioned, there are two main types of hierarchical clustering. Each of them moves in a different direction. These are agglomerative and divisive algorithms. In the first case, we start with each object of the study having its own cluster, and with each iteration, clusters begin to merge until all objects end up in one cluster. With the divisive algorithm, on the other hand, everything is the opposite. Initially, we have one large cluster that includes all objects of our study, and with each iteration, we begin to divide this cluster into smaller ones, which ultimately leads to each object having only its own cluster [16, 18]. Approximate working of both algorithms is shown in Figure 2.

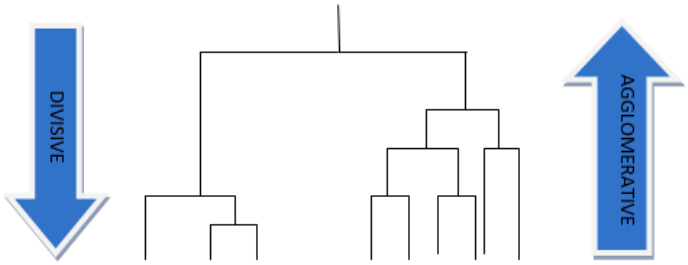



**Fig. 2.** Agglomerative and divisive algorithms

**Agglomerative hierarchical clustering.** Let's take a closer look at the agglomerative method. As a dataset, we will take several coordinates X and Y described in table 1 to demonstrate the operation of the agglomerative method.

**Table 1**

The objects under study

| № | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| x | 6 | 4.9 | 8.2 | 7.1 | 2 | 1 | 1.5 | 2.8 | 3 | 6.9 | 6.1 | 8 | 7.1 |
| y | 0.6 | 3 | 2.1 | 3.8 | 6 | 7.8 | 8.3 | 7 | 7.9 | 6.9 | 8.2 | 7.9 | 8.8 |

First, we have our objects of study, which are shown in Figure 3.

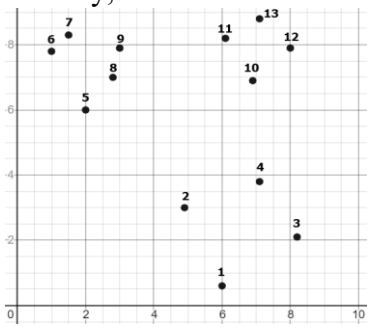



**Fig. 3**. Study objects represented on the coordinate plane

At the beginning of the algorithm, all of our objects form their own clusters (Fig. 4).

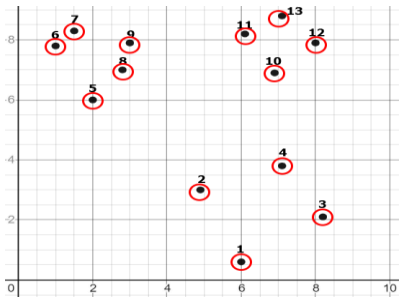



**Fig. 4**. Initial clusters

Now each point is in its own cluster. The next step is to merge clusters with the smallest distance between them. In our case, these turned out to be clusters 6 and 7, as their Euclidean distance for X and Y are:

$$d = \sqrt{(1.5-1)^2 + (8.3-7.8)^2} = \sqrt{0.5} \approx 0.7071 \quad (1)$$

Now we have a new cluster number 14 (Fig. 5).

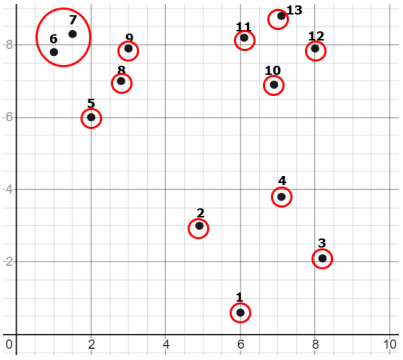



**Fig. 5**. Points 6 and 7 forms a new cluster

Continuing to search for clusters with the smallest distances, we find that clusters 8 and 9, 11 and 13, and 10 and 12 have the smallest distances, and they form new clusters. Data is provided in table 2 and visualized on Figure 6.

**Table 2**

Cluster combination

| Clusters A & B | Cluster A coordinates | Cluster B coordinates | Distance between clusters | New cluster number |
|---|---|---|---|---|
| 8 and 9 | {2.8, 7} | {3, 7.9} | 0.9219 | 15 |
| 11 and 13 | {6.1,8.2} | {7.1,8.8} | 1.1661 | 16 |
| 10 and 12 | {6.9,6.9} | {8,7.9} | 1.4866 | 17 |

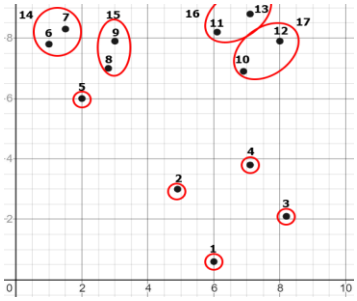


**Fig. 6**. New cluster formation

At this stage, our dendrogram looks as on Figure 7.

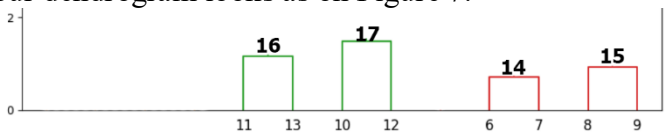



**Fig. 7**. Updated dendrogram based on recent updates

Now we can see that the clusters closest to each other are not single-point clusters, but a cluster with multiple objects and a single-point cluster. In this case, we need a measure to compute the distances between clusters with multiple points. We will continue merging clusters using Complete Linkage. The results are seen in Table 3 and Figure 8.

**Table 3**

Cluster combination

| Clusters A & B | Distance between clusters | New cluster number |
|---|---|---|
| 16 and 17 | 1.9235 | 18 |
| 14 and 15 | 2.0024 | 19 |

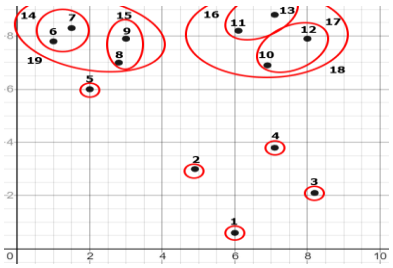


**Fig. 8.** Newly formed clusters

Continuing this algorithm, we will eventually end up with all objects in one cluster. Let's try to finish the work using Python. After performing agglomerative clustering using the Complete Linkage method, which we will discuss later, I obtained the following data array displayed on Figure 9.

```
array([[ 6,  7,  0.70710678, 2, 14],
       [ 8,  9,  0.92195445, 2, 15],
       [11, 13,  1.16619038, 2, 16],
       [10, 12,  1.48660687, 2, 17],
       [16, 17,  1.92353841, 4, 18],
       [14, 15,  2.00249844, 4, 19],
       [ 3,  4,  2.02484567, 2, 20],
       [ 5, 19,  2.35372046, 5, 21],
       [ 1,  2,  2.64007576, 2, 22],
       [20, 22,  3.42052628, 4, 23],
       [18, 21,  7.00071425, 9, 24],
       [23, 24,  9.18313672, 13, 25]])
```

**Fig. 9**. The array obtained using Python

The first two columns represent the cluster numbers that will be merged. The third column represents the distance between them, the fourth column indicates the number of objects that will be in the new cluster, and the last column represents the number of the newly created cluster. Indeed, if we look at this table, its first half completely matches the data that we calculated above (Fig. 10).

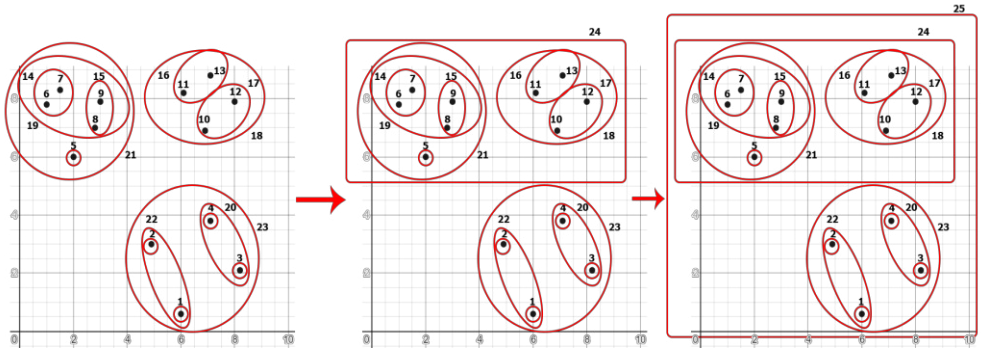


**Fig. 10.** The last iterations of the agglomerative hierarchical clustering method

Now all research objects are in the same cluster with number 25. It may seem illogical as it does not give us any meaningful data. However, in fact, it is quite the opposite. Since the algorithm is hierarchical, we can obtain data that was obtained at a certain iteration. This can be visualized more clearly by looking at the result displayed in the dendrogram (Fig. 11).

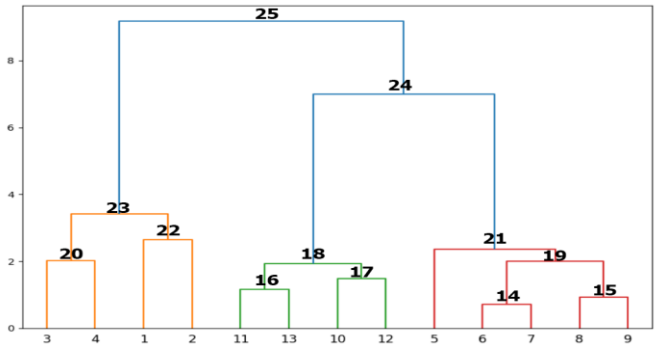



**Fig. 11**. Newly formed dendrogram based on coordinates

Now we have obtained a dendrogram that shows all the iterations of creating new clusters. Our dendrogram also shows that indeed points 6 and 7 were the first to merge, then 8 and 9, and 11 and 13. There are also three strongly pronounced clusters 23, 18, and 21. Let's visualize them on a plot by coloring the points belonging to these clusters in different colors (Fig. 12).

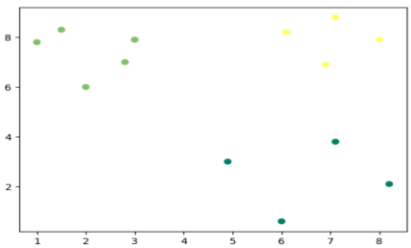


**Fig. 12.** The three clusters obtained using the agglomerative method

We can also notice that cluster 23 is much taller than the other two. This is due to the fact that the distances between points in this cluster are much larger, making the cluster itself larger in size (Fig. 13).

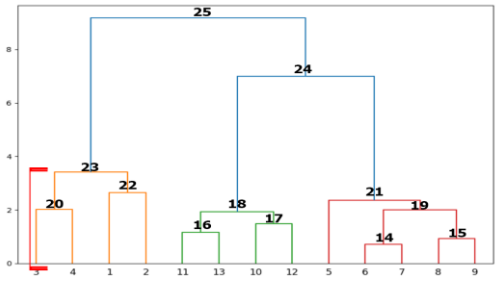



**Fig. 13**. The height of cluster 23

**Divisive clustering method.** In divisive or divisive clustering, everything happens the opposite way. First, we have a cluster that contains all the objects we are studying (Fig. 14).

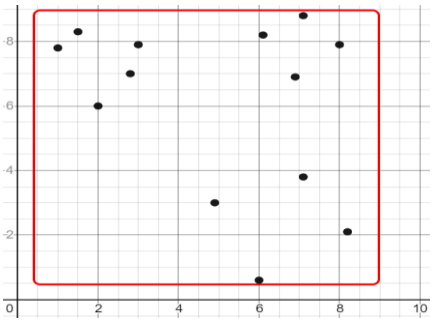


**Fig. 14**. Beginning of divisive clustering algorithm method

Immediately after the first iteration, our large cluster is divided into smaller clusters, and those in turn into even smaller ones. This process continues until all objects end up in their own clusters (Fig. 15).

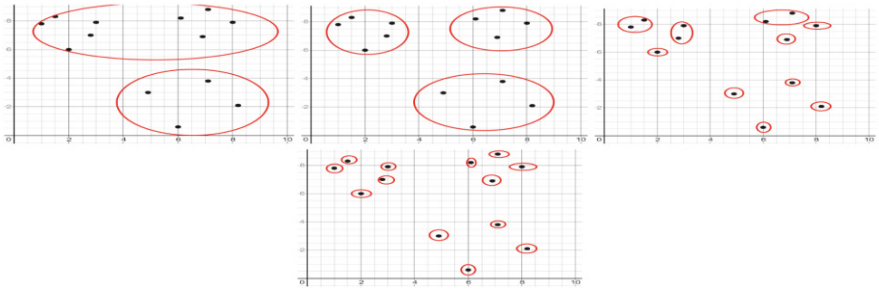


**Fig. 15**. Results of divisive clustering algorithm method

**The recalculation of distance between clusters.** To determine which clusters to merge, we need a metric that will measure the similarity between clusters. There are five main methods to identify which objects and clusters are most similar.

Single Linkage is a method that begins by finding the two closest objects that form the primary cluster. Each subsequent object is then added to the cluster that is closest to one of its objects (Formula 2).

$$d_{min}(C_i, C_j) = min_{x_i \in C_i, x_i, \in C_i} d(x_i, x_j) \quad (2)$$

Complete Linkage, also known as the maximum linkage method, is the inverse of Single Linkage. The rule for combining clusters in this method is based on finding the two objects that are furthest apart from each other (Formula 3).

$$d_{max}(C_i, C_j) = max_{x_i \in C_i, x_i, \in C_i} d(x_i, x_j) \quad (3)$$

Advantage Linkage – at each step, the average distance between each object from one cluster and each object from another cluster is calculated. An object is assigned to a given cluster if the average distance is smaller than the average distance to any other

cluster (Formula 4).

$$d_{avg}(C_i, C_j) = \frac{1}{|C_i||C_j|} \sum_{x_i \in C_i} \sum_{x_i \in C_i} d(x_i, x_j) \quad (4)$$

Ward's Method: this method minimizes the variance between all clusters by selecting clusters that result in the smallest increase in overall variance [8].

Centroid Method: this method calculates the distance between the centroids of each cluster.

**Comparison of hierarchical and non-hierarchical clustering.** Before moving on to the topic of cluster count, it would be helpful to understand other clustering methods. Non-hierarchical clustering also has many methods and algorithms, but we will only discuss the K-means algorithm (nearest neighbor method).

Let's suppose there are hypotheses about the number of clusters, let's say N clusters. In this case, the program can be set to N clusters. This is precisely the use case for the K-means method. While in hierarchical clustering, we can choose the number of clusters after the program has finished processing the data, in non-hierarchical clustering, specifically in the nearest neighbor method, we must determine the number of clusters in advance. This is a major drawback of the algorithm because it is not always possible to know or guess how many clusters there may be after processing the data.

In general, K-means is a popular clustering algorithm that has several advantages:

1. Simplicity: The algorithm is easy to understand and implement, making it a popular choice for data analysts and scientists.
2. Scalability: K-means is a relatively fast and efficient algorithm that can handle large datasets.
3. Ability to handle continuous variables: K-means works well with continuous variables, such as age or income, as opposed to categorical variables.
4. Reproducibility: The results of K-means are reproducible, meaning that if you run the algorithm multiple times with the same inputs, you should get the same results every time.

But also, K-means algorithm has its own disadvantages which is provided in the list below:

1. Requires the number of clusters to be specified: One of the main disadvantages of K-means is that it requires the number of clusters to be specified beforehand. This can be a major drawback, especially when the data does not naturally lend itself to a specific number of clusters.
2. Sensitive to initial cluster centers: The final clusters produced by K-means can be highly dependent on the initial random selection of cluster centers. This can lead to suboptimal results if the initial centers are not representative of the data
3. Outliers can heavily influence results: K-means is highly sensitive to outliers in the data. Outliers can heavily influence the position of the cluster centers and lead to suboptimal clustering results.
4. Cannot handle non-linear data: K-means algorithm assumes that the data can be separated into clusters based on linear boundaries. Therefore, it may not perform well on non-linear data.

**Result and discussion.** One of the main advantages of hierarchical clustering is that it does not require prior knowledge of the number of clusters. At the end of the process, we obtain a hierarchy of clusters, which is usually represented as a dendrogram. By analyzing the dendrogram, we can determine the number of clusters. Additionally, the number of clusters depends on how distinct we want the differences between objects in the cluster to be. If more detail is required, then the dendrogram of the research can be truncated somewhere around the 1 mark. If less differentiation between objects in one cluster is

needed or a smaller number of clusters is desired, then the dendrogram can be truncated at the 2.5 mark (Fig. 16).

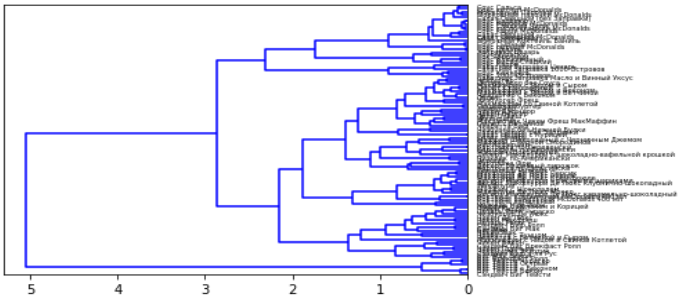



**Fig. 16.** Determining the optimal number of clusters

**Conclusion.** Hierarchical clustering is a popular method for grouping objects into clusters based on their similarity. It involves creating a hierarchy of nested clusters, represented as a dendrogram, which can be used to explore relationships among the objects. There are two main types of hierarchical clustering: agglomerative and divisive. Agglomerative clustering involves starting with each object in its own cluster and iteratively merging the closest clusters until all objects belong to a single cluster. In contrast, divisive clustering starts with all objects in a single cluster and recursively splits them into smaller clusters. While hierarchical clustering does not require prior knowledge of the number of clusters, it can be computationally expensive and may be less effective than other clustering methods for large datasets. K-means clustering, a non-hierarchical method, is often used instead as it requires a prior specification of the number of clusters and is more computationally efficient. However, hierarchical clustering has some unique advantages. The dendrogram can provide insights into the structure of the data and can be used to determine the number of clusters based on the desired level of similarity among objects. Additionally, agglomerative clustering allows for exploration of the hierarchy of clusters, which can be useful in identifying subgroups and relationships among objects. As a conclusion, the choice of clustering method depends on the specific characteristics of the data and the research question. Hierarchical clustering can be a powerful tool for exploratory analysis and uncovering patterns in data.

**References**

1. Kuchaki Rafsanjani M., Asghari Varzaneh Z., Emami Chukanlo N. A Survey of Hierarchical Clustering Algorithms. *TJMCS*. 2012. Vol. 5. No. 3. pp. 229-240. DOI: 10.22436/jmcs.05.03.11
2. Luben M. C. Cabezas, Izbicki R., Stern R. B. Hierarchical clustering: visualization, feature importance and model selection. 2023. URL: https://arxiv.org/pdf/2112.01372.pdf
3. Rathiga P., Selvi P. To determine the optimal number clusters. *Journal of Emerging Technologies and Innovative Research (JETIR)*. 2020. Vol.7, Issue 10. pp. 167-172.
4. Pavithra M.S., Parvathi R. M. A Survey on Clustering High Dimensional Data Techniques. *International Journal of Applied Engineering Research*. 2017. Vol. 12, No. 11. pp. 2893-2899.
5. Ward Jr. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*. 1963. Vol. 58(301). pp. 236–244.
6. Timofeeva A. Evaluating the robustness of goodness-of-fit measures for hierarchical clustering. *Journal of Physics: Conference Series*. 2019. Vol. 1145. pp. 012049.
7. Roux M. A Comparative Study of Divisive and Agglomerative Hierarchical Clustering Algorithms. *Journal of Classification*. 2018. Vol. 35 (2). pp. 345-366. ff10.1007/s00357-018-9259-9ff. ffhal02085844.

8. Murtagh F., Legendre P. Ward's Hierarchical Agglomerative Method: Which Algorithms Implement Ward's Criterion? *Journal of Classification*. 2014. Vol. 31. pp. 274–295.
9. Sarker A., Shamim S.M., Shahiduz Zama Dr. Md., Mustafizur Rahman Md. Employee's performance analysis and prediction using K-means clustering & decision tree algorithm. *Global Journal of Computer Science and Technology*. 2018. Vol. 18. pp. 1-4.
10. Fraley C., Raftery A. E. How Many Clusters? Which Clustering Method? Answers Via Model-Based Cluster Analysis. *Technical Report. Department of Statistics University of Washington*. 1998. No. 329.
11. Murtagh F. A survey of recent advances in hierarchical clustering algorithms which use cluster centers. *Computer Journal*. 2020. vol. 26, no. 4, pp. 354-359.
12. Saxena A., Prasad M., Gupta A., Bharill N., Patel O. P., Tiwari A. & Lin C. T. A review of clustering techniques and developments. *Neurocomputing.* 2017. Vol. 26. p. 664-681.
13. Sneath P., Sokal R. Hierarchical Cluster Analysis: Comparison of Three Linkage Measures and Application to Psychological Data. *The Quantitative Methods for Psychology*. 2015. Vol. 11(1). pp. 8-21. DOI: 10.20982/tqmp.11.1.p008
14. Yim O., Ramdeen K. T. Hierarchical Cluster Analysis: Comparison of Three Linkage Measures and Application to Psychological Data. *The Quantitative Methods for Psychology*. 2015. Vol. 11. no. 1. pp. 8-21. DOI: 10.20982/tqmp.11.1.p008
15. Ptitsyn A., Hulver M., Cefalu W., York D., & Smith S. R. Unsupervised clustering of gene expression data points at hypoxia as possible trigger for metabolic syndrome. *BMC Genomics.* 2016. Vol. 7(1). pp. 318. doi:10.1186/1471-2164-7-318.
16. Tung A.K., Hou J., Han J. Spatial clustering in the presence of obstacles. *The 17th Intern. conf. on data engineering (ICDE'01). Heidelberg*. 2001. pp. 359–367. DOI: 10.1109/ICDM.2002.1184042.
17. Boehm C., Kailing K., Kriegel H., Kroeger P. Density connected clustering with local subspace preferences. *IEEE Computer Society. Proc. of the 4th IEEE Intern. conf. on data mining. Los Alamitos*. 2004. pp. 27–34. DOI: 10.1007/978-0-387-39940-9_605.
18. Boyko N., Kmetyk-Podubinska K., Andrusiak I. Application of Ensemble Methods of Strengthening in Search of Legal Information. *Lecture Notes on Data Engineering and Communications Technologies*. 2021. Vol. 77. pp. 188-200. URL: https://doi.org/10.1007/978-3-030-82014-5_13.
19. Boyko N., Hetman S., Kots I. Comparison of Clustering Algorithms for Revenue and Cost Analysis. *Proceedings of the 5th International Conference on Computational Linguistics and Intelligent Systems (COLINS 2021)*. Lviv, Ukraine. 2021. Vol.1. pp. 1866-1877.
20. Procopiuc C.M., Jones M., Agarwal P.K., Murali T.M. A Monte Carlo algorithm for fast projective clustering. *ACM SIGMOD Intern. conf. on management of data, Madison, Wisconsin, USA.* 2002. pp. 418–427.

# АЛГОРИТМ ІЄРАРХІЧНОЇ КЛАСТЕРИЗАЦІЇ ДЛЯ ПОБУДОВИ ДЕНДРОГРАМИ ТА ПІДРАХУНКУ КЛАСТЕРІВ

Н.І. Бойко, О.А. Ткачик

Національний університет “Львівська політехніка”, вул. Князя Романа, 5, Львів, 79013; Україна; nataliya.i.boyko@lpnu.ua; oleksandr.a.tkachyk@lpnu.ua

Подано вичерпний огляд ієрархічної кластеризації та побудови дендрограми з акцентом на методи визначення оптимальної кількості кластерів. Розглядаються теоретичні основи ієрархічної кластеризації та процес побудови дендрограм, а також описуються кілька популярних методів визначення кількості кластерів. Метою дослідження є застосування різних алгоритмів для побудови дендрограм та визначення оптимальної кількості кластерів в ієрархічній кластеризації. Дослідження має на меті забезпечити теоретичні основи ієрархічної кластеризації, пояснити процес побудови дендрограми та обговорити різні методи визначення кількості кластерів. Об’єкт дослідження включає як ієрархічні, так і неієрархічні алгоритми кластеризації, і має на меті забезпечити вичерпний і практичний посібник із різних методів кластеризації для дослідників і практиків у цій галузі. Робота зосереджена як на методах роздільної, так і на агломераційній кластеризації, а також на дендрограмі, перевагах і недоліках кожного методу, а також на тому, як дендрограми використовуються для візуалізації результатів ієрархічної кластеризації. Він також забезпечує порівняння ієрархічної кластеризації з неієрархічною кластеризацією, зокрема алгоритмом K-середніх, і обговорює їхні відповідні переваги та недоліки. Однією з ключових переваг ієрархічної кластеризації є те, що вона не вимагає від користувача заздалегідь вказувати кількість кластерів, як у випадку з неієрархічною кластеризацією. Зазначається, що для визначення відповідної кількості кластерів можна використовувати дендрограму. В результаті відзначається корисність ієрархічної кластеризації для ряду додатків, зокрема для пошукового аналізу даних. У дослідженні також розглядаються основні методи визначення найбільш схожих об’єктів і кластерів. Крім того, у статті надається огляд методу кластеризації K-середніх і порівнюється його з ієрархічною кластеризацією.
**Ключові слова:** агломеративна кластеризація, роздільна кластеризація, ієрархічна кластеризація, k-середні.

DOI 10.15276/imms.v13.no1-2.16
UDC 004.9

Informatics and Mathematical Methods in Simulation
Vol. 13 (2023), No. 1-2, pp.16-26

# INTELLECTUALIZATION OF SEARCH FOR THE CAUSES OF FAILURE OF COMPONENTS OF A COMPLEX TECHNICAL SYSTEM

V.V. Vychuzhanin

National Odesa Polytechnic University
Ave.. Shevchenko, 1, Odesa, 65044, Ukraine;
e-mail: v.v.vychuzhanin@op.edu.ua

The task set in the article is to intellectualize the search for the causes of failures of subsystems (components), intersystem (intercomponent) connections of ship complex technical systems based on the assessment of the technical condition of systems by diagnostic features and predicting the risk of failures in their composition. The purpose of the article is to ensure the reliability of complex technical systems. The novelty of the results obtained lies in the fact that in the course of the study the principles of functioning of an intelligent system for searching for the causes of failures of a complex technical system with insensitivity to incomplete technological data about it were formulated. The principle of functioning of an intelligent system for searching for the causes of failures of a complex technical system by assessing and predicting the risk of failures of subsystems (components), intersystem (intercomponent) connections, its structure, in terms of technical and technological foundations of construction, is implemented on the example of a ship power plant. The result of the research is also the developed model for searching for the causes of failures of complex technical systems, which can be considered as a conceptual model with relative insensitivity to incomplete technological data about the system. Intellectualization of the search for the causes of failures of a complex technical system, taking into account hierarchical levels, makes it possible to determine vulnerable subsystems (components) on the basis of assessing the technical condition by diagnostic features and predicting the risk of failures.

**Keywords**: complex technical system, subsystem, component, intersystem and interelement communications, diagnostics, forecasting, model, failure risk assessment, intelligent system, search for failure causes

**Introduction.** Odern complex technical systems (CTS) are diverse in equipment, consist of many interconnected and interdependent subsystems, components [1,2]. The complication of the composition and the increase in the number of CTS installed on ships lead to an increase in the failure rate of such systems, to the need to repair CTS equipment, and hence to ship downtime. The use of intelligent systems for searching for the causes of failures of subsystems (FS), components (FC), intersystem (FIC) and intercomponent links (FI) CTS based on the assessment of their technical condition (TC) by diagnostic features and predicting the risk of failures in systems can significantly extend the life cycle ship CTS [3,4,5]. This article is devoted to the solution of this problem.

Studies of ship CTS survivability models show that the defeat of any FS, FC in systems gives rise to a significant number of possible scenarios and options for the development of emergency conditions of such systems, and hence to possible marine accidents and incidents [6,7], the statistics of which are reflected in the well-known bases [8,9,10,11,12]. According to statistics, one of the main CTS - ship power plant (SPP) accounts for 60-80% of all failures of ship systems].

The operational reliability of CTS is effectively achieved by the system operation strategy as a result of searching for the causes of failures based on equipment diagnostic data [13,14,15,16,17], predicting their TS [18,19,20,21,22].

The reliability of ship CTS can be reflected in the form of an assessment of the risk of failures [23,24,25,26,27,28]. For maritime shipping, the International Maritime Organization (IMO) has developed a consolidated text formalized safety assessment should comprise the following steps: identification of hazards; risk analysis; risk control options; 4 cost assessment benefit; recommendations for decision-making [29]. To analyze the risk of failure of system components, the world-wide Reliability Centered Maintenance method [30] is also widely used.

Currently, the volume of implementation of automation tools and artificial intelligence technologies continues to grow in various industries [31]. In accordance with the requirements of the Register of Maritime Navigation, all modern ships must be equipped with automation systems for technical means using digital technologies, as well as artificial intelligence technologies [1,32,33,34,35]. Such systems should constantly monitor FS, FC of ship CTS, analyze trends in changes in the TC, search for the causes of system equipment failures. To implement such tasks, appropriate algorithmic and software tools are needed.

In artificial intelligence, knowledge representation models are actively developing - Bayesian Belief Networks (BBN) [36,37,38]. They can be used to assess the risk of failures in CTS, providing a probabilistic basis for modeling the relationships between different failure modes and their root causes.

The algorithms used to search for the causes of failures FS, FC, FIC and FI based on the diagnosis of the vehicle, as a rule, are based on the control of tolerances of individual diagnostic parameters. However, the analysis and integral assessment of the technical condition of subsystems and complexes, the development of control actions in most cases is carried out by ship operators on the basis of heuristic rules.

Thus, the problems associated with ensuring the reliable operation of ship CTS require improvement and search for appropriate new methods, models and algorithms. They should be aimed not only at the prompt detection of equipment failure conditions, at solving problems of assessing and predicting the risk of system failures, but also at finding their causes under conditions of relative insensitivity to incomplete technological data on FS, FC. Since all modern ships must be equipped with automation systems for technical means using technologies based on artificial intelligence, the introduction of approaches based on such methods, models and algorithms should ensure the reliable operation of ship CTS. That is, taking into account the specifics and existing problems in ensuring reliability during the operation of ship CTS, the intellectualization of the search for the causes of failures based on the evaluation of TC systems by diagnostic features and predicting the risk of failures in their composition is an important direction in the development of modern technologies aimed at ensuring the safety and reliability of complex systems. systems and is an urgent task.

Statement of the problem: intellectualization of the search for the causes of failures of FS, FC, FIC and FI of ship CTS based on the assessment of the TC of systems by diagnostic features and predicting the risk of failures in their composition and eliminating the consequences of their occurrence.

Purpose of the work: ensuring the reliability and safety of the work of ship CTS.

**Main part.** The initial data for constructing an intellectualization model for searching for the causes of failures of components of a complex technical system based on TC assessment and predicting the risk of failures of complex systems using the example of an SPP based on BBN are: scheme and principle of operation of the SPP; failure probabilities FS, FC, FIC and FI of CTS links [39]. When modeling the BBN of the SPP for various values of the risk of failure of the input element BBN, the probabilities of loss of operability FS, FC, FIC and FI of connections for 20,000 hours of operation of the SPP were determined (Fig. 1).

From the retrospective analysis of the research results in the simulation of the SPP, the components that affect the overall performance of the system are identified. In the study of emergency situations, the analysis of incidents in the CTS, the main goal is to determine the cause of the accident. It follows from the research results that the maximum non-operating state during the operation of the SPP is 20,000 hours. corresponds to the CSPSC subsystem. Since the CSPSC subsystem is dependent at the level of the hierarchical structure of the SPP, therefore, it is necessary to check the subsystem in order to find the cause of its failure. Namely, to check the subsystems and all related subsystems at other levels of the BBN scheme.

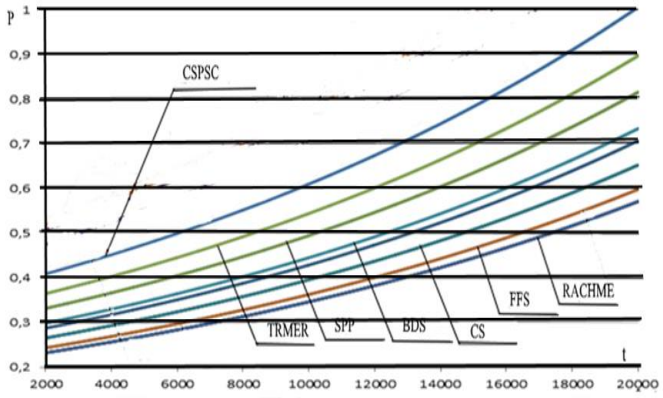



**Fig.1**. Probability of loss of operability of SPP subsystems

The scheme for searching for the cause of failure, for example, the CSPSC subsystem in the diagnostic model of the technical condition of the power plant using BBN is shown in Fig.2. For the BBN blocks of the SPP, we single out the blocks IE, CAS, SPP, CSPSC and intersystem communications IE-CAS, CAS - SPP, SPP - CSPSC for detailed consideration as an example to explain the principle of the model. Sets of risk of failures IE, CAS, SPP, CSPSC and interconnections IE-CAS, CAS - SPP, SPP - CSPSC at the initial moment of time and taking into account the dynamics of technical conditions in time based on a priori data on the failure rates BBN when the subsystems of the SPP IE, CAS, SPP, CSPSC and intersystem communications IE-CAS, CAS - SPP, SPP - CSPSC.

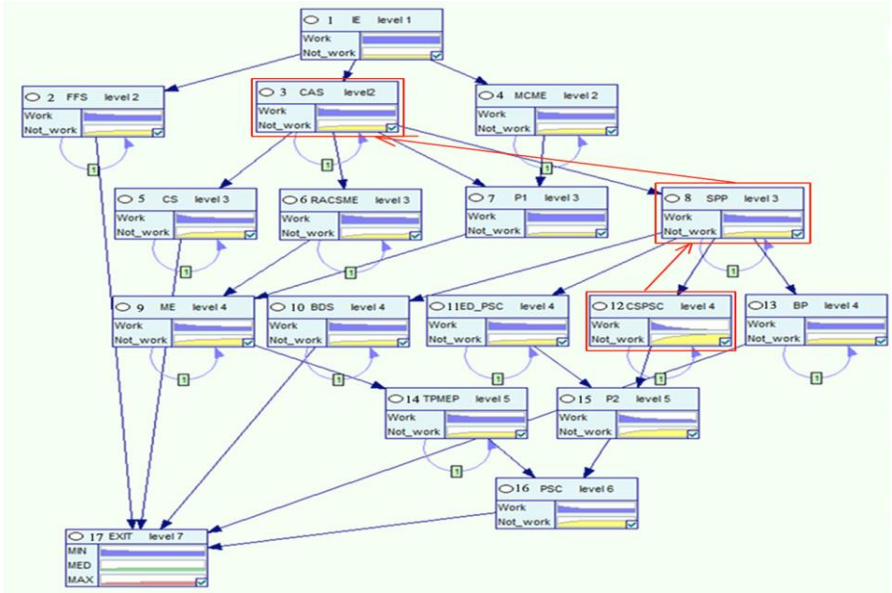



**Fig.2.** Scheme for searching for the cause of failure of the CSPSC subsystem in the diagnostic model of the technical condition of the SPP BBN

The search for the cause of failure of the CSPSC subsystem was performed in accordance with the algorithm shown in Fig. 3.

Symbols of subsystems, components of the SPP in BBN (Fig. 2): Input element - IE; Fire fighting system - FFS; Compressed air system - CAS; Manual control of the main engine - MCME; Control system - CS; Remote automated control system of the main engine - RACSME; Intermediate component - P1; Ship power plant - SPP; Main engine - ME; Ballast drainage system - BDS; Emergency drive propulsion and steering complex - ED PSC; Control system for propulsion and steering complex - CSPSC; Boiler plant - BP; Transfer of power from the main engine to the propeller - TPMEP; Intermediate component = P2; Propulsion and steering complex - PSC; Output component - EXIT.

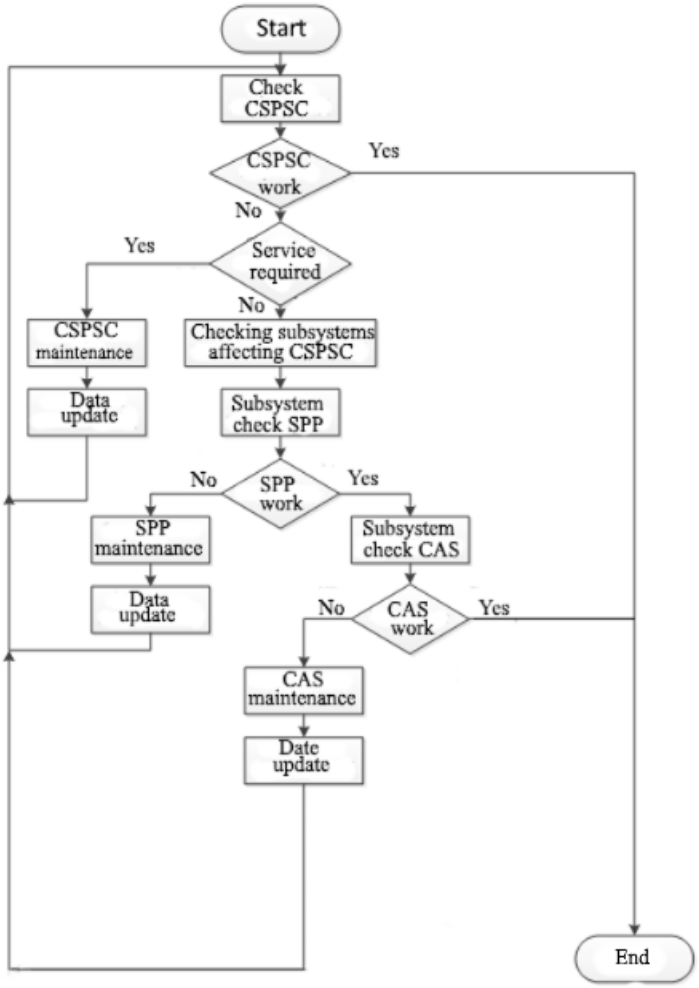



**Fig. 3.** Algorithm for troubleshooting the CSPSC subsystem

The technique for building a model based on BBN can be represented as follows:

1. Building BBN:

1.1. Vertices and intersystem (intercomponent) BBNs are created, denoting FS, FC, FIC and FI STS, taking into account their TC:

1.1.1. Each FS, FC may be in the following technical condition:

$Work_{n_{S(C)}}^{<m_{S(C)}>}$ - operational state $n_{S(C)}$ - th FS (FC) $m_{S(C)}$ - th level;

$Not\_work_{a(z)_{I_{S(C)}}}^{<b,q>}$ - partial (complete) failure $n_{S(C)}$ - th FS (FC) $m_{S(C)}$ - th level.

1.1.2. Each FIC and FI connection is in the following states:

$Work_{a(z)_{I_{S(C)}}}^{<b,q>}$ - operational state $a(z)_{I_{S(C)}}$ - th FIC (FI) connection $b(q)$ level;

$Not\_work^{<b,q>}_{a(z)_{I_{S(C)}}}$ - partial failure (complete) $a(z)_{I_{S(C)}}$ - th FIC (FI) communication *b(q)* level

where $S(C)$ - is the set FS (FC) CTC;

$I_S(I_C)$ - a set of FIC (FI) CTS connections;

$n_{s(c)}$ - number FS (FC) CTC;

$m_{s(c)}$ - number of the hierarchical level FS (FC) CTC;

*a(z)* – FIC (FI) number of CTS communication connections;

*b(q)* – number of the hierarchical level FIC (FI) of CTS communication links

1.2. Links between BBN vertices are indicated, denoting FS, FC, FIC and FI CTC links

2. BBN parameters are specified:

2.1. The risk of failures at the initial moment of time for FS, FC, FIC and FI of the CTS connection, assuming that before the start of the CTS operation they are all operable:

$$R(Work^{<m_{S(C)}>}_{n_{S(C)}})_{t=0} = F(P(Work^{<m_{S(C)}>}_{n_{S(C)}})_{t=0}) = 0 \quad (1)$$

$$R(Work^{<b,q>}_{a(z)_{I_{S(C)}}})_{t=0} = F(P(Work^{<b,q>}_{a(z)_{I_{S(C)}}})_{t=0}) = 0$$

The risk of failures at the initial moment of time for FS, FC, FIC and FI of the CTS connection, assuming that before the start of the CTS operation they are all inoperable:

$$R(Not\_work^{<m_{S(C)}>}_{n_{S(C)}})_{t=0} = F(P(Not\_work^{<m_{S(C)}>}_{n_{S(C)}})_{t=0}) = 1 \quad (2)$$

$$R(Not\_work^{<b,q>}_{a(z)_{I_{S(C)}}})_{t=0} = F(P(Not\_work^{<b,q>}_{a(z)_{I_{S(C)}}})_{t=0}) = 1$$

2.3. The risk of failure of FS, FC, FIC and FI of the CTS connection at the current time, provided that some subsystems (components), intersystem (intercomponent) connections failed at the previous time:

$$R((Not\_work^{<m_{S(C)}>}_{n_{S(C)}})_t / (Not\_work^{<m_{S(C)}>}_{n_{S(C)}})_{t-1}) = 1 \quad (3)$$

$$R((Not\_work^{<,b,q>}_{a(z)_{I_{S(C)}}})_t / (Not\_work^{<b,q>}_{a(z)_{I_{S(C)}}})_{t-1})) = 1$$

For the BBN blocks of the SPP (Fig. 2) IE, CAS, SPP, CSPSC and interconnections IE-CAS, CAS - SPP, SPP - CSPSC, sets of failure risk at the initial time and taking into account the dynamics of technical conditions in time based on a priori data on failure rates:

$$R(Work^{1,2,3,4}_{1,3,8,12})_{t=0} = 0;$$

$$R(Not\_work^{1,2,3,4}_{1,3,8,12})_{t=0} = 1;$$

$$R(Work^{2,3,4}_{IE-CAS,CAS-SPP,SPP=CSPSC})_{t=0} = 0 \quad (4)$$

$$R(Not\_work^{2,3,4}_{IE-CAS,CAS-SPP,SPP=CSPSC})_{t=0} = 1;$$

$$R((Work^{1,2,3,4}_{1,3,8,12})_t / (Work^{1,2,3,4}_{1,3,8,12})_{t-1}) = 0{,}1;$$

$$R((Work^{2,3,4}_{IE\_CAS,CAS\_SPP,SPP-CSPSC})_t / (Work^{2,3,4}_{IE\_CAS,CAS-SPP,SPP-CSPSC})_{t-1}) = 0{,}1$$

Sets of risk of failures at the current moment of time, taking into account the previous state of subsystems and intersystem communications, can be within:

- the level of risk of failure is assessed as minimal, the consequences of an accident are minimal when:

$$R((Not\_work^{1,2,3,4}_{1,3,8,12})_t / (Work^{1,2,3,4}_{1,3,8,12})_{t-1}) = 0{,}1 - 0{,}2 \quad (5)$$

$$R((Not\_work^{2,3,4}_{IE\_CAS,CAS\_SPP,SPP-CSPSC})_t/(Work^{1,3,4}_{IE\_CAS,CAS-SPP,SPP-CSPSC})_{t-1}) = 0{,}1-0{,}2$$

- the level of risk of failure is assessed as acceptable, the consequences of the accident are insignificant when:

$$R((Not\_work^{1,2,3,4}_{1,3,8,12})_t/(Work^{1,2,3,4}_{1,3,8,12})_{t-1}) = 0{,}2-0{,}37 \quad (6)$$

$$R((Not\_work^{2,3,4}_{IE\_CAS,CAS\_SPP,SPP-CSPSC})_t/(Work^{1,3,4}_{IE\_CAS,CAS-SPP,SPP-CSPSC})_{t-1}) = 0{,}2-0{,}37$$

- the level of risk of failure is estimated as maximum, the consequences of the accident are significant when:

$$R((Not\_work^{1,2,3,4}_{1,3,8,12})_t/(Work^{1,2,3,4}_{1,3,8,12})_{t-1}) = 0{,}37-0{,}63 \quad (7)$$

$$R((Not\_work^{2,3,4}_{IE\_CAS,CAS\_SPP,SPP-CSPSC})_t/(Work^{1,3,4}_{IE\_CAS,CAS-SPP,SPP-CSPSC})_{t-1}) = 0{,}37-0{,}63$$

- the failure risk level is assessed as critical when:

$$R((Not\_work^{1,2,3,4}_{1,3,8,12})_t/(Work^{1,2,3,4}_{1,3,8,12})_{t-1}) = 0{,}63-1 \quad (8)$$

$$R((Not\_work^{2,3,4}_{IE\_CAS,CAS\_SPP,SPP-CSPSC})_t/(Work^{1,3,4}_{IE\_CAS,CAS-SPP,SPP-CSPSC})_{t-1}) = 0{,}63-1$$

The risk distribution of failures of subsystems (components), intersystem (intercomponent) links in BBN, taking into account failures and restorations, has the form:

- for failure risk distributions Control system for propulsion and steering complex –CSPSC SPP in BBN:

$$\begin{aligned}
&R(Work^4_{12})_t = R((Work^4_{12})_t/(Work^4_{12})_{t-1})\cdot R(Work^1_1)_{t-1}\cdot R(Work^2_3)_{t-1}\cdot R(Work^3_8)_{t-1}\times\\
&\times R(Work^4_{12})_{t-1}\cdot R(Work^2_{IE_CAS})_{t-1}\cdot R(Work^3_{CAS-SPP})_{t-1}\cdot R(Work^4_{SPP-CSPSC})_{t-1}+\\
&+R(Work^4_{12})_t = R((Work^4_{12})_t/(Not_work^4_{12})_{t-1})\cdot R(Work^1_1)_{t-1}\cdot R(Work^2_3)_{t-1}\cdot R(Work^3_8)_{t-1}\times\\
&\times R(Not_work^4_{12})_{t-1}\cdot R(Work^2_{IE_CAS})_{t-1}\cdot R(Work^3_{CAS-SPP})_{t-1}\cdot R(Work^4_{SPP-CSPSC})_{t-1}+\\
&+R(Work^4_{12})_t = R((Work^4_{12})_t/(Work^4_{12})_{t-1})\cdot R(Not_work^1_1)_{t-1}\cdot R(Work^2_3)_{t-1}\cdot R(Work^3_8)_{t-1}\times\\
&\times R(Work^4_{12})_{t-1}\cdot R(Work^2_{IE_CAS})_{t-1}\cdot R(Work^3_{CAS-SPP})_{t-1}\cdot R(Work^4_{SPP-CSPSC})_{t-1}+\\
&+R(Work^4_{12})_t = R((Work^4_{12})_t/(Work^4_{12})_{t-1})\cdot R(Work^1_1)_{t-1}\cdot R(Not_work^2_3)_{t-1}\cdot R(Work^3_8)_{t-1}\times\\
&\times R(Work^4_{12})_{t-1}\cdot R(Work^2_{IE_CAS})_{t-1}\cdot R(Work^3_{CAS-SPP})_{t-1}\cdot R(Work^4_{SPP-CSPSC})_{t-1}+\\
&+R(Work^4_{12})_t = R((Work^4_{12})_t/(Work^4_{12})_{t-1})\cdot R(Work^1_1)_{t-1}\cdot R(Work^2_3)_{t-1}\cdot R(Not_work^3_8)_{t-1}\times\\
&\times R(Work^4_{12})_{t-1}\cdot R(Work^2_{IE_CAS})_{t-1}\cdot R(Work^3_{CAS-SPP})_{t-1}\cdot R(Work^4_{SPP-CSPSC})_{t-1}+\\
&+R(Work^4_{12})_t = R((Work^4_{12})_t/(Work^4_{12})_{t-1})\cdot R(Work^1_1)_{t-1}\cdot R(Work^2_3)_{t-1}\cdot R(Work^3_8)_{t-1}\times\\
&\times R(Work^4_{12})_{t-1}\cdot R(Not_work^2_{IE_CAS})_{t-1}\cdot R(Work^3_{CAS-SPP})_{t-1}\cdot R(Work^4_{SPP-CSPSC})_{t-1}+\\
&+R(Work^4_{12})_t = R((Work^4_{12})_t/(Work^4_{12})_{t-1})\cdot R(Work^1_1)_{t-1}\cdot R(Work^2_3)_{t-1}\cdot R(Work^3_8)_{t-1}\times\\
&\times R(Work^4_{12})_{t-1}\cdot R(Work^2_{IE_CAS})_{t-1}\cdot R(Not_work^3_{CAS-SPP})_{t-1}\cdot R(Work^4_{SPP-CSPSC})_{t-1}+\\
&+R(Work^4_{12})_t = R((Work^4_{12})_t/(Work^4_{12})_{t-1})\cdot R(Work^1_1)_{t-1}\cdot R(Work^2_3)_{t-1}\cdot R(Work^3_8)_{t-1}\times\\
&\times R(Work^4_{12})_{t-1}\cdot R(Work^2_{IE_CAS})_{t-1}\cdot R(Work^3_{CAS-SPP})_{t-1}\cdot R(Not_work^4_{SPP-CSPSC})_{t-1}
\end{aligned} \quad (9)$$

- for ship power plant failure risk distributions in BBN:

$$R(Work_8^3)_t = R((Work_8^3)_t/(Work_8^3)_{t-1})\cdot R(Work_1^1)_{t-1}\cdot R(Work_3^2)_{t-1}\times$$
$$\times R(Work_8^3)_{t-1}\cdot R(Work_{IE\_CAS}^2)_{t-1}\cdot R(Work_{CAS-SPP}^3)_{t-1}+$$
$$+R(Work_8^3)_t = R((Work_8^3)_t/(Not\_work_8^3)_{t-1})\cdot R(Work_1^1)_{t-1}\cdot R(Work_3^2)_{t-1}\times$$
$$\times R(Not\_work_8^3)_{t-1}\cdot R(Work_{IE\_CAS}^2)_{t-1}\cdot R(Work_{CAS-SPP}^3)_{t-1}+$$
$$+R(Work_8^3)_t = R((Work_8^3)_t/(Work_8^3)_{t-1})\cdot R(Not-work_1^1)_{t-1}\cdot R(Work_3^2)_{t-1}\times \quad (10)$$
$$\times R(Work_8^3)_{t-1}\cdot R(Work_{IE\_CAS}^2)_{t-1}\cdot R(Work_{CAS-SPP}^3)_{t-1}+$$
$$+R(Work_8^3)_t = R((Work_8^3)_t/(Work_8^3)_{t-1})\cdot R(Work_1^1)_{t-1}\cdot R(Not\_work_3^2)_{t-1}\times$$
$$\times R(Work_8^3)_{t-1}\cdot R(Work_{IE\_CAS}^2)_{t-1}\cdot R(Work_{CAS-SPP}^3)_{t-1}+$$
$$+R(Work_8^3)_t = R((Work_8^3)_t/(Work_8^3)_{t-1})\cdot R(Work_1^1)_{t-1}\cdot R(Work_3^2)_{t-1}\times$$
$$\times R(Work_8^3)_{t-1}\cdot R(Not\_work_{IE\_CAS}^2)_{t-1}\cdot R(Work_{CAS-SPP}^3)_{t-1}+$$
$$+R(Work_8^3)_t = R((Work_8^3)_t/(Work_8^3)_{t-1})\cdot R(Work_1^1)_{t-1}\cdot R(Work_3^2)_{t-1}\times$$
$$\times R(Work_8^3)_{t-1}\cdot R(Work_{IE\_CAS}^2)_{t-1}\cdot R(Not\_work_{CAS-SPP}^3)_{t-1}$$

- for distributions of the risk of failure of the compressed air system of the power plant in BBN:

$$R(Work_3^2)_t = R((Work_3^2)_t/(Work_3^2)_{t-1})\cdot R(Work_1^1)_{t-1}\cdot R(Work_3^2)_{t-1}\cdot R(Work_{IE\_CAS}^2)_{t-1}+$$
$$+R((Work_3^2)_t/(Not\_work_3^2)_{t-1})\cdot R(Work_1^1)_{t-1}\cdot R(Not\_work_3^2)_{t-1}\cdot R(Work_{IE\_CAS}^2)_{t-1}+$$
$$+R((Work_3^2)_t/(Work_3^2)_{t-1})\cdot R(Not\_work_1^1)_{t-1}\cdot R(Work_3^2)_{t-1}\cdot R(Work_{IE\_CAS}^2)_{t-1}+$$
$$+R((Work_3^2)_t/(Work_3^2)_{t-1})\cdot R(Work_1^1)_{t-1}\cdot R(Work_3^2)_{t-1}\cdot R(Not\_work_{IE\_CAS}^2)_{t-1} \quad (11)$$
$$R(Work_1^1)_t = R((Work_1^1)_t/(Work_1^1)_{t-1})\cdot R(Work_1^1)_{t-1}+ \quad (12)$$

- for the distributions of the risk of failure of the input component of the EMS in BBN:

$$+R((Work_1^1)/(Not\_work_1^1)_{t-1})\cdot R(Not\_work_1^1)_{t-1}$$

If after 20000 hours. operation, the CSPSC subsystem is in a working state, then a study is carried out on the operability of the CAS, SPP subsystems that affect the operability of the CSPSC, the failure of which can lead to the failure of the entire SPP.

After maintenance of the CSPSC subsystem, the assessments of the risk of failures of the SPP subsystems are recalculated. Because The SPP directly affects the CSPSC, so this subsystem needs to be tested. The failure of the SPP will be the probabilistic cause of the failure of the CSPSC subsystem. After the maintenance of the SPP, the data on the technical condition of the SPP subsystem is updated, and the assessments of the risk of failures of the SPP subsystems will be recalculated. If after maintenance of the CSPSC and SPP subsystems, as well as recalculation of the failure risk assessment for these subsystems, then it is necessary to check the CAS subsystem. The failure of the CAS will be the probabilistic cause of the failure of the CSPSC subsystem. After the CAS maintenance, the data on the technical condition of the CAS subsystem are updated, and the assessments of the risk of failures of the SPP subsystems will be recalculated.

Thus, based on the intellectualization of the estimation of the TS FS, FC, FIC and FI of the CTS links by diagnostic features, it is possible to search for the causes of failures of the ship's CTS components.

**Conclusions.** Based on the evaluation of the TC systems by diagnostic features and predicting the risk of failures in their composition, the search for the causes of failures

of FS, FC, FIC and FI of ship CTS communications was intellectualized. In the course of the study, the principles of functioning of an intelligent system for searching for the causes of CTS failures with insensitivity to incomplete technological data about it were formulated. The principle of functioning of the intelligent system for searching for the causes of CTS failures by assessing and predicting the risk of failures of FS, FC, FIC and FI links, its structure, in terms of technical and technological foundations of construction, is implemented using the example of a ship power plant. A model for searching for the causes of CTS failures has been developed, which can be considered as a conceptual model with relative insensitivity to incomplete technological data about the system. Intellectualization of the search for the causes of CTS failures, taking into account hierarchical levels, makes it possible to determine vulnerable subsystems (components) on the basis of assessing the technical condition by diagnostic features and predicting the risk of failures.

**References**


1. Vychuzhanin V.V., Rudnichenko N.D. Metody informatsionnykh tekhnologiy v diagnostike sostoyaniya slozhnykh tekhnicheskikh sistem. Monografiya. Odessa: Ekologiya, 2019. 178 p.
2. Vychuzhanin V., Rudnichenko N. Devising a method for the estimation and prediction of technical condition of ship complex systems. *Eastern-European Journal of Enterprise Technologies.* 2016. V.84. No.6/9. P.4-11.
3. Vychuzhanin V.V., Rudnichenko N.D., Vychuzhanin A.V., Yurchenko M.A. Programnoye prilozheniye dlya avtomatizatsii postroyeniya modeli otsenok riska otkazov slozhnykh tekhnicheskikh system. *Ínformatika ta matematichní metodi v modelyuvanní*. 2018. V.8. No.3. P. 200-208.
4. ISO 13381-1:2015 Condition monitoring and diagnostics of machines – Prognostics – Part 1: General guidelines. 2015. 21 p.
5. Andersen B.A. Diagnostic System for Remote Real-Time Monitoring of Marine Diesel-Electric Propulsion Systems. Leipzig. 2011. 45 p.
6. Vychuzhanin V.V. Model otsenki zhivuchesti sudovykh tekhnicheskikh sistem. *Vestnik Mikolaïvs'kogo korablebudívnogo uníversitetu.* 2012. No.3. P.62-67
7. Vychuzhanin V.V., Rudnichenko N.D. Otsenki funktsional'nykh riskov sudovykh energeticheskikh ustanovok. *Vestnik VGAVT*. 2014. No. 40. P.244-249.
8. Marine Accident Investigation Branch Reports. URL: https://gisis.imo.org/Public/Default.aspx
9. Marine Accident Investigation Branch Reports. URL: https://www.gov.uk/maib-reports
10. Mars Reports. URL: https://www.nautinst.org/resource-library/mars/mars-reports.html
11. Investigstion Reports. URL: https://ntsb.gov/investigations/AccidentReports/Pages/marine.aspx
12. Casuality and Events. URL: https://ihsmarkit.com/products/casualty-and-events.htm l
13. O'Neill J. Technical Risk Assessment: a Practitioner's Guide. Australia, 2007. 29 p.
14. Krarowski R. Diagnosis modern systems of marine diesel engine. *Journal of KONES Powertrain and Transport.* 2014. P. 191-198. DOI: 10.5604/12314005.1133203

15. Boullosa-Falces D., Barrena J.L.L., Lopez-Arraiza A., Menendez J., Solaetxe M.A.G. Monitoring of fuel oil process of marine diesel engine. Appl. *Therm. Eng*. 2017. No.127. P.517–526. DOI: https://doi.org/10.1016/j.applthermaleng.2017.08.036
16. Raptodimos Y., Lazakis I. Using artificial neural network-self-organising map for data clustering of marine engine condition monitoring applications. *Ships Offshore Struc.* 2018. No.13. P.649–656. DOI:10.1080/17445302.2018.1443694
17. Srensen A.J. Marine Control Systems Propulsion and Motion Control of Ships and Ocean Structures. 2013. 526 p.
18. Vyuzhuzhanin V.V., Rudnichenko N.D., Shibaeva N.O. Data Control in the Diagnostics and Forecasting the State of Complex Technical Systems. *Herald of Advanced Information Technology.* 2019. V.2. No.2. P.183-196. DOI 10.15276/hait.03.2019.2
19. Vyuzhuzhanin V., Gritsuk I. The Complex Application of Monitoring and Express Diagnosing for Searching Failures on Common Rail System Units DP. *SAE International*. 2018. DOI:10.4271/2018-01-1773.
20. Vychuzhanin V., Rudnichenko N. Complex Technical System Condition Diagnostics and Prediction Computerization. *CMIS-2020 – Computer Modeling and Intelligent Systems*. Ceur-ws.org. URL: https://ceur-ws.org/Vol-2608/
21. Sorensen A.J. Marine Control Systems Propulsion and Motion Control of Ships and Ocean Structures. Trondheim, Norway: Department of Marine Technology-NTNU, 2013. 537 p.
22. Lazakis I. Advanced ship systems condition monitoring for enhanced inspection, maintenance and decision making in ship operations. *Transportation Research Procedia*. 2016. No.14. P.1679 – 1688.
23. IEC 31010:2019 Risk management — Risk assessment techniques. 2019. 127p.
24. Marhavilas P.K., Koulouriotis D., Gemeni V. Risk analysis and assessment methodologies in the work sites: On a review, classification and comparative study of the scientific literature of the period 2000-2009. *Journal of Loss Prevention in the Process Industries.* 2011. No.24. P.477-523. DOI: https://doi.org/10.1016/j.jlp.2011.03.004
25. Vychuzhanin V., Rudnichenko N. Assessment of risks structurally and functionally complex technical systems. *Eastern-European Journal of Enterprise Technologies*. 2014. V.1. No. 2. P.18-22. DOI: 10.15587/1729-4061.2014.19846.
26. Asuquo M., Wang,J., Zhang L., Phylip-Jones G. An integrated risk assessment for maintenance prediction of oil wetted gearbox and bearing in marine and offshore industries using a fuzzy rule base method. *Proc. Inst. Mech. Eng. Part M J. Eng. Marit. Environ.* 2020. No.234. P.147509021989952. DOI:10.1177/1475090219899528
27. Zhang M., Montewka J., Manderbacka T., Kujala P., Hirdaris S. A Big Data Analytics Method for the Evaluation of Ship - Ship Collision Risk reflecting Hydrometeorological Conditions. *Reliability Eng. & System Safety.* 2021. V.213. P.107674. DOI: https://doi.org/10.1016/j.ress.2021.107674
28. Vychuzhanin V.V. Tekhnicheskiye riski slozhnykh kompleksov funktsional'no vzaimosvyazannykh strukturnykh komponentov sudovykh energeticheskikh

ustanovok. *Vísnik Odes'kogo natsíonal'nogo mors'kogo uníversitetu, zbírnik naukovikh prats*. 2014. V.2. No.40. P. 68-77.

29. Guidelines for formal safety assessment (FSA) for use in the IMO rule-making process. London: International maritime organization, 2002. 54 p.
30. SAE JA1012. A Guide to the Reliability-Centered Maintenance (RCM) Standard, Society of Automotive Engineers. 2002. 57 p.
31. Vychuzhanin V., Shibaeva N., Vychuzhanin A., Rudnichenko N. Intellectualization Method and Model of Complex Technical System's Failures Risk Estimation and Prediction. *Ceur-ws.org*. 2023. URL: https://ceur-ws.org/Vol-3392/paper11.pdf
32. Pearl J. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. New York: Morgan Kaufman Publ., 1991. DOI:10.2307/2026705
33. Coraddu A.L., Ghio O.A., Savio S., Anguita D., Figari M. Machine Learning Approaches for Improving Condition-based Maintenance of Naval Propulsion Plants. *Proceedings of the Institution of Mechanical Engineers, Part M: J. of Engineering for the Maritime Environment.* 2016. V. 230. No.1. P. 136–153.
34. Liu Z., Liu J., Zhou F., Liu R.W., Xiong N. A Robust GA/PSO-Hybrid Algorithm in Intelligent Shipping Route Planning Systems for Maritime Traffic Networks. *Journal of Internet Technology*. 2018. V.19. No.6. P.1635–1644.
35. Jokioinen E. Remote and autonomous ship – the next steps. *AAWA Project Position*. 2017. URL: http://docplayer.net/19502019-Remote-and-autonomous-ships-the-next-steps.html
36. Wang C.R., Guan C.A Bayesian inference-based approach for performance prognostics towards uncertainty quantification and its applications on the marine diesel engine. *ISA Trans.* 2021. No.118, P.159–173. DOI: 10.1016/j.isatra.2021.02.024 .
37. Jensen F.V. Bayesian Networks and Decision Graphs. Berlin: Springer, 2007. 457 p.
38. Handayani D., Sediono W. Anomaly Detection in Vessel Tracking: A Bayesian Networks (Bns) Approach. *International Journal of Maritime Engineering (RINA Transactions Part A).* 2015. V.157 No.A3. P.145–152. DOI: 10.3940/rina.ijme.2015.a3.316.
39. Vychuzhanin A.V. Intelligent system for assessing and forecasting the risk of failure of components of a complex technical system. *Informatics and Mathematical Methods in Simulation. 2022.* V.12. No. 3. P. 154–161.

V.V. Vychuzhanin

# ІНТЕЛЕКТУАЛІЗАЦІЯ ПОШУКУ ПРИЧИН ВІДМОВ КОМПОНЕНТІВ СКЛАДНОЇ ТЕХНІЧНОЇ СИСТЕМИ

В.В. Вичужанін

Національний університет «Одеська політехніка»
просп. Шевченка, 1, Одеса, 65044, Україна
e-mail: v.v.vychuzhanin@op.edu.ua

Поставлене у статті завдання полягає в інтелектуалізації пошуку причин відмов підсистем (компонентів), міжсистемних (міжкомпонентних) зв'язків суднових складних технічних систем на основі оцінювання технічного стану систем за діагностичними ознаками та прогнозування ризику відмов у їхньому складі. Метою статті є забезпечення надійності роботи складних технічних систем. Новизна одержаних результатів полягає в тому, що в ході дослідження сформульовано принципи функціонування інтелектуальної системи пошуку причин відмов складної технічної системи з нечутливістю до неповних технологічних даних про неї. Принцип функціонування інтелектуальної системи пошуку причин відмов складної технічної системи за оцінками та прогнозування ризику відмов підсистем (компонентів), міжсистемних (міжкомпонентних) зв'язків, її структура, у термінах технічних та технологічних основ побудови реалізовано на прикладі суднової енергетичної установки. Результатом досліджень також є розроблена модель пошуку причин відмов складних технічних систем, яка може розглядатися як концептуальна модель, що володіє відносною нечутливістю до неповних технологічних даних про систему. Інтелектуалізація пошуку причин відмов складної технічної системи з урахуванням ієрархічних рівнів дозволяє на основі оцінювання технічного стану за діагностичними ознаками та прогнозування ризику відмов визначати вразливі підсистеми (компоненти).

**Ключові слова:** складна технічна система, підсистема, компонент, міжсистемні та міжелементні зв'язки, діагностика, прогнозування, модель, оцінка ризику відмови, інтелектуальна система, пошук причин відмов

DOI 10.15276/imms.v13.no1-2.27
УДК 52.50

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp.27-33

# СИНТЕЗ ТА МОДЕЛЮВАННЯ ОПТИМАЛЬНОЇ ЗА ШВИДКОДІЄЮ СЛІДКУЮЧОЇ СИСТЕМИ

С.О. Бобріков, Л.Л. Прокофьєва, А.А. Савельєв

Національний університет «Одеська політехніка»,
Проспект Шевченка, 1, Одеса, 65044, Україна;E-mail:
bobrikov1932@gmail.com, luleopro@gmail.com, sanp277@gmail.com

Запропоновано метод побудови і розрахунку структурної схеми системи управління об'єктом, який конструктивно утворено послідовним з'єднанням аперіодичної ланки першого порядку та інтегратором, а формалізовано описано математичною моделлю у вигляді добутку відповідних передаточних функцій. Поширеною типової реалізацією зазначеного об'єкта управління може слугувати електричний двигун з джерелом вхідної напруги. Запропонований метод дозволяє побудувати систему управління, близьку до оптимальної за швидкодією. Причому оптимальність, в даному випадку, визначається тим, що об'єкт в процесі функціонування вмикається в релейному режимі. Перехідний процес для визначеного об'єкта управління, в умовах оптимального режиму, являє собою два етапи: розгону та гальмування, які здійснюються за максимальних значень вхідного впливу — тобто вхідної напруги. Для визначення моментів перемикання вхідного впливу (сигналу) використано метод фазової площини, а фазовими координатами при цьому виступають: по осі абсцис — вихідна величина (для наведеного прикладу — частота обертання валу двигуна), а по осі ординат — швидкість зміни вихідної величини (або похідна від частоти обертання валу двигуна, тобто відповідне прискорення). Розроблено структурну схему системи управління, що забезпечує розгін та гальмування об'єкта при максимальних значеннях вхідного сигналу ($U^{+}_{\max}$ та $U^{-}_{\max}$). Оптимальність перехідного процесу забезпечується тим, що похибка управління порівнюється з сигналом, що надходить від ланки, налаштованої лінії перемикання. Наведено формули обчислення лінії перемикання на фазовій площині, а також програму розрахунку лінії перемикання, написану в середовищі MATLAB у форматі m-file. Проведено моделювання системи у пакеті MATLAB-Simulink, що підтверджує оптимальність процесу управління.

**Ключові слова**: структурна схема, система управління, передаточна функція,

**Вступ.** Розглянемо оптимальні системи, в яких критерієм оптимальності є мінімальний час відпрацювання системою заданого сигналу. В якості об'єкта управління приймаємо виконавчий пристрій (наприклад, двигун постійного струму) з підсилювачем потужності. Вважаємо, що рівняння об'єкта управління (тобто його математична модель) має вигляд:

$$T_0 \frac{d^2 y}{dt^2} + \frac{dy}{dt} = W_0 u(t), \tag{1}$$

Функціональну схему замкнутої оптимальної за швидкодією системи наведено на рис.1. Приймаємо умову, що об'єкт управління ОУ описується диференціальним рівнянням (1) і що корені характеристичного рівняння речові та недодатні.

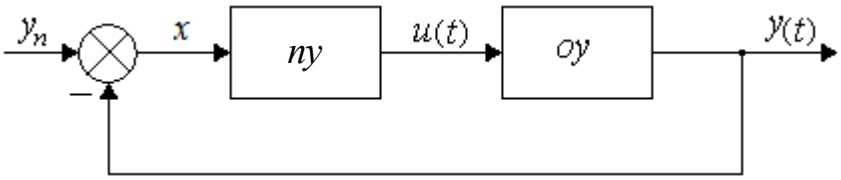



**Рис. 1.** Функціональна схема замкнутої системи.
*ny* — управляючий пристрій; *oy* — об'єкт управління

Відповідно до теореми про $n$-інтервалів [1] для забезпечення максимальної швидкодії в такій системі сигнал управління $u(t)$ повинен мати два гранично допустимі значення: $U^{+}_{\max}$ та $U^{-}_{\max}$, а перехідний процес при цьому складається з двох частин: розгону та гальмування.

Задачею оптимального за швидкодією управління в постановці, яка розглядається, є переведення вихідного сигналу (або — в термінах теорії управління — «вихідної координати») об'єкта (для наведеного прикладу — частоти обертання валу двигуна) з початкового стану (наявної частоти обертання на початок моменту управління) в кінцевий стан (до бажаного значення частоти обертання за наслідками управління) за мінімальний час. Алгоритм оптимального управління зводиться до визначення моментів перемикання сигналу управління $u(t)$ зі значення $U^{+}_{\max}$ (тобто з режиму розгону) у значення $U^{-}_{\max}$ (тобто у режим гальмування). Моменти перемикання можна визначити, розв'язуючи відповідні диференціальні рівняння. Слід зазначити, що спільне розв'язування системи диференціальних рівнянь (визначення періоду часу, що відповідає режиму розгону та періоду часу, що відповідає режиму гальмування) призводить до необхідності розв'язувати трансцендентні рівняння [1 — 4].

Нижче показано, що побудувати систему, близьку до оптимальної можна, *не визначаючи моменти перемикання*.

**Мета роботи.** *Метою* роботи є розробка управляючого пристрою оптимальної за швидкодією системи, його структурної схеми та методу розрахунку, причому такого, що останній *не вимагає визначення моментів перемикання керуючого сигналу* і виконання, у зв'язку з цим, громіздких обчислень, пов'язаних з розв'язком трансцендентних рівнянь.

**Основна частина.** Сигнал управління $u(t)$ (рис.1) повинен змінюватися за релейним законом, отже, така система є суттєво нелінійною. Для опису та розрахунку такої системи використаємо метод *фазової площини*. При цьому розглядатимемо фазову площину, по координатних осях якої відкладено: похибку управління $x(t) = y_n - y(t)$ — вісь абсцис, і швидкість зміни похибки $\varepsilon(t) = dx(t)/dt$ — вісь ординат.

Рівняння (1), записане щодо похибки, набуває вигляду:

$$T_0 \frac{d^2 x(t)}{dt^2} + \frac{d\,x(t)}{dt} = -k_0 u(t). \qquad (2)$$

Щоб перейти від диференціального рівняння системи (2) до рівняння в координатах фазової площини, потрібно з рівняння (2) виключити незалежний параметр — час $t$. Для цього представимо другу похідну змінної $x(t)$ наступним чином:

$$\frac{d^2x(t)}{dt}=\frac{d\varepsilon(t)}{dt}=\frac{d\varepsilon(t)}{dt}\cdot\frac{dx(t)}{dx(t)}=\frac{\varepsilon(t)d\varepsilon(t)}{dx(t)}$$ або (спрощено) $$\frac{d^2x(t)}{dt}=\frac{\varepsilon\,d\varepsilon}{dx}.$$

Підставляючи в рівняння (2) першу та другу похідні змінної $x(t)$ отримаємо, у координатах фазової площини $(x(t),\varepsilon(t))$ або $(x,\varepsilon)$, наступне:

$$T_0\frac{\varepsilon\,d\varepsilon}{dx}+\varepsilon=-k_0u(t). \tag{3}$$

Рівняння (3) являє собою диференційне рівняння «зі змінними, що розділяються». Після очевидних перетворень, отримаємо:

$$dx=-T_0\frac{\varepsilon\,d\varepsilon}{\varepsilon+k_0u}.$$

В останньому рівнянні справа — табличний інтеграл. В результаті інтегрування, маємо:

$$x=-T_0\varepsilon+T_0k_0u\times\ln|k_0u+\varepsilon|+C_1, \tag{4}$$

де $C_1$ — постійна інтегрування.

По суті, вираз (4) описує сімейство рівнянь — фазових траєкторій, кожне з яких визначається постійною інтегрування $C_1$ і залежить від початкових умов (які, в свою чергу, власно і визначають конкретне значення постійної інтегрування $C_1$). Визначимо рівняння фазової траєкторії, яка проходить через центр координат. Для цього приймемо кінцеві умови перехідного процесу: $x_{\text{кін}}=\dot{x}_{\text{кін}}=\varepsilon_{\text{кін}}=0$.

Підставивши кінцеві умови в рівняння (4) визначимо значення постійної інтегрування $C_1$:

$$C_1=-T_0k_0u(t)\times\ln|k_0u(t)|. \tag{5}$$

Далі, підставляючи значення $C_1$ у рівняння (4),отримаємо:

$$x=-T_0\varepsilon+T_0k_0u(t)\times\ln|k_0u(t)+\varepsilon|-T_0k_0u(t)\times\ln|k_0u(t)|. \tag{6}$$

Рівняння (6) — це рівняння фазових траєкторій, які проходять через центр координат, тобто: $x=\varepsilon=0$.

Розглянемо два випадки.

1. Нехай при $t=0$, $y(t)=0$ на вхід подано сигнал $y_n$. При цьому похибка управління дорівнює $x(t)=y_n$, управляючий пристрій видає сигнал $U^+_{\max}=U_{\max}$. Рівняння фазової траєкторії, що відповідає цьому режиму, отримаємо з рівняння (6), підставивши $u(t)=U_{\max}$. Для цього режиму введемо позначення: $x(t)=x_1>0$, $\varepsilon=\varepsilon_1<0$, $-k_0U_{\max}\le\varepsilon_1\le0$. Рівняння (6) набуває вигляду:

$$x_1=-T_0\varepsilon_1+T_0k_0U_{\max}\times\ln|k_0U_{\max}+\varepsilon_1|-T_0k_0U_{\max}\times\ln|k_0U_{\max}|. \tag{7}$$

2. Нехай при $t=0$, $y(t)=0$ на вхід подано сигнал $-y_n$. При цьому похибка управління дорівнює $x(t)=-y_n$, управляючий пристрій видає сигнал $U^-_{\max}=-U_{\max}$. Як і у попередньому випадку, для цього режиму введемо позначення: $x(t)=x_2<0$, $\varepsilon=\varepsilon_2>0$, $0\le\varepsilon_2\le k_0U_{\max}$. Тоді рівняння фазової траєкторії, що відповідає цьому режиму, набуває вигляду:

$$x_2 = -T_0\varepsilon_2 - T_0 k_0 U_{\max} \times \ln\left|-k_0 U_{\max} + \varepsilon_1\right| + T_0 k_0 U_{\max} \times \ln\left|-k_0 U_{\max}\right|. \quad (8)$$

Розглянемо приклад. Нехай параметри об'єкта управління мають значення: $k_0 = 0{,}25;\ T_0 = 0{,}5c;\ U_{\max} = \pm 220B$.

Для побудови та розрахунку лінії перемикання за формулами (7) та (8) скористаємося програмою, наведеною нижче. Програму складено за допомогою платформи MATLAB [5] та представлено у форматі m-file:

```
T=0.5; k=0.25; Um=220; ku=k.*Um
e1=[-55:0.1:0];
x1=-T.*e1-T.*ku.*log(abs(-ku+e1))+T.*ku.*log(ku);
e2=[0:0.1:55];
x2=-T.*e2+T.*ku.*log(abs(+ku+e2))-T.*ku.*log(ku);
e=[e1,e2];
x=[x1,x2];
plot(x,e); grid
```

На рис.2 показано лінія перемикання фазової площини для заданої системи.

На рис.3 показано фазовий портрет замкнутої слідкуючої системи. Лінія перемикання *MON* поділяє всю фазову площину на дві частини. Праворуч від лінії перемикання $u(t) = +U_{\max}$, ліворуч від лінії перемикання $u(t) = -U_{\max}$. Перехідний процес у системі відображається на фазовій площині рухом зображуючої точки фазової траєкторії. Кожна траєкторія залежить від початкових умов. Потрапивши на фазову траєкторію, зображуюча точка рухається строго по ній. Перехід на іншу траєкторію можливий лише лінії перемикання, якою є лінія *MON*. У верхній частині фазової площини $\varepsilon(t) = (dx(t)/dt) > 0$ напрямок руху становить «зліва-направо», у нижній частині — «справа-наліво».

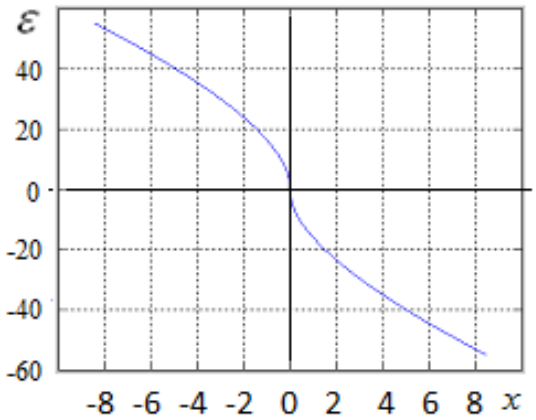



**Рис. 2**. Лінія перемикання на фазовій площині

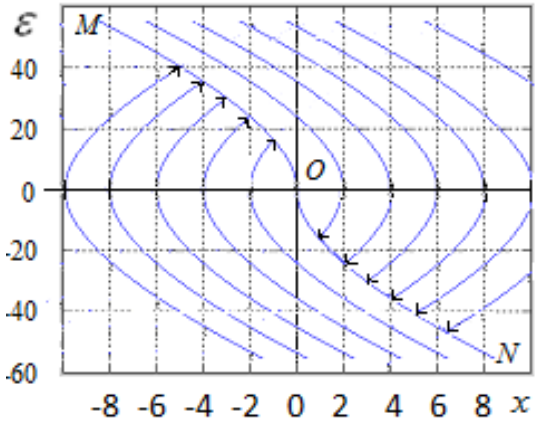



**Рис. 3**. Фазовий портрет слідкуючої системи

Щоб отримати алгоритм функціонування оптимальної за швидкодією системи необхідно забезпечити рух зображуючої точки з максимальною швидкістю по будь-якій траєкторії до перетину з лінією перемикання.

На фазовому портреті видно, що кожна фазова траєкторія перетинається з лінією перемикання. Потрапивши на лінію перемикання, зображуючи точка неминуче рухається до центру координат з максимальною швидкістю. Таким чином, управляючий пристрій повинен вмикати сигнал управління в релейному режимі $u(t) = \pm U_{\max}$ (залежно від початкових умов) і в момент попадання зображуючої точки на лінію перемикання релейний пристрій повинен перемикати

поточний сигнал управління на зворотний за знаком. Такий алгоритм управління можна отримати, якщо порівнювати похибку управління з величиною, що відповідає лінії перемикання, і давати сигнал на перемикання у разі, якщо похибка не дорівнює нулю.

Структурну схему замкнутої оптимальної за швидкодією системи показано на рис.4. На схемі (рис.4) представлено наступні блоки: блок 1 — реле, блок 2 — функціональний блок, який формує лінію перемикання. Цей блок забезпечує залежність вихідної величини блоку (на схемі позначено як $[x]$) від похідної вихідної величини системи $\varepsilon(t)$.

Для того, щоб уникнути можливих автоколивань у релейній системі, потрібно ввести в релейну характеристику зону нечутливості. При цьому *зменшується точність* роботи слідкуючої системи, оскільки вхідні сигнали, які менші за величиною зони нечутливості, така система не сприймає. Тому, щоб, з одного боку — уникнути можливості автоколивань, а з іншого — забезпечити відсутність у системі зони нечутливості, релейну характеристику побудовано за допомогою двох елементів, увімкнутих послідовно: пропорційної ланки з великим коефіцієнтом підсилення та нелінійної ланки типу «насичення».

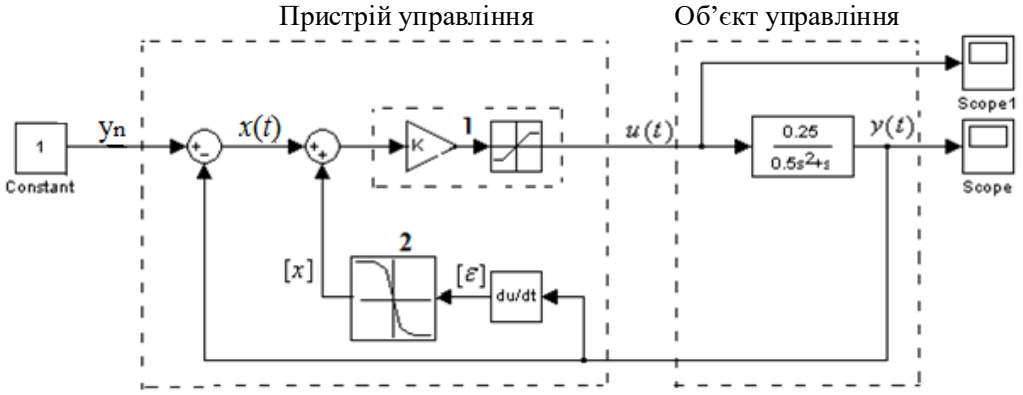



**Рис. 4**. Структурна схема оптимальної за швидкодією слідкуючої системи

Числовим експериментом встановлено [6 — 9], що коефіцієнт посилення пропорційної ланки слід прийняти у діапазоні $\left(10^5 \ldots 10^6\right)$, а величину насичення нелінійної ланки обрати як $u(t) = \pm U_{\max}$.

На рис. 5-7 наведено результат моделювання системи, побудованої у відповідності до структурної схеми, представленої на рис. 4.

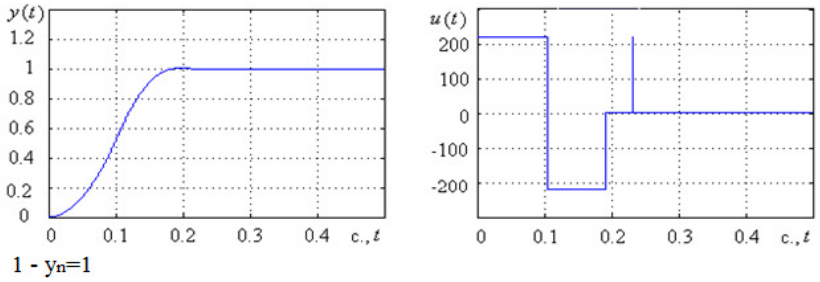



**Рис. 5**. Результа ти моделювання оптимальної за швидкодією системи для $1 - y_n = 1$;

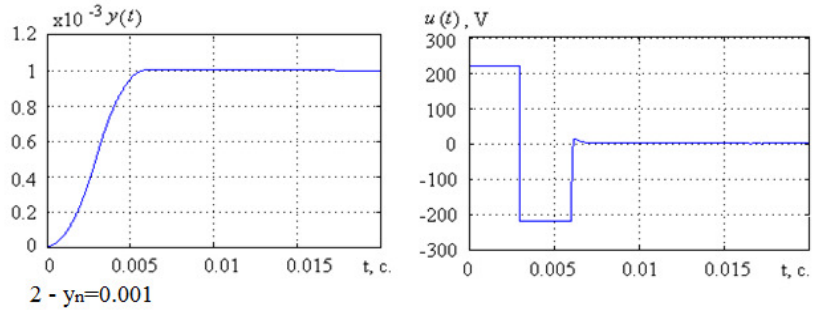



**Рис. 6**. Результати моделювання оптимальної за швидкодією системи для 2 - $y_n = 0{,}001$;

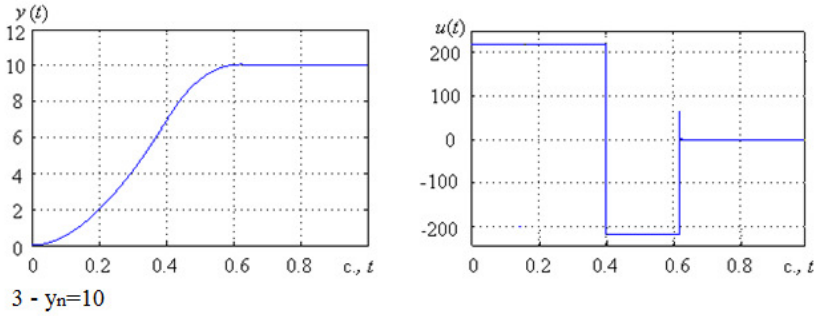



**Рис. 7**. Результати моделювання оптимальної за швидкодією системи для 3 - $y_n = 10$

За графіками рис. 5-7 можна дійти висновку, що система зберігає властивості оптимальної по швидкодією у широкому діапазоні змін задаючого сигналу. Оптимальність процесу підтверджується графіками змін сигналу управління $u(t) = \pm U_{\max}$.

**Висновки.** Розроблено структурну схему і метод розрахунку управляючого пристрою для об'єкта управління другого порядку в слідкуючій системі, яка є оптимальною за швидкодією. Розглянутий метод дозволяє розрахувати оптимальну систему без виконання громіздких розрахунків. У роботі наведено програму, що дозволяє виконати нескладні обчислення для побудови управляючого пристрою, по виду заданого диференціального рівняння незмінної частини системи (суть — об'єкта управління) або за відповідною передаточною функцією.

**Список літератури**

1. Писаренко А.В., Репнікова Н.Б. Оптимальні та адаптивні системи. Методи теорії оптимального керування. К.: НТУУ «КПІ ім. І. Сікорського», 2013. 128с.
2. Луцька Н.М., Ладанюк А.П. Оптимальні та робастні системи керування технологічними об'єктами. К.: Ліра, 2015. 288 с.
3. Albertos P., Sala A. Multivariable control systems: an engineering approach. Valencia: Polytechnic University of Valencia, 2004. 339 p.
4. Краснопрошина А.А., Репникова Н.Б., Ильченко А.А. Современный анализ систем управления с применением MATLAB, Simulink, Control System. К.: Корнійчук, 1999. 144 с.
5. Бобриков С.А., Пичугин Е.Д. Оптимальная настройка ПИ-регулятора с одноёмкостным объектом // Электромашиностроение и электрооборудование. К.: Техніка. 2009, №72. С.179-181.
6. Бобриков С.А., Пичугин Е.Д. Оптимальное цифровое управляющее устройство в системе с запаздыванием при заданном коэффициенте

усиления. *Электротехнические и компьютерные системы*. 2010. № 01 (77). С. 49–52.
7. Бобриков, С.А. Пичугин Е.Д. Оптимальная настройка цифрового регулятора для объекта высокого порядка с запаздыванием. *Електромашинобудування та електрообладнання*. 2010, № 75. С.57–61.
8. Бобриков С.А. Пичугин Е.Д., Бабийчук О.Б. Оптимальная настройка цифрового регулятора для системы управления с астатизмом второго порядка. *Электрические и компьютерные системы*. 2015. №17(93). С.80-86.

## SYNTHESIS AND SIMULATION OF THE OPTIMUM ACCORDING TO THE SPEED OF THE TRACKING SYSTEM

S. O. Bobrikov, L. L. Prokofieva, A. A. Saveliev

National Odesa Polytechnic University , Shevchenko str., 1, Odesa, 65044
bobrikov1932@gmail.com, luleopro@gmail.com, sanp277@gmail.com

A method of construction and calculation of the structural scheme of the facility management system is proposed, which is constructively formed by the serial connection of the first-order aperiodic link and the integrator, and is formally described by a mathematical model in the form of a product of the corresponding transfer functions. An electric motor with an input voltage source can serve as a common typical implementation of the specified control object. The proposed method makes it possible to build a control system that is close to optimal in terms of speed. Moreover, optimality, in this case, is determined by the fact that the object in the process of functioning is turned on in relay mode. The transient process for a defined control object, under the conditions of the optimal mode, is two stages: acceleration and braking, which are carried out at the maximum values of the input influence - that is, the input voltage. To determine the switching moments of the input influence (signal), the phase plane method is used, and the phase coordinates are: on the abscissa axis, the output value (for the given example, the frequency of rotation of the motor shaft), and on the ordinate axis, the rate of change of the output value (or derivative from the rotation frequency of the motor shaft, i.e. the corresponding acceleration). A structural diagram of the control system was developed, which ensures acceleration and braking of the object at the maximum values of the input signal. The optimality of the transition process is ensured by the fact that the control error is compared with the signal coming from the link of the configured switching line. Formulas for calculating the switching line on the phase plane are presented, as well as the program for calculating the switching line, written in the MATLAB environment in m-file format. The system was modeled in the MATLAB-Simulink package, which confirms the optimality of the control process.

**Keywords**: structural diagram, control system, transfer function, optimal system, speed, relay mode, phase plane, switching line, control error, simulation.

DOI 10.15276/imms.v13.no1-2.34
УДК 004.7:519.8

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp.34-47

# ПОРІВНЯЛЬНИЙ АНАЛІЗ МЕТОДІВ РОЗВ'ЯЗАННЯ ОПЕРАТИВНИХ ЗАДАЧ ТРАНСПОРТНОЇ ЛОГІСТИКИ

Д. Р. Горпенко, В.О. Болтьонков

Національний університет «Одеська політехніка»,
пр. Шевченко, 1, Одеса, 65044, Україна;
e-mails: horpenko@op.edu.ua, boltenkov@op.edu.ua

Робота присвячена питанню визначення найкращого логістичного маршруту під час розв'язання оперативних задач транспортної логістики в умовах військового стану командою волонтерів. В роботі проводиться огляд багатокритеріальних методів прийняття рішень, які використовуються для розв'язання оперативних задач транспортної логістики. На вибір найкращого маршруту для перевезення пасажирів або вантажу впливають зміни стану зовнішнього середовища, яке динамічно змінюється під час військових дій. Для оперативної оцінки стану зовнішнього середовища долучаються всі члени команди волонтерів, які володіють актуальною інформацією щодо станів маршрутів. Дана інформація передається координатору команди волонтерів, який і приймає рішення щодо вибору логістичного маршруту. В умовах військового стану можуть динамічно змінюватись як кількість логістичних маршрутів, так і стан маршрутів. Таким чином, методи, які застосовуються координатором волонтерів для підтримки прийняття рішення мають бути стійкими до зміни кількості можливих логістичних маршрутів та їх зміни. Вони також повинні мати невелику обчислювальну складність та підтримувати групове прийняття рішень. В роботі проводилось порівняння методів AHP (Analytic Hierarchy Process) та mSmart на основі таких факторів як: адекватність до змін альтернатив або критеріїв; гнучкість у процесі прийняття рішень; обчислювальна складність; адекватність підтримки групового прийняття рішень; кількість альтернативних маршрутів та критеріїв. Було розглянуто задачу вибору найкращого з можливих маршрутів з Одеси до Херсону. Результати показали, що обидва метода підходять для розв'язання задачі вибору найкращого маршруту, але метод mSmart виявився кращим за оцінкою розглянутих в роботі факторів.

**Ключові слова:** багатокритеріальні методи прийняття рішень; транспортна логістика; команда волонтерів; AHP; mSmart

**Вступ.** Однією з задач транспортної логістики є визначення маршрутів доставки ресурсів [1]. Якщо розглядається задача визначення маршрутів за стаціонарних умов транспортування, це є задача пошуку оптимального маршруту [2].У разі динамічних змін умов транспортування, які виникають внаслідок військових дій, маємо задачу оперативної транспортної логістики в якій необхідно враховувати різні фактори такі як небезпека, швидкість доставки, військові ризики. При прийняті рішень в таких умовах використовуються багатокритеріальні методи прийняття рішень [3], які дозволяють враховувати різні фактори та визначити найкращий маршрут з можливих. Волонтерські організації відігрівають важливу роль у вирішенні транспортно-логістичних завдань в умовах воєнних дій. До основних завдань, які вони виконують відносяться: евакуація та транспортування людей до безпечних місць; транспортування гуманітарних вантажів до постраждалих районів, де вони потрібні найбільше [4]. Згідно з концептуальною моделлю, запропонованою в [5] волонтери, які знаходяться на маршрутах є важливими джерелами інформації для

координації дій та прийняття рішень логістичних задач. Вони систематично збирають інформацію про стан доріг, об'єктів, безпеку та інші фактори, які можуть вплинути на логістичні операції. Ця інформація передається координатору команди волонтерів. В сучасних умовах волонтери можуть використовувати технології, а саме мобільні додатки для передачі оперативної інформації в координаційний центр, де координатор команди волонтерів аналізує отриману інформацію та приймає рішення, щодо вибору найкращого маршруту з можливих. Мобільні додатки, в яких реалізовані багатокритеріальні методи прийняття рішень, дозволять координатору команди волонтерів отримати допомогу під час прийняття рішення. Проте, для таких додатків необхідні методи, які мають низьку обчислювальну складність, методи мають бути стійкими до зміни кількості можливих логістичних маршрутів та підтримувати групове прийняття рішень [6].

**Аналіз літературних джерел та постановка проблеми.** Багатокритеріальні методи вирішення логістичних задач в умовах військових дій дозволяють враховувати різні аспекти та критерії при прийнятті рішень. Існує низка багатокритеріальних методів, які можуть бути застосовані для рішення логістичних задач [7-8]. Метод аналізу ієрархій (Analytic Hierarchy Process, AHP) [9] достатньо часто використовується для рішення задач транспортної логістики [10-11]. AHP використовує ієрархічну структуру для оцінки важливості різних критеріїв та альтернатив. Цей метод дозволяє визначити вагомість кожного критерію та порівняти альтернативи на основі їх відповідності критеріям. Розроблено модифікації методу AHP, які також використовуються для рішення задач транспортної логістики. Наприклад, метод аналізу ієрархій за допомогою комплексного вагового коефіцієнта (Analytic Network Process, ANP) є розширенням AHP дозволяє моделювати взаємозв'язки між елементами ієрархії [12]. Цей метод корисний у випадках, коли вирішується складна логістична задача з взаємодіючими критеріями. Метод Fuzzy ANP [13] доповнює ANP за допомогою нечіткої логіки, яка використовуються для оцінки взаємозв'язків між елементами ієрархії, а також для призначення ваг критеріям і альтернативам. Застосування методу Fuzzy ANP в логістичних задачах воєнних дій може допомогти вирішити складні та багатокритеріальні проблеми, де потрібно враховувати не тільки числові оцінки, але й нечіткі та неоднозначні фактори. Він дозволяє враховувати різноманітність даних, неоднорідність та нечіткість у процесі прийняття рішень, що є важливим в умовах військових дій. Метод VIKOR (VlseKriterijumska Optimizacija I Kompromisno Resenje) [14] є багатокритеріальним методом прийняття рішень, який дозволяє знаходити компромісні розв'язки для задач оптимізації з кількома конфліктуючими критеріями. Метод VIKOR може бути застосований у різних галузях, включаючи логістичні задачі в умовах воєнних дій. Він дозволяє знайти компромісні розв'язки, які враховують багато критеріїв, такі як вартість, час, ефективність та інші, що допомагає забезпечити ефективну логістику в умовах воєнного конфлікту. Метод Fuzzy VIKOR [13] є розширенням методу VIKOR, який враховує нечіткість та неоднозначність даних при вирішенні багатокритеріальних задач. У методі Fuzzy VIKOR застосовуються нечіткі числа для представлення такої інформації, як оцінки критеріїв та альтернатив. Цей метод дозволяє моделювати неоднозначність та невизначеність у вирішенні задачі, що може бути корисним в умовах воєнних дій, де доступність точних даних може бути обмеженою або неповною. Метод TOPSIS (Technique for Order Preference by Similarity to Ideal Solution, TOPSIS) [15] TOPSIS використовує поняття "ідеального" та "найгіршого" розв'язку для оцінки альтернатив. Цей метод ранжує альтернативи на основі їх відстані до ідеального розв'язку, а також до найгіршого розв'язку. Основна ідея

методу Fuzzy TOPSIS [16] полягає в тому, щоб використовувати нечіткі числа або нечіткі множини для представлення невизначеності в даних. Замість точних значень критеріїв та оцінок альтернатив, використовуються нечіткі числа або лінгвістичні терміни, які виражають ступінь нечіткості. Метод SMART (Simple Multi-Attribute Rating Technique) [17] є одним з методів багатокритеріального прийняття рішень, використанням мобільного додатку який використовується для ранжування альтернатив за допомогою вагових коефіцієнтів для кожного критерію. Основна ідея методу SMART полягає в тому, щоб оцінити кожну альтернативу за допомогою кількох критеріїв і присвоїти вагові коефіцієнти цим критеріям відповідно до їх важливості. Далі проводиться математичне обчислення для кожної альтернативи з урахуванням оцінок критеріїв та їх вагових коефіцієнтів. Метод SMART є простим і широко використовується для прийняття рішень в багатьох сферах, де потрібно враховувати багато критеріїв і вагові коефіцієнти. Він допомагає систематизувати процес прийняття рішень і забезпечує об'єктивність та структурованість в оцінці альтернатив. В роботі [18] розроблено модифікація методу SMART (mSmart) на основі поєднання методу SMART та методу TOPSIS. Цей метод дозволяє обробляти дані різного типу, такі як кількісні, якісні та бінарні («так»/«ні»). Пропонується будувати матрицю розв'язків, використовуючи підхід методу TOPSIS. Для нормалізації кількісних елементів матриці розв'язків враховується тип дії над кожним критерієм. Якщо максимізується значення критерію, то застосовується певна формула нормалізації. В [5] досліджено багатокритеріальні методи прийняття рішень AHP, MAHP [19], ELECTRE [20], TOPSIS, SMART та mSmart з точки зору можливості обробки даних різних типів та мінімального часу введення даних (кількість матриць парних порівнянь). Встановлено, що методи TOPSIS, SMART та mSmart є найкращими з точки зору мінімального часу введення даних, оскільки вони вимагають побудови лише однієї матриці парних порівнянь. Методи TOPSIS та ELECTRE допускають обробку тільки кількісних даних. Проте, методи AHP та MAHP вимагають побудови декількох матриць парних порівнянь, в залежності від кількості альтернатив, але ці методи дозволяють обробку даних різного типу і надають високу достовірність отриманих рішень.

На основі проведеного аналізу багатокритеріальним методом прийняття рішень, в рамках даної статті запропоновано виконати порівняння методів AHP та mSmart, враховуючи такі фактори як: адекватність до змін альтернатив або критеріїв; гнучкість у процесі прийняття рішень; обчислювальна складність; адекватність підтримки групового прийняття рішень; кількість альтернативних маршрутів та критеріїв [6].

**Мета роботи.** Метою даної роботи є проведення порівняльного аналізу методів AHP та mSmart для розв’язання оперативних задач транспортної логістики, що дозволить визначити метод з для підтримки прийняття рішень координатора команди волонтерів.

Для досягнення поставленої мети були сформульовані наступні задачі:

– визначити дані для задачі вибору найкращого маршруту для команди волонтерів в умовах динамічних змін зовнішнього середовища;

– розв’язати поставлену задачу методами AHP та mSmart;

– виконати порівняння методів AHP та mSmart на основі факторів, вказаних в [6].

**Основна частина.** Розглянемо задачу доставки ресурсу з Одеси до Херсону. Було обрано чотири маршрути (див. рис. 1). Кожен маршрут розбито на ділянки за якими закріплено експерт-волонтер. Так, перший (довжина 231 км) та другий (довжина 232 км) маршрути розбито на 13 ділянок, третій (довжина 201 км) та четвертий (довжина

297 км) на 12 ділянок. На рисунку 5 показані вікна, які відображають перше вікно користувацького інтерфейсу мобільного додатку трьох експертів-волонтерів з внесеними даними щодо стану відповідної ділянки першого маршруту: якості дорожнього покриття, наявності небезпеки, можливості заправки та ремонту техніки.

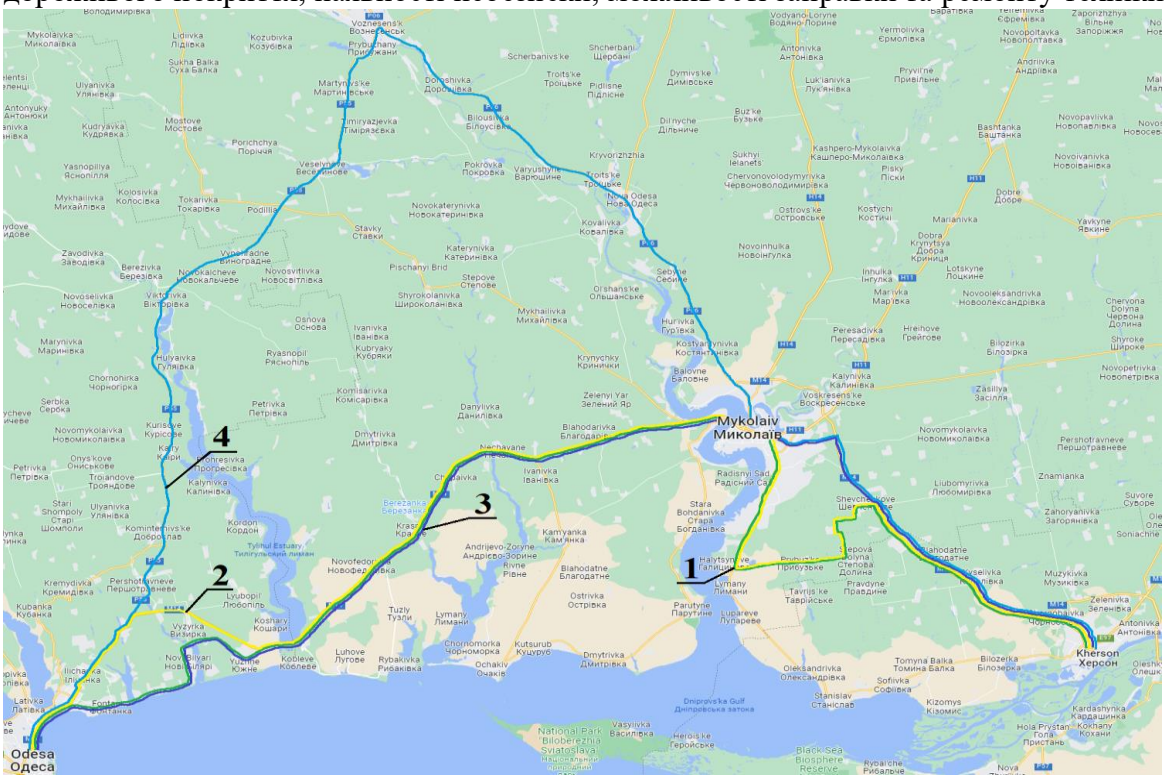



**Рис. 1.** Логістичні маршрути з Одеси до Херсону [21]

Для вибору відповідного маршруту було обрано було визначено наступні критерії, які представляються різними типами даних:

– Відстань між точками відправки до точки потреби ресурсу (критерій С1.
– Середній час доставки ресурсу (критерій С2);
– Якість дорожнього покриття (критерій С3);
– Наявність небезпеки (критерій С4);
– Можливість дозаправки (критерій С5);
– Можливість ремонту техніки (критерій С6).

В таблиці 1 представлено зведені дані за критеріями отриманими від волонтерів експертів та координатора команди волонтерів.

**Таблиця 1**

Дані для рішення логістичного завдання

| | С1 (відстань) | С2 (час) | С3 (якість дороги) | С4 (рівень небезпеки) | С5 (дозаправка) | С6 (ремонт) |
|---|---|---|---|---|---|---|
| А1 | 231 | 3.54 | задовільне | середній | в наявності | немає |
| А2 | 232 | 4 | задовільне | низький | немає | в наявності |
| А3 | 201 | 2.3 | задовільне | середній | немає | немає |
| А4 | 297 | 3.3 | добре | середній | в наявності | в наявності |

Застосуємо метод AHP для розв'язку поставленої задачі. Представимо нашу задачу у вигляді ієрархії: Мета – Критерії-Альтернативи (рис. 2).

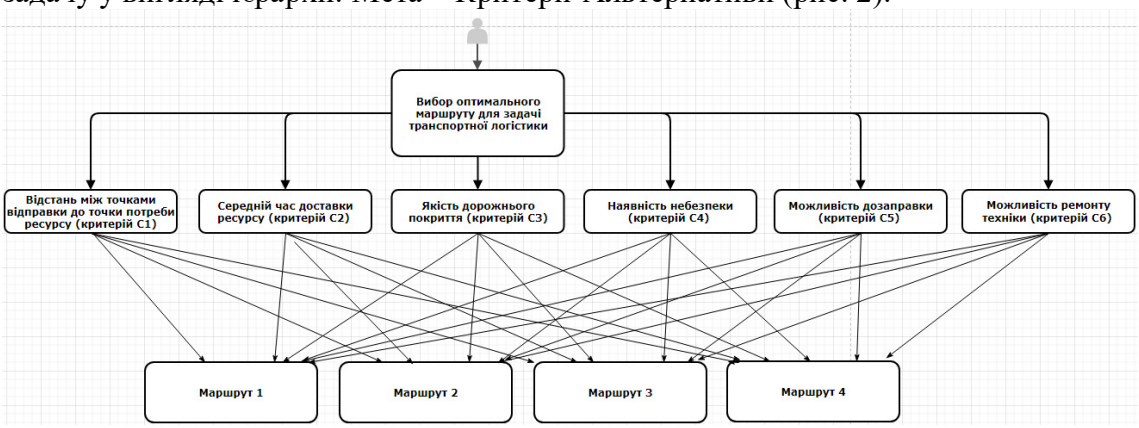



**Рис.2.** Ієрархії для задачі вибору найкращого маршруту

На рисунку 3 представлено попарне порівняння критеріїв та альтернатив за критеріями.

**Попарне порівняння критеріїв**

| Мета | C1 | C2 | C3 | C4 | C5 | C6 |
|---|---|---|---|---|---|---|
| C1 | 1 | 3 | 7 | 6 | 3 | 3 |
| C2 | 1/3 | 1 | 1/5 | 1/7 | 3 | 3 |
| C3 | 1/7 | 5 | 1 | 3 | 5 | 5 |
| C4 | 1/6 | 7 | 1/3 | 1 | 5 | 5 |
| C5 | 1/3 | 1/3 | 1/5 | 1/5 | 1 | 1 |
| C6 | 1/3 | 1/3 | 1/5 | 1/5 | 1 | 1 |

**Попарне порівняння альтернатив по відношенню до критеріїв**

| C1 | A1 | A2 | A3 | A4 |
|---|---|---|---|---|
| A1 | 1 | 3 | 1/5 | 5 |
| A2 | 1/3 | 1 | 1/5 | 3 |
| A3 | 3 | 5 | 1 | 5 |
| A4 | 1/5 | 1/3 | 1/5 | 1 |

| C2 | A1 | A2 | A3 | A4 |
|---|---|---|---|---|
| A1 | 1 | 3 | 1/5 | 1/3 |
| A2 | 1/3 | 1 | 1/5 | 1/3 |
| A3 | 5 | 5 | 1 | 3 |
| A4 | 3 | 3 | 1/3 | 1 |

| C3 | A1 | A2 | A3 | A4 |
|---|---|---|---|---|
| A1 | 1 | 1/3 | 3 | 3 |
| A2 | 5 | 1 | 5 | 3 |
| A3 | 1/5 | 1/5 | 1 | 1/3 |
| A4 | 1/3 | 1/3 | 3 | 1 |

| C4 | A1 | A2 | A3 | A4 |
|---|---|---|---|---|
| A1 | 1 | 1/3 | 3 | 3 |
| A2 | 3 | 1 | 3 | 3 |
| A3 | 1/3 | 1/3 | 1 | 1 |
| A4 | 1/3 | 1/3 | | 1 |

| C5 | A1 | A2 | A3 | A4 |
|---|---|---|---|---|
| A1 | 1 | 1 | 1/3 | 1/5 |
| A2 | 1 | 1 | 1/3 | 1/5 |
| A3 | 3 | 3 | 1 | 1/3 |
| A4 | 5 | 5 | 3 | 1 |

| C6 | A1 | A2 | A3 | A4 |
|---|---|---|---|---|
| A1 | 1 | 1 | 3 | 1/3 |
| A2 | 1 | 1 | 3 | 1/3 |
| A3 | 1/3 | 1/3 | 1 | 1/3 |
| A4 | 3 | 3 | 3 | 1 |

**Рис.3.** Матриці попарних порівнянь для двох рівнів ієрархії

Отримано наступні значення глобальних оцінок альтернатив: S2= 0,476; S1= 0,206; S4= 0,176; S3= 0,141. Після впорядкування альтернатив за глобальними оцінками, отримано наступне ранжування: $A2 \succ A1 \succ A4 \succ A3$.

На рисунку 4 показано значення глобальних оцінок альтернатив.

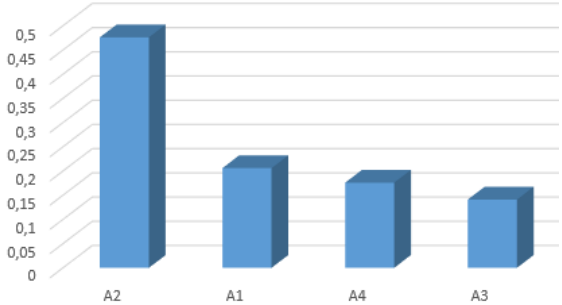



**Рис. 4.** Значення глобальних оцінок альтернатив

Знайдемо рішення поставленої задачі методом mSmart. Після внесення значень критеріїв Відстань та Час в матрицю рішень виконано перехід до методу mSmart [15]з кількісними даними (табл.2).

**Таблиця 2**

Кількісні дані для рішення завдання

| | C1 (відстань) | C2 (час) | C3 (якість дороги) | C4 (рівень небезпеки) | C5 (дозаправка) | C6 (ремонт) |
|---|---|---|---|---|---|---|
| A1 | 231 | 3.54 | 0,6 | 0,46 | 0,512 | 0,435 |
| A2 | 232 | 4 | 0,66 | 0,31 | 0,513 | 0,434 |
| A3 | 201 | 2.3 | 0,46 | 0,5 | 0,528 | 0,415 |
| A4 | 297 | 3.3 | 0,53 | 0,5 | 0,579 | 0,488 |

При застосуванні методу mSmart до вирішення задачі знаходження найкращого маршруту отримано наступне ранжування альтернатив: $A2 \succ A1 \succ A3 \succ A4$. На рис. 5 показано значення глобальних оцінок альтернатив.

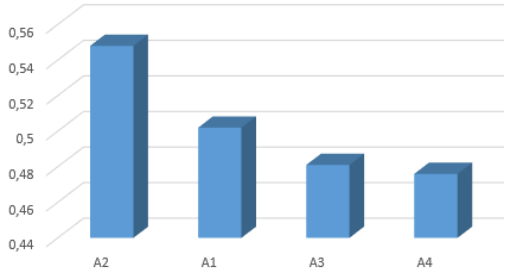



**Рис. 5**. Ранжування альтернатив, отримане за методом mSmart

**Порівняльний аналіз методів AHP та mSmart.** Для порівняння методів AHP та mSmart були розглянуті фактори, запропоновані в [6]:

1. адекватність до змін альтернатив або критеріїв;
2. гнучкість у процесі прийняття рішень;
3. обчислювальна складність;
4. адекватність підтримки групового прийняття рішень;
5. кількість альтернативних маршрутів та критеріїв.

**Адекватність до змін альтернатив.** Під час прийняття рішення координатором волонтерів, щодо вибору маршруту перевезення гуманітарної допомоги або людей в умовах військових дій може змінюватись кількість доступних маршрутів транспортування. Бажано, щоб метод був стійким до зміни кількості альтернатив.

Під час розв'язку визначення найкращого маршруту транспортування командою волонтерів з м. Одеса до м. Херсон під час використання методу AHP отримано наступне ранжування альтернатив:

$A2 \succ A1 \succ A4 \succ A3$.

Для перевірки стійкості методу щодо зміни кількості альтернатив, було введено додатковий п’ятий альтернативний маршрут. В результаті експерименту додаткова альтернатива A5 покроково обиралась рівною однієї з існуючих альтернатив.

У випадку, коли альтернатива A5 дорівнювала альтернативі A1, отримано ранжування $A2 \succ A5=A1 \succ A4 \succ A3$, значення глобальних оцінок альтернатив у цьому випадку показано на рисунку 6а.

У випадку, коли альтернатива A5 дорівнювала альтернативі A2, отримано ранжування $A2=A5 \succ A1 \succ A4 \succ A3$, значення глобальних оцінок альтернатив у цьому випадку показано на рисунку 6б.

У випадку, коли альтернатива A5 дорівнювала альтернативі A3, отримано ранжування $A2 \succ A1 \succ A4 \succ A3=A5$, значення глобальних оцінок альтернатив у цьому випадку показано на рисунку 6в.

У випадку, коли альтернатива A5 дорівнювала альтернативі A4, отримано ранжування $A2 \succ A1 \succ A4=A5 \succ A3$, значення глобальних оцінок альтернатив у цьому випадку показано на рисунку 6г.

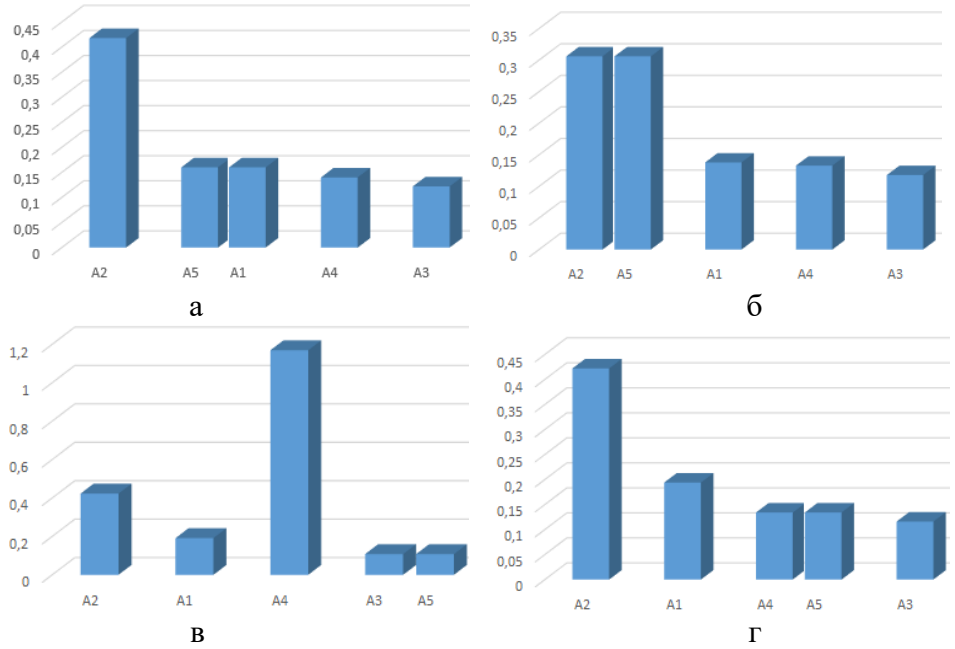



**Рис. 6.** Ранжування альтернатив при дослідженні адекватності до змін альтернатив методу AHP

В результаті дослідження встановлено, що у всіх випадках найкращою є альтернатива A2 і порядок ранжування не змінився, тобто метод AHP є стійким щодо зміни альтернатив.

Проведено дослідження методу mSmart на стійкість щодо зміни кількості альтернатив так як ми це зробили для методу AHP. При застосуванні методу mSmart до вирішення задачі знаходження найкращого маршруту було отримано наступне ранжування альтернатив: $A2 \succ A1 \succ A3 \succ A4$.

У випадку, коли альтернатива A5 дорівнювала альтернативі A1, отримано ранжування $A2 \succ A1=A5 \succ A3 \succ A4$, значення глобальних оцінок альтернатив у цьому випадку показано на рисунку 7а.

У випадку, коли альтернатива A5 дорівнювала альтернативі A2, отримано ранжування $A2= A5 \succ A1 \succ A3 \succ A4$, значення глобальних оцінок альтернатив у цьому випадку показано на рисунку 7б.

У випадку, коли альтернатива A5 дорівнювала альтернативі A3, отримано ранжування $A2 \succ A1 \succ A3=A5 \succ A4$, значення глобальних оцінок альтернатив у цьому випадку показано на рисунку 7в.

У випадку, коли альтернатива A5 дорівнювала альтернативі A4, отримано ранжування $A2 \succ A1 \succ A3 \succ A4=A5$, значення глобальних оцінок альтернатив у цьому випадку показано на рисунку 7г.

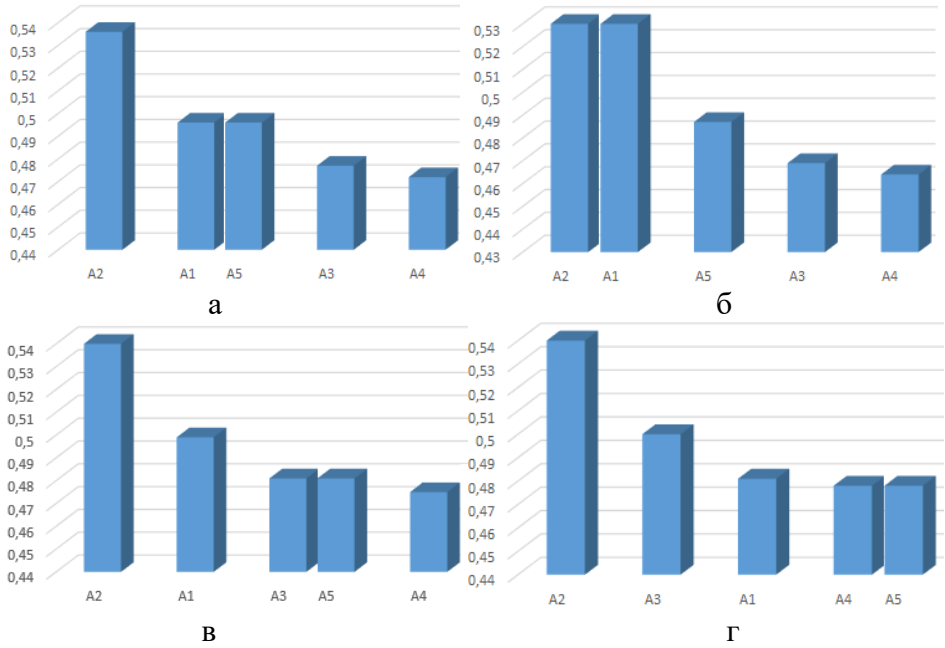



**Рис. 7**. Ранжування альтернатив при дослідженні адекватності до змін альтернатив методу mSmart

В результаті дослідження встановлено, що у всіх випадках найкращою є альтернатива A2 і порядок ранжування не змінився, отже метод mSmart також виявився стійким щодо зміни кількості альтернатив.

**Адекватність до змін критеріїв.** Аналогічне дослідження проведено щодо зміни кількості критеріїв. Будемо послідовно добавляти додатковий критерій C7, якій буде дорівнювати існуючим критеріям, і перевіримо, як внесення додаткової альтернативи буде впливати на ранжування альтернатив.

Застосування методу AHP. Отримано наступне ранжування: $A2 \succ A1 \succ A4 \succ A3$.

$A2 \succ A1 \succ A3 \succ A4$, якщо C1=C7 (з відповідними значеннями глобальних оцінок [0,456 0,209 0,69 0,166]);

$A2 \succ A1 \succ A4 \succ A3$, якщо C2=C7 (з відповідними значеннями глобальних оцінок [0,459 0,208 0,171 0,162]);

$A2 \succ A1 \succ A4 \succ A3$, якщо C3=C7 (з відповідними значеннями глобальних оцінок [0,507 0,211 0,167 0,115]);

$A2 \succ A1 \succ A4 \succ A3$, якщо C4=C7 (з відповідними значеннями глобальних оцінок [0,514 0,205 0,157 0,124]);

$A2 \succ A4 \succ A1 \succ A3$, якщо C5=C7 (з відповідними значеннями глобальних оцінок [0,447 0,203 0,199 0,151])

$A2 \succ A1 \succ A4 \succ A3$, якщо C6=C7 (з відповідними значеннями глобальних оцінок [0,454 0,206 0,202 0,138]).

Отримано, що у двох випадках коли: C1=C7, C5=C7 альтернатива A2 залишилась найкращою, але в першому випадку пріоритетність альтернатив A3, A4 змінилася. У другому випадку альтернатива A4 виявилась найкращою ніж альтернативи A1 та A3.

Розглянемо метод mSmart. Також послідовно додаємо критерій C7, якій буде дорівнювати існуючим альтернативам. Отримано наступне ранжування альтернатив:

A2 ≻ A1 ≻ A3 ≻ A4, якщо C1=C7 (з відповідними значеннями глобальних оцінок [0,544 0,504 0,493 0,465]);
A2 ≻ A1 ≻ A3 ≻ A4, якщо C2=C7 (з відповідними значеннями глобальних оцінок [0,536 0,499 0,494 0,478]);
A2 ≻ A1 ≻ A4 ≻ A3, якщо C3=C7 (з відповідними значеннями глобальних оцінок [0,507 0,211 0,167 0,115]);
A2 ≻ A1 ≻ A3 ≻ A4, якщо C4=C7 (з відповідними значеннями глобальних оцінок [0,567 0,499 0,474 0,470]);
A2 ≻ A1 ≻ A4 ≻ A3, якщо C5=C7 (з відповідними значеннями глобальних оцінок [0,539 0,499 0,484 0,482]);
A2 ≻ A1 ≻ A4 ≻ A3, якщо C6=C7 (з відповідними значеннями глобальних оцінок [0,540 0,500 0,485 0,479]).

Отже у двох випадках коли: C3=C7, C5=C7, C6=C7 альтернатива A2 залишилась найкращою далі йде альтернатива A1, але змінилась приорітетність альтернатив A3 та A4.

Таким чином, у разі зміни кількості критеріїв оба методи не є стійкими, але метод mSmart виглядає більш стійким, так як пріоритетність перших двох альтернатив залишається незмінною.

**Гнучкість у процесі прийняття рішень.** В [6] запропоновано для оцінювання гнучкості у процесі прийняття рішень підраховувати кількість оцінок, які виконують експерти під час застосування методів MCDM.

Розглядається задача знаходження найкращої альтернативи *Ai* з n можливих, для вибору якої експерти мають оцінити кожну альтернативу за m критеріями *Cj*.

Підрахуємо показник гнучкості у процесі прийняття рішення під час застосування методу метода AHP.

В методі AHP експертами будується *n+1* матриця парних порівнянь [7]. Матриця парних порівнянь має властивість зворотної симетричності, тобто $a_{ji} = 1/a_{ij}$ , де $a_{ij}$ $a_{ij}$ – елемент матриці порівнянь $i, j = 1,..., n$ (для матриці парних порівнянь альтернатив) або $i, j = 1,..., m$ (для матриці парних порівнянь критеріїв).

На першому етапі проводиться структуризація задачі у вигляді ієрархічної структури з декількома рівнями: мета – критерії – альтернативи. Ієрархія будується з вершини, через проміжні рівні, до найнижчого рівня. Для другого рівня ієрархії порівнюються критерії між собою по відношенню до загальної мети і будується одна матриця парних порівнянь розміру $m \times m$. В силу зворотної симетричності матриці парних порівнянь експертами для порівняння критеріїв виконується $m(m-1)/2$ оцінок.

Для третього рівня ієрархії порівнюються *n* альтернатив *Ai* по відношенню до кожного з *m* критеріїв *Cj*. В силу зворотної симетричності матриці парних порівнянь експертами для порівняння альтернатив виконується $mn(n-1)/2$ оцінок.

Таким чином, отримуємо, що показник гнучкості *J* у процесі прийняття рішення під час застосування методу метода AHP дорівнює:

$$J_{AHP} = m(m-1)/2 + mn(n-1)/2 . \quad (1)$$

Для розрахунку показника гнучкості у процесі прийняття рішення під час застосування методу метода mSmart будемо враховувати те, що експерти призначають ваги *m* критеріями *Cj*, для кожного з критеріїв обирають дію над m критеріями (максимізація, мінімізація) і будують одну матрицю рішень [15], яка містить оцінки n альтернатив *Ai* за *m* критеріями *Cj*.

Таким чином, отримуємо, що показник гнучкості у процесі прийняття рішення під час застосування методу метода mSmart дорівнює:

$$J_{mSmart} = 2m + mn = m(n+2). \quad (2)$$

Було проведено дослідження показника гнучкості у разі застосування методів AHP та mSmart, кількість альтернатив та критеріїв змінювалась від 2 до 9 (рис. 8). Отримано, що при $n$=2 (кількість альтернатив) для $m$=2…6 та $n$=3, $m$=3.. 4 показник гнучкості у разі застосування методу AHP краще ніж для методу mSmart. При $n$=2, $m$=7 та $n$=3, $m$=5 показник гнучкості у разі застосування методів AHP та mSmart однаковий. У всіх інших випадках показник гнучкості у разі застосування методу mSmart краще ніж для методу AHP.

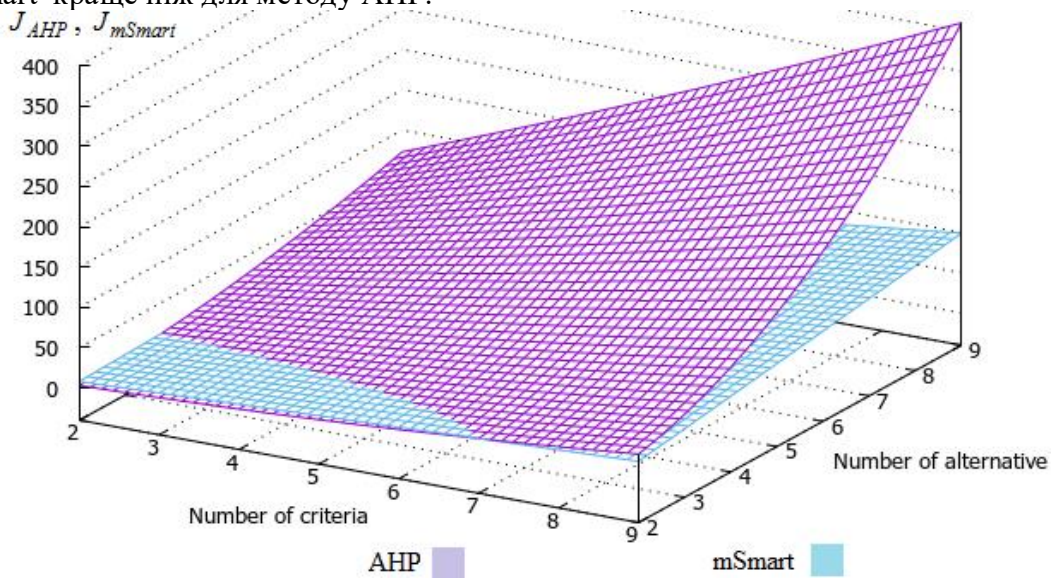



**Рис. 8.** Показник гнучкості для методів AHP та mSmart

Для розглянутої в роботі задачі $n$=4, $m$=6 показник гнучкості для методу AHP склав 51 а для методу mSmart - 36.

**Обчислювальна складність.** Обчислювальна складність $T$ в роботі розраховувалась, як запропоновано в роботі [6], а саме враховувалось кількість операцій множення, ділення, зведення в ступінь.

В методі AHP під час обчислення власного вектору для матриці попарних порівнянь другого рівня (порівняння критеріїв) операція множення виконується $m(m-1)$ разів, а з ведення у ступінь $m$ разів.

Під час обчислення власного вектору для матриці попарних порівнянь третього рівня (порівняння альтернатив) відповідно до $n$ критеріїв, операція множення виконується $mn(n-1)$ разів, а зведення у ступінь $nm$ разів.

На етапі обчислення глобальних пріоритетів для альтернатив операція множення виконується $nm$ разів.

Таким чином, отримуємо, що показник обчислювальної складності у процесі прийняття рішення під час застосування методу метода AHP дорівнює:

$$T_{AHP} = m(m-1) + m + mn(n-1) + nm + nm = m^2 + mn(n+1). \quad (3)$$

В методі mSmart відбувається нормалізація ваг критеріїв під час чого операція ділення виконується $m$ разів. Під час нормування елементів матриці рішень операция ділення і зведення в ступінь виконується $mn$ разів віддповідно. На етапі обчислення глобальних пріоритетів для альтернатив операція множення виконується $nm$ разів.

Таким чином, отримуємо, що показник обчислювальної складності у процесі прийняття рішення під час застосування методу метода mSmart дорівнює:

$$T_{Smart} = m + 2mn + nm = m(3n+1)\,. \quad (4)$$

Було проведено дослідження показника обчислювальної складності у разі застосування методів AHP та mSmart, кількість альтернатив та критеріїв змінювалась від 2 до 9 (рис. 9).

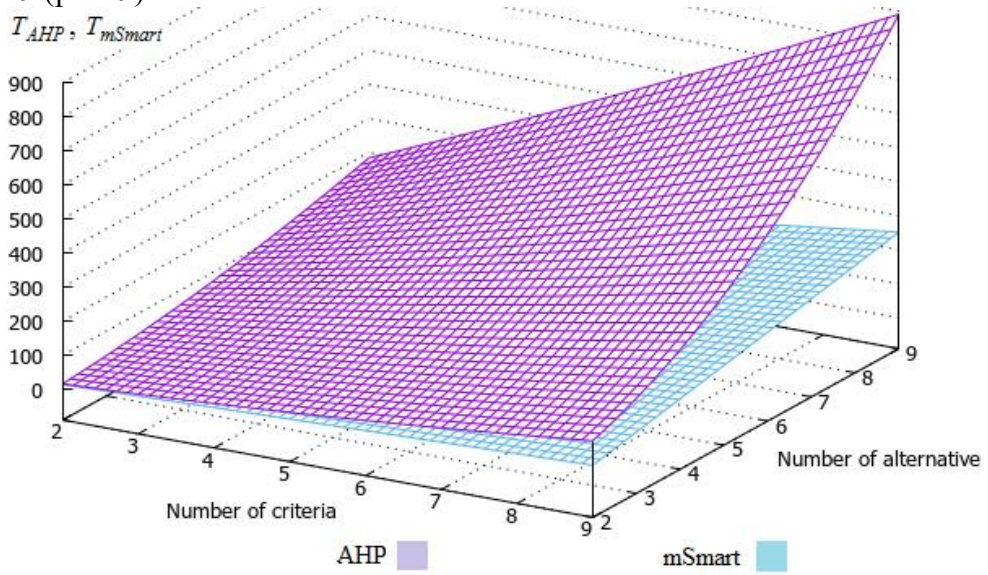



**Рис. 9.** Показник обчислювальної складності для методів AHP та mSmart

Отримано, що всіх інших випадках показник обчислювальної складності у разі застосування методу mSmart нижче ніж для методу AHP в 1,14-3,54 разів. В залежності від кількості альтернатив та критеріїв

Для розглянутої в роботі задачі $n$=4, $m$=6 показник обчислювальної складності для методу AHP склав 156 а для методу mSmart – 78, що нижче в 2 рази.

**Адекватність підтримки групового прийняття рішень.** Обидва метода AHP та mSmart враховують судження більш ніж однієї людини яка приймає рішення. В методі AHP та mSmart агрегація суджень ОПР виконується шляхом обчислення середнього арифметичного суджень. Але у разі збільшення кількості ОПР буде зростати обчислювальна складність, тому краще використовувати метод mSmart.

**Кількість альтернативних маршрутів та критеріїв.** Як зазначалося в [6-7] при використанні методу AHP існує обмеження, щодо кількості альтернатив та критеріїв. Саати [7] передбачає, що кількість критеріїв або альтернатив повинні бути обмежені дев'ятьма, щоб не порушувати людське судження та послідовність оцінок.

**Висновки.** Сформульовано багатокритеріальну задачу вибору найкращого маршруту. На основі цієї задачі було порівняно методі AHP та mSmart. Запропоновані та розглянуті фактори для порівняння методів AHP та mSmart. В результаті дослідження адекватності до змін альтернатив отримали, що у всіх випадках найкращою є альтернатива A2 і порядок ранжування не змінився, тобто методи AHP та mSmart є стійкими щодо зміни альтернатив. В результаті дослідження адекватності до змін критеріїв отримали що, у разі зміни кількості критеріїв оба методи не є стійкими, але метод mSmart виглядає більш стійким, так як пріоритетність перших двох альтернатив залишається незмінною. В результаті дослідження гнучкості у процессі прийняття рішень було виявленно що показник

гнучкості був кращим у метода AHP тільки коли кількість альтернатив не перевищувала 2. В інших випадках показник метода mSmart був такий самий, або був кращій ніж показник для методу AHP. Також було проведено дослідження показника обчислювальної складності у разі застосування методів AHP та mSmart коли кількість альтернатив та критеріїв змінювалась від 2 до 9. Отримано, що показник обчислювальної складності у разі застосування методу mSmart нижче ніж для методу AHP в 1,14-3,54 разів в залежності від кількості альтернатив та критеріїв. В аналізі адекватності підтримки групового прийняття рішень було виявлено що у разі збільшення кількості ОПР буде зростати обчислювальна складність, тому краще використовувати метод mSmart. Щоб не порушувати людське судження та послідовність оцінок Саати передбачає, що кількість критеріїв або альтернатив повинні бути обмежені дев'ятьма. На відміну від AHP у метода mSmart немає такого обмеження у кількості альтернатив.

**Список літератури**


1. Панасюк Н.В., Лютяниця Д.В. Логістичне обслуговування в сфері міжнародних перевезень. *Актуальні проблеми сучасного бізнесу: обліково-фінансовий та управлінський аспекти:* матеріали І Міжнародної науково-практичної інтернет конференції, 19-21 березня 2019 р. Ч. 2. Львів: ЛНАУ, 2019. 274-276. URL: http://surl.li/irxoz
2. Christopher M.A. Multicriteria optimization approach to deploy humanitarian logistic operations integrally during floods. *International Transactions in Operational Research.* 2018. V.25. No.3. P.1053-1079. DOI: https://doi.org/10.1111/itor.12508
3. Moslem S., Saraji M.K., Mardani A., Alkharabsheh A., Duleba S., Esztergár-Kiss D. A Systematic Review of Analytic Hierarchy Process Applications to Solve Transportation Problems: From 2003 to 2022. *IEEE Access.* 2023. V.11. P.11973-11990. DOI: https://doi.org/10.1109/ACCESS.2023.3234298.
4. Закон України «Про волонтерську діяльність». URL: https://zakon.rada.gov.ua/laws/show/3236-17#Text.
5. Horpenko D.R. A conceptual model of decision-making support of the volunteer team in conditions of dynamic changes. *Herald of Advanced Information Technology*. 2022. V.5. No.4. P.275–286. DOI: https://doi.org/10.15276/hait.05.2022.20.
6. Lima F.R., Lauro O., Carpinetti L.C.R. A comparison between Fuzzy AHP and Fuzzy TOPSIS methods to supplier selection. *Applied soft computing*. 2014. V.21. P.194-209. DOI: https://doi.org/10.1016/j.asoc.2014.03.014
7. Zlaugotne B., Zihare L., Balode L., Kalnbalkite A., Khabdullin A., Blumberga D. Multi-Criteria Decision Analysis Methods Comparison. *Environmental and Climate Technologies*. 2020; V. 24, No. 1, P.454–471. DOI: https://doi.org/10.2478/rtuect-2020-0028
8. Beskorovainyi V., Draz O. Mathematical Models Of Decision Support In The Problems of Logistics Networks Optimization. *Innovative Technologies and Scientific Solutions for Industries.* 2021. V.18. No.4. P. 5-14. DOI: https://doi.org/10.30837/ITSSI.2021.18.005
9. Saaty T.L., The Analytic Hierarchy Process, first ed., McGraw Hill, New York, 1980. DOI: https://doi.org/10.1002/0470011815.b2a4a002
10. Kumru M., Kumru P.Y. Analytic hierarchy process application in selecting the mode of transport for a logistics company. *Journal of Advanced Transportation*. 2014. V.48. No.8. P.974-999. DOI: https://doi.org/10.1002/atr.1240

11. Vieira J.G.V. An AHP-based framework for logistics operations in distribution centres. *International Journal of Production Economics*. 2017. V.187. P. 246-259. DOI: https://doi.org/10.1016/j.ijpe.2017.03.001
12. Jharkharia S., Shankar R. Selection of logistics service provider: An analytic network process (ANP) approach. *Omega*. 2007. V.35, No.3, P. 274-289. DOI: https://doi.org/10.1016/j.omega.2005.06.005
13. Tadić S., Zečević S., Krstić M. A novel hybrid MCDM model based on fuzzy DEMATEL, fuzzy ANP and fuzzy VIKOR for city logistics concept selection. *Expert systems with applications*. 2014 V.41. No.18. P.8112-8128. DOI:10.1016/j.eswa.2014.07.021
14. Mardani A. VIKOR technique: A systematic review of the state of the art literature on methodologies and applications. *Sustainability*. 2016. V. 8 No.1, P. 37. DOI: https://doi.org/10.3390/su8010037
15. Rahman M.A., Pereda V.A. Freight transport and logistics evaluation using entropy technique integrated to TOPSIS algorithm. In: Design Solutions for User-Centric Information Systems. IGI Global, 2017. P.63-89. DOI: https://doi.org/10.4018/978-1-5225-1944-7.ch004
16. Chen T.Y., Lin Y.J. A fuzzy TOPSIS approach for supplier evaluation and selection in supply chain management. *International Journal of Production Economics*. 2011. V.136. No 1. P.201-208. DOI: https://doi.org/10.1016/j.ijpe.2011.08.016
17. Kozina Y., Volkova N., Horpenko D. Mobile Application for Decision Support in Multi-Criteria Problems. *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)*, Lviv, Ukraine, 2018, P. 56-59. DOI: https://doi.org/10.1109/DSMP.2018.8478499.
18. Kozina Y., Volkova N., Horpenko D. Mobile Decision Support System To Take Into Account Qualitative Estimation By The Criteria. *2020 IEEE Third International Conference on Data Stream Mining & Processing (DSMP)*. Lviv, Ukraine, 2020, P. 357-361. DOI: https://doi.org/10.1109/DSMP47368.2020.9204134.
19. Si C., Dong C., Mi G. The regional logistics hubs location problem based on the technique for order preference by similarity to an ideal solution and genetic algorithm: A case of Sichuan. *Journal of Computational and Theoretical Nanoscience*. 2016. V.13. No 9. P. 6065-6075. DOI: https://doi.org/10.1166/jctn.2016.5529.
20. Kannan G. Selection of a sustainable third-party reverse logistics provider based on the robustness analysis of an outranking graph kernel conducted with ELECTRE I and SMAA. *Omega*. 2019. V.85. P.1-15. DOI: https://doi.org/10.1016/j.omega.2018.05.007
21. Google Maps. 2023. URL: https://www.google.com/maps

# COMPARATIVE ANALYSIS OF METHODS FOR SOLVING OPERATIONAL PROBLEMS OF TRANSPORT LOGISTICS

D.R. Horpenko, B.O. Boltenkov

National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
e-mails: horpenko@op.edu.ua, boltenkov@op.edu.ua

The article is devoted to the issue of determining the best logistics route when solving operational problems of transport logistics under martial law by a team of volunteers. The article reviews multi-criteria decision-making methods used to solve operational problems of transport logistics. The choice of the best route for transporting passengers or cargo is affected by changes in the state of the external environment, which dynamically changes during military operations. All members of the volunteer team, who have up-to-date information on the conditions of the routes, are involved in the rapid assessment of the state of the external environment. This information is transferred to the coordinator of the volunteer team, who makes a decision on the choice of the logistics route. In the conditions of martial law, both the number of logistics routes and the state of the routes can change dynamically. Thus, the methods used by the volunteer coordinator to support decision-making should be robust to changes in the number of possible logistical routes and their changes. And also have a small computational complexity and support group decision making. The article compared AHP (Analytic Hierarchy) and mSmart methods based on such factors as: adequacy to changes in alternatives or criteria; flexibility in the decision-making process; computational complexity; adequacy of group decision-making support; number of alternative routes and criteria. The task of choosing the best possible route from Odesa to Kherson was considered. The results showed that both methods are suitable for solving the problem of choosing the best route, but the mSmart method turned out to be better in terms of the factors considered in the work.

**Keywords**: multi-criteria decision-making methods; transport logistics; a team of volunteers; AHP; mSmart

DOI 10.15276/imms.v13.no1-2.48
УДК 004.056.5

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp.48-55

# РОЗРОБКА ПЛАТФОРМИ ДЕЦЕНТРАЛІЗОВАНОГО РЕЄСТРУ З ПОКРАЩЕНИМИ ХАРАКТЕРИСТИКАМИ

С.С. Грибняк

Національний університет «Одеська Політехніка»
просп. Шевченка, 1, Одеса, 65044, Україна
e-mail: ssgrybniak@op.edu.ua

Технології розподілених реєстрів за п'ятнадцять років свого існування знайшли широке застосування у сфері фінансового обороту, криптовалютних додатках, системах захищеного документообігу. Найбільш популярними на сьогоднішній день є блокчейн системи Bitcoin та Etherium. Незважаючи на їх переваги (незмінність відпрацьованих транзакцій, децентралізованість, прозорість), вони мають серйозні недоліки – невисоку швидкість обробки транзакцій і обмежену масштабованість. У цьому плані їм складно конкурувати з централізованими фінансовими платформами, що мають швидкість обробки транзакцій на порядки вище. Мета цієї роботи – розробка платформи, на основі технології децентралізованого реєстру з покращеними характеристиками масштабованості та швидкості обробки транзакцій. В основу побудови системи покладено архітектуру, засновану на спрямованому ациклічному графі – BlockDAG, яка вигідно відрізняється від Blockchain асинхронністю функціонування. Упорядкування блоків у ній здійснюється шляхом топологічного лінійного сортування спрямованого графа. Крім того, швидкість обробки у BlockDAG зростає зі збільшенням числа користувачів. При побудові платформи запропоновано використовувати двошарову схему, яка складається з двох мереж – основної BlockDAG мережі та координаційної мережі, побудованої на основі Blockchain. В основній мережі відбувається створення блоків та розповсюдження їх по мережі. Координаційна мережа виконує функції атестації блоків та його фіналізації. Застосовано протокол консенсусу Proof of Stake. Проведено реалізацію запропонованої схеми у вигляді експериментальної платформи Waterfall, яка призначена для обслуговування транзакцій з різними токенами, включаючи NFT, обслуговування смарт-контрактів і розробки розподілених додатків. Тестування показало високу швидкість обробки транзакції у поєднанні з необхідною масштабованістю.

**Ключові слова:** технології розподілених реєстрів, обробка транзакцій, двошарова мережа, BlockDAG , платформа Waterfall

**Вступ.** Технології розподілених реєстрів (Distrsbuted Ledger Technology, DLT) [1] є вибуховою і найбільш стрімко розвиваючоюся гілкою інформаційних технологій 21-го століття. Першим поколінням практичного застосування DLT став класичний блокчейн, описаний у 2008 році, який є одноранговою піринговою децентралізованою системою обробки транзакцій. Класичний блокчейн став основою децентралізованого фінансового обороту і першої криптовалюти – біткойн. Наступним поколінням DLT стала Etherium – платформа для створення децентралізованих онлайн-сервісів на базі блокчейна, які працюють на базі смарт контрактів, та відповідна криптовалюта [2]. За 15 років інтенсивного розвитку систем, заснованих на DLT, та їх практичного застосування, виявилися як їхні переваги, так і властиві їм недоліки. На усунення цих недоліків спрямовано розробку та створення численних DLT платформ. У цьому аспекті тема даної роботи, присвяченої розробці та тестовим випробуванням швидкодіючої та високомасштабованої децентралізованої системи обробки транзакцій є досить актуальною.

**Аналіз існуючих технологій побудови розподілених реєстрів.** DLT, що засновані як на класичному блокчейні, так і на платформах Etherium, широко використовуються в різних практичних додатках [3]: фінансових послугах, включаючи платіжні системи [4,5], медицині та охороні здоров'я [6,7] , секторі нерухомості [8], підтримці Інтернету речей ( IoT ) [9,10], управлінні логістикою [11], енергетиці [12,13], послугах, що засвідчують особу (ID) [14] та ін. В процесі широкого впровадження DLT гостро проявилася передбачена раніше Бутеріним трилема блокчейна [15]. Трилема блокчейна є концепцією, яка описує три основні аспекти блокчейн-технології: децентралізацію, масштабованість та безпеку. Ця концепція стверджує, що неможливо одночасно досягти повної децентралізації, високої масштабованості та безпеки в блокчейні . Компоненти трилеми розуміються таким чином.

1. Децентралізація. Будь-яка DLT спрямована на децентралізацію, тобто рівномірний розподіл контролю та участі між вузлами мережі. Це гарантує стійкість мережі та підвищує довіру учасників. Однак повна децентралізація може призвести до низької пропускної здатності мережі та тривалого підтвердження транзакцій.
2. Масштабованість. Масштабованість відноситься до здатності децентралізованого реєстру обробляти велику кількість транзакцій і підтримувати мережу, що росте. Висока масштабованість дозволяє швидко та ефективно обробляти транзакції. Однак реалізація високої масштабованості може призвести до зменшення децентралізації та вразливості безпеки. Розподілена система повинна мати механізм масштабування для адаптації до зміни робочого навантаження у дуже широких межах. Однак, наприклад, швидкість обробки даних у відомих популярних системах Bitcoin та Etherium невисока – ці системи обробляють приблизно 7 та 20 транзакцій за секунду (tps) відповідно. Ці показники незрівнянні з традиційними централізованими системами, що обробляють тисячі транзакцій за секунду [16].
3. Безпека. Блокчейн забезпечує безпеку шляхом використання відповідних елементів криптографії та згоди більшості учасників мережі. Високий рівень безпеки в блокчейні вимагає високої обчислювальної потужності та достатньої кількості вузлів у мережі для підтвердження та підтримки надійності транзакцій. Однак підвищення безпеки може призвести до збільшення часу та ресурсів, необхідних для обробки транзакцій. Трилема блокчейна вказує на необхідність балансування цих трьох аспектів та вибору пріоритетів у розробці та реалізації блокчейн-рішень. Компроміс між децентралізацією, масштабованістю та безпекою є ключовим фактором при проектуванні та впровадженні блокчейн-систем. Таким чином, існує гостра необхідність у розробці та створенні високомасштабованої розподіленої системи з високою швидкістю обробки транзакцій.

**Мета роботи.** Мета роботи – розробка платформи децентралізованого реєстру з покращеними характеристиками масштабованості та швидкості обробки транзакцій.

**Основна частина.** Наступним поколінням технології побудови розподілених реєстрів слід вважати подання послідовності транзакцій як спрямованого ациклічного графа
(directed acyclic graph , DAG) [18]. Технологія на основі DAG побудована на поданні всієї множини транзакцій у вигляді направленого графа. Вершинами графа є транзакції (архітектура TxDAG ) або блоки транзакцій (архітектура BlokDAG ). Ребра графа з'єднують кожну вершину з усіма раніше утвореними (батьківськими) блоками, які ще не мають посилань. Таким чином, вся множина транзакцій представляється у вигляді спрямованого дерева. Далі вирішується відома в теорії графів задача лінійного топологічного впорядкування графа [19]. У результаті

дерево транзакцій перетворюється на лінійно впорядковану послідовність блоків з однаковим напрямом ребер – від останніх за часом утворення до раніше утворених. Це створює можливість застосування правила консенсусу для підтвердження і верифікації блоків. Архітектура BlokDAG була обрана як основна для побудови системи розподіленого реєстру з підвищеною швидкістю обробки транзакцій при високій масштабованості. Ці переваги BlokDAG у порівнянні з традиційним блокчейном досягаються за рахунок асинхронності утворення нових блоків у BlokDAG. Блокчейн побудовано на очікувані утворення нового блоку після верифікації попереднього елементу ланцюжка. У BlokDAG блоки утворюються незалежно від верифікації попередніх, а власне ланцюжок формується внаслідок топологічного впорядкування. Крім того, BlokDAG архітектура має парадоксальну, на перший погляд, але доведену властивість [18] – зі збільшенням числа користувачів системи швидкість обробки блоків / транзакцій зменшується.

Для подальшого прискорення обробки транзакцій запропоновано двошарову модель побудови системи. Основна мережа, призначена для формування та розповсюдження блоків, виконана за технологією BlokDAG (шар 1). Операції, пов'язані з верифікацією блоків на підставі протоколу консенсусу, винесені до координаційної мережі, що працює за технологією традиційного консенсусу (шар 2).

Як і в системі Etherium [2] для обробки транзакцій застосована дискретна часова шкала. Мінімальним інтервалом часу є слот тривалістю 4 с протягом якого дії шарів синхронізуються. Користувачі повинні створити та розподілити по мережі свій блок під час слота. Слоти поєднуються в епохи. Епохи призначені для підбиття проміжних результатів мережі. У запропонованій моделі епоха складається із 32 слотів.

Основним технологічним структурним елементом мережі є вузол (node, нода). Вузлом є зареєстрований сервер в мережі, що зберігає всі відповідні записи у вигляді реєстру. На кожному вузлі може бути розгорнуто певну кількість логічних структурних елементів, які умовно називатимемо Workers (Працівники), їх облікові записи мають необхідні дані для участі у протоколі консенсусу PoS [20]. Кожен Worker після включення до системи складається з двох компонентів з незалежними адресами – Утворювач блоків (Creator) та Координатор (Coordinator) (рис.1).

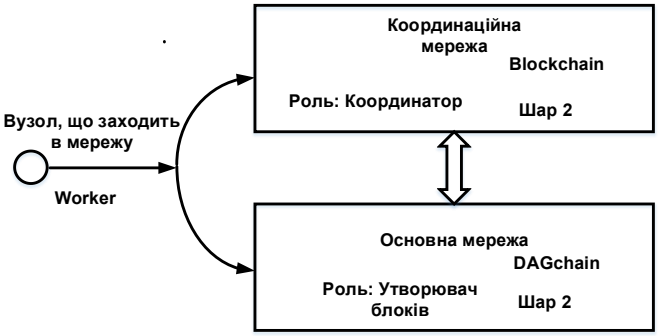



**Рис. 1.** Двошарова модель системи розподіленого реєстру

Функції обробки транзакцій розподілені між шарами системи в певний спосіб.

В основній мережі виконуються такі основні операції.

1. Пошук інших вузлів та підключення до них.
2. Прийом транзакцій від користувачів, розміщення в пул транзакцій та пересилання транзакцій далі по мережі.

3. Визначення на підставі методу перемішування, хто із творців створює блок у кожному слоті.
4. Вилучення транзакцій з пулу, додавання в блок, відправлення даного блоку іншим учасникам мережі.
5. Передача своєму координуючому вузлу порядку блоків, що входять в цей вузол.
6. Отримання від свого координуючого вузла порядку блоків для фіналізації. Під фіналізацією (або остаточністю) розуміється процес, після завершення якого транзакція в мережі може вважатися остаточною і не існує ризику фальсифікації (зміни) транзакції або блоку в упорядкованому ланцюжку.
7. Збереження історії блоків та транзакцій у вузлах реєстру.
8.Участь у синхронізації блоків та транзакцій.

Основні операції, які виконуються в координаційній мережі такі.

1. Визначення в кожному слоті епохи складу комітетів Proof of Stake та ролей координаторів у них (творець, атестатор, агрегатор).
2. Прийом результатів атестації від інших вузлів та передача її далі через мережу.
3. Додавання утворювачем в блок атестацій, раніше не доданих в блок.
4. Об'єднання сформованих агрегаторами атестацій в один мультипідпис .
5. Відповідно до отриманих атестацій та алгоритму консенсусу формування ланцюжка блоків, що підлягають фіналізації ; відправка фінального ланцюжка блоків у свій вузол основної мережі для фіналізації.
6. Синхронізація результату консенсусу з основною мережею.
7. Зберігання історії блоків та атестацій.
8. Зберігання стану координаторів (баланси, статус) у загальному стані мережі.

На підставі викладених архітектурних та алгоритмічних рішень розроблено платформу розподіленого реєстру Waterfall [21].

Основні властивості системи.

Мова програмування – Golang .

Архітектура – двошарова, основна мережа – BlockDAG , координаційна мережа – Blockchain .

Протокол консенсусу – удосконалений Proof of Stake.

Функціональність, що забезпечується, – обслуговування транзакцій з відомими і вбудованими токенами (включаючи NFT), обслуговування смарт-контрактів, розробка розподілених додатків dApps .

На рис. 2 наведено одну з екранних форм платформи Waterfall.


**Рис.2.** Екран виведення результуючого лінійного топологічного впорядкування BlockDAG.

Проведено навантажувальні експерименти для розробленої системи. Тестова мережа була сформована на базі серверів Amazon Elastic Compute Cloud. Тестова мережа працювала на 64 екземплярах t3.small (два ядра ЦП та 2 ГБ пам'яті) Amazon EC2. У ході експериментів було згенеровано пул приблизно зі 100 000 транзакцій та зафіксовано час, за який остання з них буде записана до реєстру. Були зроблені виміри масштабованості, Середня швидкість становила 2234 tps. Модифікована версія Waterfall обробила понад 3600 транзакцій за секунду при завантаженні ЦП менше 20%. На рис.3 наведено один із екранів з результатами тестування.

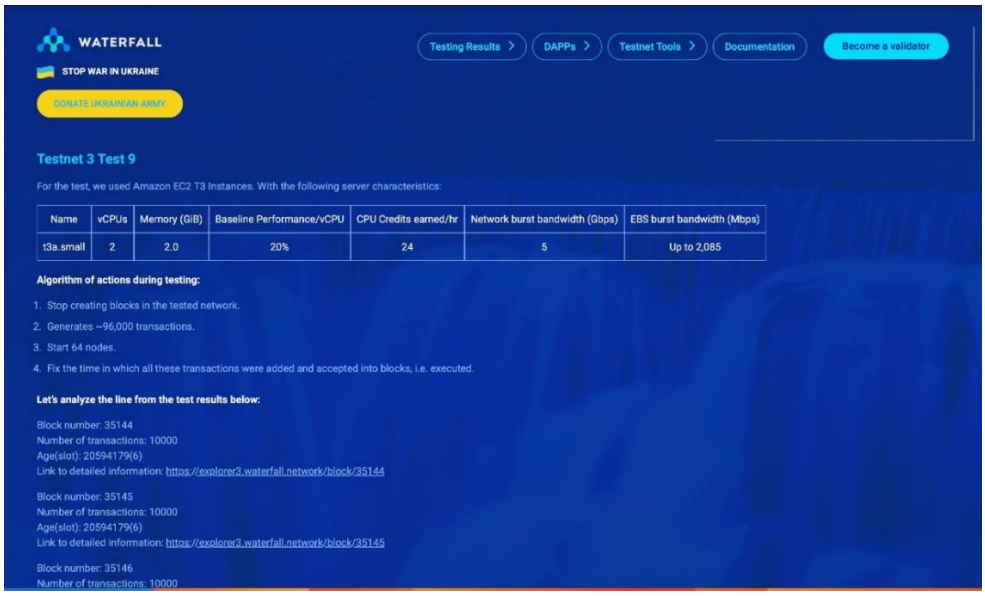



**Рис.3.** Приклад екрану з результатами тестування навантаження.

Поточна реалізація Waterfall використовує технологію Etherium EVM, що полегшує використання смарт-контрактів. Смарт-контракти для Ethereum можна копіювати і вставляти, при цьому вони успішно працюють у тестовій мережі Waterfall. Експерименти показали, що для імпорту більшої кількості децентралізованих додатків може знадобитися незначна зміна їхнього коду.

Безпека реалізації системи підтримується тими самими криптографічними протоколами, які добре зарекомендували себе у мережі Ethereum.

Хеш-функції – основна мережа – Keccak-256/SHA3, координаційна мережа – SHA256.

Цифрові підписи для формування ключів – вузли основної мережі – ECDSA ( Elliptic Curve Digital Signature Algorithm ) secp256k1, вузли координаційної мережі - підпис BLS (Boneh-Lynn-Shacham). Встановлено рекомендовану конфігурацію для успішного запуску вузла:

– Число ЦП – 2;
– Оперативна пам'ять – 4 Гб;
– SSD – 80 Гб;
– Інтернет трафік – 400 Гб/міс.

**Результат та обговорення.** Розроблена платформа Waterfall успадковує та покращує Ethereum 2.0. Крім того, платформа має ряд переваг:

1. Висока продуктивність – масштабовані в системі блокові структури на основі DAG дозволяють одночасно публікувати кілька блоків. Це формує DAG та забезпечує завершеність всіх транзакцій за умови, що блоки не конфліктують один з одним. Тому Waterfall може одночасно обробляти карти Visa, MasterCard та

Union.Pay на децентралізованому рівні навіть у години пік. Згідно з останніми проміжними лабораторними тестами, система могла обробляти 3600 транзакцій в секунду. Для порівняння зазначимо, що Visa обробляє близько 1700 транзакцій на секунду.

2. Низькі комісії за транзакції – архітектура спроектована таким чином, щоб підтримувати мінімальні комісії у різних сценаріях. Протокол динамічно масштабується зі зростанням навантаження на мережу. У той час, як продуктивність всієї системи збільшується, в тому самому слоті одночасно публікується більше блоків, а транзакційні збори знижуються в міру масштабування системи. Це знижує кількість операцій у пулі транзакцій навіть у години пік.

3. Низький фінансовий поріг входу – вузол із 6 Worker'ів коштує приблизно 1 920 доларів США.

4. Платформа обслуговує вбудовані токени – випуск та обслуговування токенів (включно з NFT) не потребують спеціальних смарт-контрактів, а виконуються за допомогою звичайних транзакцій, що значно знижує накладні витрати. Крім того, це робить їх використання більш доступним для широкого кола користувачів.

5. Динамічна настройка – платформа має механізми динамічної адаптації параметрів системи залежно від ситуації, що змінюється, зокрема, час слота, оптимальна кількість Worker'ів та деякі інші параметри налаштовуються автоматично.

Таким чином, платформа Waterfall забезпечує сприятливе середовище для надання та споживання широкого спектру послуг для ведення бізнесу та соціальної діяльності у зручному форматі загальнодоступного децентралізованого реєстру.

**Висновки.** Розроблено експериментальну платформу Waterfall, на основі технології децентралізованого реєстру з покращеними характеристиками масштабованості та швидкості обробки транзакцій. В основі платформи лежить запропонована автором двошарова модель розподіленого реєстру, що поєднує в собі основну мережу, побудовану за технологією DAGChain , та координаційну мережу, засновану на традиційній блокчейн -технології.

Платформа призначена для обслуговування транзакцій з різними токенами, включаючи NFT, обслуговування смарт-контрактів і розробки розподілених додатків. Тестування системи показало, що вона може обробляти в середньому 2234 транзакції за секунду при досить високому рівні масштабованості.

**Список літератури**


1. Maull R., Godsiff P., Mulligan C., Brown A., Kewell B. Distributed ledger technology: Applications and implications. *Strategic Change* . 2017. Vol. 26. No.5. P. 481-489. URL: https://doi.org/10.1002/jsc.2148.
2. Ethereum 2.0 Specifications. URL: https://bounties.gitcoin.co/grants/551/the-ethereum-20-annotated-specification
3. Jaoude J., Saade R. Blockchain Applications – Usage in Different Domains. *IEEE Access*. 2019. P. 45360-45381. https://doi.org/ 10.1109/ACCESS.2019.2902501.
4. Mohd J., Abid H., Ravi P.S, Rajiv S., Shahbaz K. Review of Blockchain Technology applications for financial services. *BenchCouncil Transactions on Benchmarks, Standards and Evaluations*. 2022. Vol.2. No.3. P.100073. https://doi.org/10.1016/j.tbench.2022.100073.
5. Mihus I. Evolution of practical use of blockchain technologies by companies. *Economics, Finance and Management Review*. 2022. No.1. P. 42–50. URL: https://doi.org/10.36690/2674-5208-2022-1-42 .
6. Abid H., Mohd J., Ravi P.S, Rajiv S., Shanay R. Blockchain technology applications in healthcare: An overview. *International Journal of Intelligent Networks*. 2021. V.2. P.130-139. URL: https://doi.org/10.1016/j.ijin.2021.09.005.

7. Ghosh P.K, Chakraborty A., Hasan M., Rashid K., Siddique AH Blockchain Application в Healthcare Systems: A Review. *Systems* . 2023. V.11. P. 38. URL: https://doi.org/10.3390/systems11010038 .
8. Garcia-Teruel R. Legal challenges and opportunities of blockchain technology in the real estate sector. *Journal of Property, Planning and Environmental Law*. 2020. URL: https://doi.org/10.1108/JPPEL-07-2019-0039 .
9. Abbassi Y., Benlahmer H. IoT and Blockchain combined: for decentralized security. *Procedia Computer Science* . 2021. V. 191. P. 337-342. URL: https://doi.org/10.1016/j.procs.2021.07.045.
10. Patel C. IoT private preservation using blockchain IoT privacy preservation using blockchain. *Information Security Journal: Global Perspective* . 2021. No.31. URL: https://doi.org/10.1080/19393555.2021.1919795.
11. Perboli G., Musso S., Rosano M. Blockchain в Logistics and Supply Chain: A Lean Approach for Designing Real-World Use Cases. *IEEE Access*. 2018. P.1-11. URL: https://doi.org/10.1109/ACCESS.2018.2875782.
12. Borkovcová A., Černá M., Sokolová M. Blockchain in Energy Sector – Systematic Review. *Sustainability*. 2021. V.22. No.14. P.14793. URL: https://doi.org/10.3390/su142214793.
13. Amanda A., Mika G., Masaru Y., Kenji T., Daishi S. Challenges and opportunities of blockchain energy applications: Challenges and opportunities of blockchain energy applications: Interrelatedness among technological, economic, social, environmental, and institutional dimensions. *Renewable i Sustainable Energy Reviews*. 2022. V. 166, P.112623. URL: https://doi.org/10.1016/j.rser.2022.112623
14. Sung C.S., Park J.Y. Understanding of blockchain-based identity management system adoption in the public sector. *Journal of Enterprise Information Management*. 2021.Vol. 34. No. 5. P. 1481-1505. URL: https:// doi.org/10.1108/JEIM-12-2020-0532
15. Buterin V. Proof Stake: Making of Ethereum i Philosophy of Blockchains. N.Y.: Seven Stories Press, 2022. 322 p.
16. Hafid A., Hafid A. S., Samih M. Scaling Blockchains: A Comprehensive Survey. *IEEE Access*. 2020. V.8. P.125244-125262. URL: https://doi: 10.1109/ACCESS.2020.3007251.
17. Qin W., Jiangshan Y., Shiping C., Yang X. SoK: Diving into DAG-based Blockchain Systems. 2022. 38p. URL: https://arxiv.org/pdf/2012.06128.pdf
18. Zverovich V. Modern Applications of Graph Theory. Oxford: Oxford University Press, 2021. 416p.
19. Chen T. Understanding Ethereum via Graph Analysis. *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, Honolulu, HI, USA. 2018. P. 1484–1492. URL: doi: 10.1109/INFOCOM.2018.8486401.
20. Waterfall - DAG-based scalable smart-contract platform. URL: https://www.waterfall.network/

# DEVELOPMENT OF A DECENTRALIZED LEDGER PLATFORM WITH IMPROVED CHARACTERISTICS

S.S. Grybniak

National Odesa Polytechnic University,
1, Shevchenko Ave. Odesa, 65044, Ukraine; e-mail: ssgrybniak@op.edu.ua

Over the fifteen years of its existence, distributed ledger technologies have found wide application in the field of financial turnover, cryptocurrency applications, and secure document management systems. The most popular today are the blockchain systems of Bitcoin and Etherium. Despite of their advantages – the immutability of processed transactions, decentralization, transparency – they have a serious drawback: low transaction processing speed and limited scalability. In this regard, it is difficult for them to compete with decentralized financial platforms that have transaction processing speeds orders of magnitude higher. The purpose of this work is to develop a decentralized ledger platform with improved scalability and transaction processing speed. The system was built on an architecture based on a directed acyclic graph – BlockDAG. BlockDAG architecture compares favorably with Blockchain in asynchronous operation. The ordering of blocks in it is carried out by topological linear sorting of the directed graph. In addition, the processing speed in BlockDAG increases with the increase in the number of users. The platform is built on a two-layer scheme and consists of two networks – the main BlockDAG network and the coordination network built on the basis of Blockchain. In the main network, blocks are created and distributed over the network. The coordination network performs the functions of block certification and their finalization. Proof of Stake consensus protocol applied. The system is implemented as an experimental Waterfall platform. The platform is designed to serve transactions with various tokens, including NFT, to serve smart contracts and develop distributed applications. System testing showed a high transaction processing speed combined with the required scalability.

**Keywords:** distributed ledger technologies , two-layer network , BlockDAG , Waterfall platform

DOI 10.15276/imms.v13.no1-2.56
УДК 004.056.5

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp.56-62

# ПІДВИЩЕННЯ СТІЙКОСТІ СТЕГАНОГРАФІЧНОГО МЕТОДУ З КОДОВИМ УПРАВЛІННЯМ ВБУДОВУВАННЯМ ДОДАТКОВОЇ ІНФОРМАЦІЇ ДО АТАК ЗАШУМЛЕННЯМ

Д.О. Гулід, А.В. Соколов

Національний університет «Одеська політехніка»
Україна, Одеса, 65044, пр-т Шевченка, 1, radiosquid@gmail.com

Зростання обсягів мультимедійного контенту, що генерується, зберігається та передається сучасними інформаційними системами, призводить до збільшення ролі стеганографічної компоненти в системах захисту інформації. При цьому, до застосовуваних стеганографічних методів висуваються значні вимоги щодо їх ефективності, які включають достатню пропускну спроможність, забезпечення надійності сприйняття, стійкості до атак проти вбудованого повідомлення. Велике значення має швидкодія стеганографічного методу, особливо, якщо передбачається його застосування у режимі реального часу на ресурсообмежених платформах. Одним із сучасних стеганографічних методів, що характеризується забезпеченням основних показників ефективності при незначній обчислювальній складності через виконання стеганоперетворення у просторовій області контейнера, є стеганографічний метод з кодовим управлінням вбудовуванням додаткової інформації. Однак, незважаючи на досить високі показники стійкості до атаки зашумленням, яка може мати місце в багатьох практичних застосуваннях, актуальним лишається питання підвищення стійкості стеганоперетворення до даного типу атак. Метою роботи є підвищення стійкості до атак зашумленням стеганографічного методу з кодовим управлінням вбудовуванням додаткової інформації. У роботі розглянуто атаки проти вбудованого повідомлення зашумленням двома видами шумів: адитивним білим гаусовим шумом та шумом типу «Salt and Pepper». У роботі проведено експериментальне дослідження впливу структури застосовуваного кодового слова на стійкість стеганографічного методу з кодовим управлінням до атаки зашумленням адитивним білим гаусовим шумом, яке дозволило виробити практичні рекомендації щодо параметрів стеганографічного методу з кодовим управлінням в умовах даного типу атаки. Запропоновано застосування методу ШОВ (широкосмуговий сигнал, обмежувач, вузькосмуговий сигнал) для підвищення стійкості стеганографічного методу з кодовим управлінням до атак зашумленням шумом «Salt and Pepper», що дозволило знизити кількість помилок при вилученні додаткової інформації в умовах зашумлення даним шумом на 48%. Отримані у даній статті результати можуть бути корисними для застосування стеганографічного методу з кодовим управлінням на практиці в умовах передавання стеганоповідомлення по каналам, що характеризуються наявністю атак зашумленням проти вбудованого повідомлення.

**Ключові слова:** стеганографія, кодове управління вбудовуванням інформації, атака зашумленням, адитивний білий гаусів шум, шум «Salt and Pepper».

**Вступ і постановка задачі.** Важливим компонентом сучасних систем захисту інформації є стеганографічні методи, що забезпечують приховування самого факту наявності інформації, що захищається. На сьогодні, до застосовуваних стеганографічних методів висуваються суворі вимоги ефективності, що включають високу пропускну спроможність, забезпечення надійності сприйняття, стійкості до атак проти вбудованого повідомлення, а також низьку обчислювальну складність стеганоперетворення.

При цьому, вимога забезпечення стійкості стеганографічного методу до атак проти вбудованого повідомлення стає у протиріччя з вимогою забезпечення

низької обчислювальної складності через необхідність застосування просторів перетворень, перехід до яких потребує значних обчислювальних витрат (перш за все, сингулярного розкладання матриць блоків контейнера [1…5]), тоді як у більшості випадків методи, що застосовують для стеганоперетворення просторові області контейнера [6…8], є нестійкими до атак проти вбудованого повідомлення.

Вирішенням даного протиріччя стала розробка стеганографічного методу з кодовим управлінням вбудовуванням додаткової інформації (ДІ) [9], що працює у просторовій області контейнера та при цьому є здатним забезпечити відповідність зазначеним критеріям ефективності навіть у більшій мірі, порівняно з відомими методами, що передбачають застосування областей перетворення. На сьогоднішній день відомо чимало атак проти вбудованого повідомлення, найбільш розповсюдженими серед яких є атаки стисненням, тим не менш, доволі часто на практиці зустрічаються атаки зашумленням різними видами шумів, що робить актуальною задачу підвищення стійкості стеганографічних методів до даного виду атак. Як показують проведені дослідження, стійкість стеганографічного методу з кодовим управлінням вбудовуванням ДІ до атак зашумленням окремими видами шумів може бути суттєво підвищена.

Метою роботи є підвищення стійкості до атак зашумленням стеганографічного методу з кодовим управлінням вбудовуванням ДІ.

**Стійкість стеганографічного методу з кодовим управлінням вбудовуванням ДІ до атак зашумленням адитивним білим гаусовим шумом.** З метою порівняння стійкості стеганографічного методу з кодовим управлінням вбудовуванням ДІ до атаки зашумленням адитивним білим гаусовим шумом (АБГШ) при застосуванні різних кодових слів був проведений наступний експеримент. У вибірку з 500 кольорових зображень з бази NRCS [10] виконувалося вбудовування ДІ із застосуванням кодових слів, що впливають на обрану трансформанту перетворення Уолша-Адамара, після чого отримане стеганоповідомлення піддавалося зашумленню АБГШ із різними рівнями дисперсії $D$ і математичним очікуванням $M = 0$. Після атаки зашумленням здійснювалося вилучення ДІ і вимірювання кількості помилок, що сталися. Результати дослідження представлені у табл. 1, де інтенсивність зашумлення представлено у вигляді PSNR [11]. У табл. 1 для стислості показані лише декілька кодових слів кожного розміру, що вибірково впливають на задану трансформанту перетворення Уолша-Адамара.

Аналіз даних, що представлені у табл. 1 показує, що стійкість методу до атак зашумленням АБГШ не залежить від обраного кодового слова. Оскільки шум АБГШ має рівномірну спектральну щільність, це призводить до того, що вибір конкретного кодового слова не має значного впливу на стійкість методу.

Розкид значень відсотку помилок при атаці проти вбудованого повідомлення зашумленням АБГШ з різними значеннями дисперсії для різних кодових слів порядку $\mu$ з елементарною структурою $\{\mu(1), 0(\mu-1)\}$ становить: до 0.2% помилок для кодових слів розміру $\mu = 4$, до 0.1% помилок для кодових слів розміру $\mu = 8$ та до 0.3% помилок для кодових слів розміру $\mu = 16$.

При застосуванні кодового слова, яке впливає на постійну складову, відсоток помилок при декодуванні в умовах атаки зашумленням в середньому більший на 0.7% для кодових слів розміру $\mu = 4$, на 0.9% для розміру $\mu = 8$ і на 1.9% для розміру $\mu = 16$.

**Таблиця 1**

Залежність кількості помилок при вилученні ДІ від PSNR

| Кодове слово / PSNR | 6.9499 | 23.2094 | 30.0960 | 33.0817 | 35.2827 | 40.0017 | 42.9501 | 49.5088 | 61.1746 |
|---|---|---|---|---|---|---|---|---|---|
| $N=4$ | | | | | | | | | |
| $T_{4,(1,1)}$ | 49.7347 | 41.9107 | 32.1191 | 25.5330 | 19.8208 | 7.4452 | 2.5829 | 0.7541 | 0.6206 |
| $T_{4,(1,2)}$ | 49.5007 | 41.6275 | 31.7966 | 25.1617 | 19.4106 | 6.9208 | 1.9746 | 0.0327 | 0 |
| $T_{4,(1,3)}$ | 49.5107 | 41.6357 | 31.7908 | 25.1949 | 19.4223 | 6.9358 | 1.9887 | 0.0342 | 0 |
| … | | | | | | | | | |
| $T_{4,(4,4)}$ | 49.5010 | 41.6453 | 31.7912 | 25.1672 | 19.4139 | 6.9157 | 1.9818 | 0.0324 | 0 |
| $N=8$ | | | | | | | | | |
| $T_{8,(1,1)}$ | 49.7010 | 34.1495 | 17.5188 | 9.4970 | 4.8697 | 1.0886 | 0.8595 | 0.6934 | 0.5338 |
| $T_{8,(1,2)}$ | 48.9184 | 33.2513 | 16.6291 | 8.5859 | 3.9630 | 0.2255 | 0.0365 | 0 | 0 |
| $T_{8,(1,3)}$ | 48.9328 | 33.2808 | 16.6394 | 8.5975 | 3.9916 | 0.2304 | 0.0380 | 0 | 0 |
| … | | | | | | | | | |
| $T_{8,(8,8)}$ | 48.9279 | 33.2686 | 16.6103 | 8.5822 | 3.9681 | 0.2224 | 0.0353 | 0 | 0 |
| $N=16$ | | | | | | | | | |
| $T_{16,(1,1)}$ | 49.6422 | 21.6210 | 4.4085 | 1.7983 | 1.2713 | 0.9367 | 0.8081 | 0.6399 | 0.4615 |
| $T_{16,(1,2)}$ | 47.7919 | 19.2586 | 2.6979 | 0.4183 | 0.0871 | 0.0035 | 0 | 0 | 0 |
| $T_{16,(1,3)}$ | 47.8877 | 19.2815 | 2.7092 | 0.4186 | 0.0913 | 0.0035 | 0 | 0 | 0 |
| … | | | | | | | | | |
| $T_{16,(16,16)}$ | 47.8049 | 19.2161 | 2.6898 | 0.4125 | 0.0898 | 0.0030 | 0 | 0 | 0 |

Зростання стійкості стеганографічного методу з кодовим управлінням вбудовуванням ДІ можливе лише за рахунок збільшення енергії кодового слова, тобто із застосуванням багаторівневих кодових слів [12].

На підставі результатів проведеного експерименту можна сформулювати наступні рекомендації щодо застосування стеганографічного методу з кодовим управлінням вбудовуванням ДІ в умовах атак зашумленням АБГШ:

1. Застосування багаторівневих кодових слів — за аналогією із вибором сигнальних конструкцій для роботи у каналах зв'язку, що зашумлені АБГШ, більша стійкість стеганографічного методу до атак зашумленням проти вбудованого повідомлення може бути забезпечена через збільшення енергії кодового слова шляхом застосування багаторівневих кодових слів.

2. Забезпечення впливу на високочастотні складові — задля забезпечення найбільшої надійності сприйняття стеганоповідомлення, зважаючи на рівний степінь стійкості кодових слів до атак зашумленням АБГШ, може бути рекомендоване застосування кодових слів, що впливають на найбільш високочастотні складові, наприклад, на трансформанту Уолша-Адамара (2,2). У разі застосування багаторівневих кодових слів, вибір даної трансформанти, для вбудовування ДІ також нівелює проблему знаходження максимальних за амплітудою елементів по краям кодового слова, що дозволить уникнути виникнення проблеми найбільшого перепаду яскравості на границях блоків та підвищити надійність сприйняття стеганоповідомлення.

Зазначені модифікації дозволяють адаптувати стеганографічний метод з кодовим управлінням вбудовування ДІ до роботи в умовах атак зашумлення АБГШ.

**Стійкість стеганографічного методу з кодовим управлінням вбудовуванням ДІ до атак зашумленням шумом «Salt and Pepper».** Для застосування стеганографічного методу з кодовим управлінням вбудовуванням ДІ в каналах з шумом типу «Salt and Pepper» можна рекомендувати додавання операції обмежування амплітуди матриць різниці $\Delta$ блоків стеганоповідомлення та оригінального контейнеру при вилученні ДІ. Це дозволить підвищити рівень стійкості стеганоповідомлення до таких атак.

Обмежування амплітуди матриць різниці полягає у встановленні її максимального значення, що відповідає максимальній амплітуді застосовуваного кодового слова. В разі використання бінарних кодових слів, максимальна амплітуда становить $\{\pm 1\}$. Цей підхід аналогічний відомому методу ШОВ, який ефективний для боротьби з імпульсними завадами.

Метод ШОВ (BAN, широка полоса — обмежувач — вузька полоса) є одним із підходів до фільтрації сигналів, зокрема використовується для боротьби з імпульсними завадами. Його назва вказує на його основні складові: широкосмуговий сигнал (Broadband), атенюатор (Attenuator) та вузькосмуговий сигнал (Narrowband) (рис. 1).


**Рис. 1.** Структурна схема ШОВ

Загальний принцип методу ШОВ може бути адаптований відповідно до специфічних вимог та властивостей шуму та сигналу у конкретній задачі, і, як показали проведені дослідження, може бути застосований для протидії шуму «Salt and Pepper» при застосуванні стеганографічного методу з кодовим управлінням вбудовуванням ДІ.

Так, відповідно до [9], при вилученні ДІ з стеганоповідомлення черговий блок $\Delta$, у загальному випадку, може містити комбінацію з багатьох частот, що відповідає широкій смузі у методі ШОВ. Після цього здійснюється обмеження блоку за амплітудою значеннями $\{\pm 1\}$, і вже після цього — виділення конкретної частотної складової, що відповідає вузькій смузі у методі ШОВ.

Розглянемо конкретний приклад. Нехай блок спотвореного шумом типу «Salt and Pepper» з дисперсією $D = 0.1$ стеганоповідомлення $S$, а також відповідний йому блок оригінального повідомлення $X$, мають вигляд

$$X = \begin{bmatrix} 157 & 150 & 151 & 149 & 146 & 142 & 141 & 139 \\ 153 & 144 & 152 & 144 & 140 & 143 & 141 & 142 \\ 149 & 141 & 148 & 145 & 137 & 140 & 136 & 145 \\ 142 & 141 & 143 & 142 & 142 & 142 & 137 & 141 \\ 143 & 143 & 142 & 143 & 143 & 146 & 147 & 142 \\ 131 & 145 & 143 & 141 & 143 & 143 & 140 & 140 \\ 133 & 145 & 144 & 142 & 147 & 146 & 130 & 129 \\ 139 & 140 & 146 & 143 & 145 & 149 & 144 & 137 \end{bmatrix}, \quad S = \begin{bmatrix} 158 & 151 & 152 & 150 & 147 & 143 & 142 & 140 \\ 154 & 255 & 153 & 145 & 141 & 255 & 255 & 143 \\ 150 & 142 & 149 & 146 & 138 & 141 & 137 & 146 \\ 143 & 142 & 144 & 143 & 143 & 143 & 138 & 142 \\ 142 & 142 & 141 & 142 & 142 & 145 & 146 & 141 \\ 130 & 144 & 255 & 140 & 142 & 142 & 139 & 139 \\ 132 & 144 & 255 & 141 & 146 & 145 & 129 & 128 \\ 138 & 139 & 255 & 142 & 144 & 148 & 143 & 255 \end{bmatrix}, \quad (1)$$

тоді як вбудовування біта ДІ $d = 1$ у стеганоповідомлення відбулося із застосуванням кодового слова $T^{+}_{16,(5,1)}$, що вибірково впливає на трансформанту перетворення Уолша-Адамара (5,1). Знайдемо матрицю $\Delta$ різниці між матрицями блоків стеганоповідомлення і оригінального повідомлення

$$\Delta = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 111 & 1 & 1 & 1 & 112 & 114 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 112 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 111 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 109 & -1 & -1 & -1 & -1 & 118 \end{bmatrix}. \quad (2)$$

Ми бачимо, що після поелеметного множення елементів отриманої матриці $\Delta$ на елементи застосованого кодового слова $T^{+}_{16,(5,1)}$, із подальшим підсумовуванням елементів результуючої матриці, і застосуванням операції $sign()$ отримуємо значення біта ДІ $d' = sign(\sum_{i=1}^{8}\sum_{j=1}^{8}\Delta(i,j)\circ T^{+}_{16,(5,1)}(i,j)) = sign(-56) = -1$, що не відповідає вбудованому біту ДІ, тобто приводить до помилки при вилученні ДІ.

Здійснимо обмеження амплітуди відповідно до методу ШОВ, в результаті чого отримуємо матрицю різниці

$$\Delta = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 1 & -1 & -1 & -1 & -1 & 1 \end{bmatrix}, \quad (3)$$

для якої після поелеметного множення елементів отриманої матриці $\Delta$ на елементи кодового слова $T^{+}_{16,(5,1)}$ із подальшим підсумовуванням елементів результуючої матриці і застосуванням операції $sign()$, отримуємо значення біта ДІ $d' = sign(\sum_{i=1}^{8}\sum_{j=1}^{8}\Delta(i,j)\circ T^{+}_{16,(5,1)}(i,j)) = sign(56) = 1$, що відповідає вбудованому біту ДІ.

Задля оцінки рівня підвищення стійкості стеганографічного методу з кодовим управлінням вбудовуванням ДІ до атаки зашумленням шумом «Salt and Pepper» був проведений обчислювальний експеримент із застосуванням 500 зображень з бази NRCS [10], що був спрямований на оцінку кількості помилок при вилученні ДІ з стеганоповідомлення в умовах атаки зашумленням, рівень якого вимірювався показником PSNR. При цьому для вбудовування ДІ застосовувалося кодове слово $T^{+}_{b,8,(5,1)}$, що впливає на трансформанту перетворення Уолша-Адамара (5,1) без застосування ШОВ, а також те саме кодове слово із застосуванням ШОВ (BAN $T^{+}_{b,8,(5,1)}$). Результати проведеного експерименту продемонстровані на рис. 2.

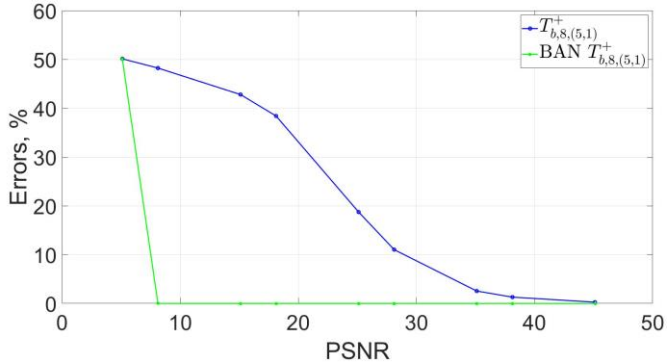



**Рис 2.** Залежність числа помилок при вилученні ДІ від інтенсивності шуму «Salt and Pepper»

Таким чином, підвищення стійкості стеганографічного методу з кодовим управлінням вбудовуванням ДІ до атак зашумленням шумом «Salt and Pepper» має місце на рівні до 48% (рис. 2) для випадку кодового слова $T^{+}_{b,8,(5,1)}$ порядку $\mu = 8$, що впливає на трансформанту Уолша-Адамара (5,1).Застосування обмеження амплітуди перед обробкою матриць різниць є важливим кроком для забезпечення якості відновлення ДІ. Воно допомагає уникнути впливу шумових компонентів, що можуть спотворити ДІ. Таким чином, обмеження амплітуди допомагає підвищити надійність та стійкість стеганографічного методу з кодовим управлінням вбудовуванням ДІ до атак зашумленням.

**Висновки.** У даній роботі було проведено дослідження, що мало на меті підвищення стійкості стеганографічного методу з кодовим управлінням вбудовуванням ДІ в умовах атак зашумленням шумом АБГШ та шумом «Salt and Pepper».

Показано, що у випадку атаки проти вбудованого повідомлення зашумленням АБГШ стійкість стеганографічного методу є інваріантною до структури застосовуваного кодового слова. Стійкість методу може бути підвищеною за рахунок збільшення енергії кодового слова, що можливо із застосуванням багаторівневих кодових слів, при чому раціональним є застосуванням багаторівневих кодових слів, що здійснюють вибірковий вплив на високі частоти задля підвищення надійності сприйняття стеганоповідомлення і уникнення проблеми виникнення областей із перепадами яскравості на границях блоків.

В каналах з шумом типу «Salt and Pepper» було запропоновано застосування методу ШОВ, що передбачає обмеження по амплітуді матриць різниці стеганоповідомлення та оригінального контейнеру. Застосування даного методу дозволило знизити рівень помилок на 48% у порівняння із застосуванням оригінального стеганографічного методу з кодовим управлінням вбудовуванням ДІ.

Рекомендації, запропоновані у роботі, можуть бути використані для подальшого вдосконалення та оптимізації методу.

**Список літератури**


1. Мельник М.А. Стеганоалгоритм, устойчивый к сжатию. *Інформаційна безпека*. 2012. №2(8). С. 99-106.
2. Song X. et al. Robust JPEG steganography based on DCT and SVD in nonsubsampled shearlet transform domain. *Multimedia Tools and Applications*. 2022. Vol. 81. No. 25. P. 36453-36472.
3. Durafe A., Patidar V. Development and analysis of IWT-SVD and DWT-SVD steganography using fractal cover. *Journal of King Saud University-Computer and Information Sciences*. 2022. Vol. 34. No. 7. P. 4483-4498.
4. Arunkumar S. et al. SVD-based robust image steganographic scheme using RIWT and DCT for secure transmission of medical images. *Measurement*. 2019. Vol. 139. P. 426-437.
5. Pilania U., Tanwar R., Gupta P. An ROI-based robust video steganography technique using SVD in wavelet domain. *Open Computer Science*. 2022. Vol. 12. No. 1. P. 1-16.
6. Hussain M. et al. Image steganography in spatial domain: A survey. *Signal Processing: Image Communication.* 2018. Vol. 65. P. 46-66.
7. Rachael O. et al. Image steganography and steganalysis based on least significant bit (LSB). Proceedings of ICETIT 2019: Emerging Trends in Information Technology. Delhi, India: Springer International Publishing, 2020. P. 1100-1111.

8. Msallam M. M. A development of least significant bit steganography technique. *Iraqi journal of computers, communications, control and systems engineering*. 2020. Vol. 20. No. 1. P. 31-39.
9. Kobozeva A.A., Sokolov A.V. Robust Steganographic Method with Code-Controlled Information Embedding. *Problemele energeticii regionale*. 2021. No. 4 (52). P. 115-130.
10. NRCS Photo Gallery // United States Department of Agriculture. URL: https://www.nrcs.usda.gov/wps/portal/nrcs/main/national/newsroom/multimedia/
11. Грибунин В.Г., Оков И.Н., Туринцев И.В. Цифровая стеганография. М.: Солон-Пресс, 2002. 272 с.
12. Kobozeva A.A., Sokolov A.V. Theoretical foundations for constructing effective codewords for the code-controlled information embedding steganographic method. *Radiotekhnika*. 2021. 4(207). P. 27-39.

## INCREASING THE ROBUSTNESS OF THE STEGANOGRAPHIC METHOD WITH CODE CONTROL OF THE ADDITIONAL INFORMATION EMBEDDING AGAINST NOISE ATTACKS

D.O. Hulid, A.V. Sokolov

National Odesa Polytechnic University
Ukraine, Odesa, 65044, Shevchenko Ave., 1, radiosquid@gmail.com

The increase in the amount of multimedia content generated, stored and transmitted by modern information systems leads to an increase in the role of the steganographic component in information protection systems. At the same time, significant requirements are put to the applied steganographic methods regarding their effectiveness, which include sufficient bandwidth, ensuring the reliability of perception, and resistance to attacks against the embedded message. The performance of the steganographic methods is of great importance, especially if it is intended to be used in real-time on a resource-constrained platforms. One of the modern steganographic methods, which is characterized by providing the main effectiveness indicators with a small computational complexity due to the execution of steganographic transformation in the space domain of the container, is a steganographic method with code control of additional information embedding. However, despite the rather high indicators of resistance to noise attacks, which can occur in many practical applications, the issue of increasing the resistance of steganographic transformation to these types of attacks remains relevant. The purpose of the paper is to increase the resistance to noise attacks of the steganographic method with code control of additional information embedding. The paper researches attacks against an embedded message with two types of noise: additive white Gaussian noise and “Salt and Pepper” noise. In the paper, experimental research on the effects of the structure of the codeword used on the robustness of the steganographic method with code control against a noise attack by additive white Gaussian noise was performed, which made it possible to develop practical recommendations regarding the parameters of the steganographic method with code control under the conditions of this type of attack. The application of the BAN method (broadband signal, attenuator, narrowband signal) is proposed to increase the resistance of the steganographic method with code control against the "Salt and Pepper" noise attack, which allowed to reduce the number of errors when extracting additional information in conditions of this noise by 48%. The results obtained in this paper can be useful for the application of the steganographic method with code control in practice in the conditions of transmission of a steganographic message over channels characterized by the presence of noise attacks against the embedded message.
**Keywords:** steganography, code control of additional information embedding, noise attack, additive white Gaussian noise, “Salt and Pepper” noise.

DOI 10.15276/imms.v13.no1-2.63
УДК 519.242:519.25

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp.63-74

# МЕТОДИ ОПТИМІЗІЦІЇ І ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В АКТИВНОМУ ЭКСПЕРИМЕНТІ

Н. М. Єршова, Л. Ю. Кривенкова

Придніпровська державна академія будівництва та архітектури
м. Дніпро, 49000, вул. Чернишевського, 24 а E-mail:
nersova107@gmail.com, lyuk2406@i.ua

Незважаючи на те, що ефективність методів планування експерименту, особливо при вирішенні прикладних завдань, була неодноразово доведена, ідеї багатофакторного експерименту дуже повільно впроваджуються в науку та інженерну практику. Причини цього: складність організації проведення експерименту; параметри досліджуваних систем носять складний динамічний характер і схильні до суттєвих впливів змін умов зовнішнього середовища; уявна складність матриці планування та розрахунків відлякує дослідників з недостатньою математичною підготовкою; багато посібників із застосування багатофакторного експерименту написані на недоступному для інженерів рівні, враховуючи підготовку з дисциплін математичного та комп'ютерного циклів у технічних вузах. Експеримент займає чільне місце серед способів отримання інформації про внутрішні взаємозв'язки явищ у природі та техніці. У міру ускладнення досліджуваних процесів та явищ зростають витрати на апаратуру та проведення експерименту. У ході випробувань збирається велика кількість експериментальних даних, що потребують обробки та аналізу. При цьому тривалість аналізу, осмислення результатів випробувань та їхнього обліку для коригування характеристик нових виробів дуже значна. Системний підхід передбачає розглядати всі елементи активного експерименту як єдиної системи. З цих позицій необхідно представити загальні властивості експерименту як об'єкта дослідження та дати рекомендації щодо вибору математичних прийомів та методів, якими може користуватися експериментатор при виборі рішень у ході підготовки експерименту, його проведенні та обробки результатів. Дуже важливо при цьому вибрати методи та інструментальні засоби обробки даних експерименту. Незважаючи на те, що починаючи з 2002 року в багатьох роботах доводиться ефективність використання пакету аналізу Excel для обробки даних експерименту, дотепер у науковій та навчальній літературі використовується кодування змінних, рандомізація проведення дослідів, вибір критичних значень критеріїв оцінки якості математичних моделей за таблицями. Крім того, у роботах для прогнозування не використовуються методи оптимізації. У цій роботі з метою прискорення процесу впровадження в науку та інженерну практику ефективних комп'ютерних способів обробки наведено реалізацію методики обробки даних активного експерименту засобами Excel.

**Ключові слова:** експеримент, наука, інженерна практика, інструменти пакету аналізу Excel, методи оптимізації.

**Вступ.** Незважаючи на те, що ефективність методів планування експерименту, особливо при вирішенні прикладних завдань, була неодноразово доведена, ідеї багатофакторного експерименту дуже повільно впроваджуються в науку та інженерну практику. Причини цього:

- складність організації проведення експерименту;
- параметри досліджуваних систем носять складний динамічний характер і схильні до суттєвих впливів змін умов зовнішнього середовища;
- уявна складність матриці планування і розрахунків відлякує дослідників з недостатньою математичною підготовкою;
- багато посібників із застосування багатофакторного експерименту

написані на недоступному для інженерів рівні, враховуючи підготовку з дисциплін математичного та комп'ютерного циклів у технічних вузах.

Крім того, в існуючих роботах не використовуються методи оптимізації для визначення прогнозних значень параметрів досліджуваного об’єкту.

У додатку Excel є пакет аналізу, надбудова «Пошук рішення», майстер функцій та інши засоби, що значно полегшують і прискорюють процес обробки даних експерименту. Крім того, спрощується методика планування та проведення експерименту.

**Аналіз останніх досліджень і публікацій.** Основою основ будь-якого наукового дослідження є експеримент. За допомогою експериментальних досліджень [1]:

- дається обґрунтування новим науковим теоріям;
- відкриваються нові закони;
- розробляються та уточнюються існуючі методи розрахунків;
- удосконалюються та створюються нові матеріали, конструкції, технологічні процеси, апарати та механізми.

Теорія експерименту покликана дати відповіді досліднику на такі питання:

- як потрібно організувати експеримент, щоб якнайкраще вирішити поставлене завдання (у сенсі витрат часу та засобів або точності результатів);
- як слід обробляти результати експерименту, щоб отримати максимальну кількість інформації про досліджуваний об'єкт чи явище;
- які обґрунтовані висновки про об'єкт, що досліджується, можна зробити за результатами експерименту.

Дуже важливо при цьому вибрати методи та інструментальні засоби обробки даних експерименту. Незважаючи на те, що починаючи з 2002 року в роботах [7-10, 13] доводиться ефективність використання пакету аналізу Excel для обробки даних експерименту, досі в науковій та навчальній літературі при обробці даних експерименту використовується кодування змінних, рандомізація проведення дослідів, вибір критичних значень критеріїв оцінки якості математичних моделей за таблицями та різноманітні не завжди ефективні пакети прикладних програм [1-6, 11, 12, 14]. І навіть у роботі [16] з обнадійливою назвою «Методи обробки еасперименталь них даних з використанням MS EXCEL», опублікованої в 2019 році, для обробки даних використовуються статистичні функції майстра функцій і лише два інструменти пакету аналізу: Описова статистика та Гістограма.

**Мета роботи.** Отже метою даної роботи є прискорення процесу впровадження у науку та інженерну практику ефективних комп'ютерних засобів обробки даних експерименту. Для цього на конкретному прикладі розкрити можливості пакета аналізу і надбудови «Пошук рішень» Excel під час обробки даних активного есперименту.

**Основна частина.** Мета проведення експерименту – створення адекватної, статистично значимої моделі регресії.

При плануванні експерименту виникають три основні задачі: скільки проводити дослідів; які значення надавати факторам; у якому поєднанні різним факторам надавати різні значення.

Відповідно до роботи [1] число дослідів $n$ має задовольняти нерівності

$$k+1 \le n < 2^k,$$

де $k$ - кількість факторів.

Оцінка адекватності рівняння регресії проводиться за допомогою $F$ - критерію Фішера, розрахункове значення якого визначається відношенням факторної дисперсії (дисперсії адекватності) до залишкової дисперсії, тобто

$$F = s_{ad}^2 / s_o^2,$$

де $s_{ad}^2 = \sum_{i=1}^{n} (\overline{Y}_i - \overline{Y}_{xi})^2 / (k+1)$; $s_o^2 = \sum_{i=1}^{n} (y_i - \overline{Y}_{xi})^2 / (n-k-1)$; $\overline{Y}_i$ – середнє значення відгуку у кожному окремому досліді; $y_i$ – значення відгуку у паралельному досліді; $k$ - кількість факторів. Таким чином, оцінка адекватності моделі регресії можлива за умови, що кількість ступенів свободи залишкової дисперсії позитивно і не дорівнює нулю, тобто

$$n > k+1.$$

Отже, при $k = 7$ $n > 8$, тобто $n = 2^4 = 16$. Таким чином, теоретично до лінійної моделі регресії можна включати до 14 факторів і в матриці планування достатньо мати 16 дослідів.

При плануванні експерименту потрібно вибрати алгебраїчний многочлен, який містить невелику кількість параметрів і задовольняє вимогі адекватності. Багатофакторна лінійна модель має мінімальну кількість параметрів та має перевагу перед нелінійними моделями [4]. Завжди існує така околиця будь-якої точки, в якій лінійна модель адекватна.

Так як параметри моделі регресії визначаються за допомогою інструмента «Регресія» пакета аналізу, то можно виключити операції переходу до кодованих значень факторів і обернено. Достатньо скористатися розробленими матрицями планування з кодованими факторами і замість символів «+ або 1» записати значення верхнього рівня фактора, а замість символів «- або -1» - значення нижнього рівня фактора.

Щоб уникнути систематичних похибок і рівномірно розподілити чи усунути некеровані впливи на весь експеримент, досліди досі проводять не за порядком, закладеним в матриці планування, а у випадковій послідовності, яка визначається таблицею випадкових чисел (рандомізація дослідів).

У цієї роботі шляхом моделювання доведено, що послідовність розташування рядків у матриці планування не впливає на значення параметрів моделі регресії. Тому перед проведенням експерименту не потрібно витрачати час на виконання операції рандомізації дослідів.

У процесі проведення експерименту необхідно переконатися, що значення відгуку, що вимірюється, належать одній генеральній сукупності і технологічний процес не вимагає регулювання. Для цього кожний дослід матриці планування експерименту проводять кілька разів (паралельні досліди). Після проведення 4-х дослідів по матриці планування експерименту необхідно переконатися в однорідності одержуваних вибірок відгуку та можливості відтворюваності дослідів.

Для оцінки адекватності рівняння регресії експериментальним даним залишкова дисперсія відгуку порівнюється з дисперсією фактичних значень відгуку, тобто оцінка адекватності відповідає оцінці однорідності вибірок відгуку.

Критеріями порівняння вибірок служать: рівність двох вибіркових дисперсій і рівність двох вибіркових середніх. У вихідній інформації інструменту «Однофакторний дисперсійний аналіз» видаються розрахункове $F$ та критичне значення $F_{kp}$ критерію Фішера. Гіпотеза про рівність дисперсій підтверджується, якщо $F \le F_{kp}$. Для порівняння двох вибіркових середніх використовують $t$ - статистику Стьюдента. Гіпотеза про рівність середніх значень підтверджується, якщо $|t| \le t_{kp}$. У разі негативних результатів слід відрегулювати прилади вимірювання, повторити досліди та їхню обробку. Після підтвердження однорідності вибірок відгуків продовжити досліди з матриці планування.

Після завершення дослідів слід виконати кореляційно-регресійний аналіз багатовимірної вибірки, у якій замість відгуку має бути його середнє значення, отримати модель регресії та оцінити її якість, тобто встановити статистичну значущість моделі регресії. Перевірка статистичної значущості рівняння регресії проводиться за $F$ - критерієм Фішера. Якщо $F > F_{kp}$, то є хороша відповідність до даних експерименту.

Оцінити статистичну значущість параметрів моделі регресії $a_j$ означає встановити, чи відрізняється значення параметра моделі регресії від нуля. Якщо $|t| > t_{kp}$, то нульову гіпотезу про рівність нулю параметра моделі регресії відкидають і параметр вважають значним. Існують додаткові критерії оцінки статистичної значущості параметрів моделі регресії: метод $P$ – значення та значення нижньої та верхньої меж довірчого інтервалу.

Вихідна інформація інструменту «Регресія» містить усі параметри та оцінки якості моделі регресії.

Приклад 1. Виконати обробку таких даних, отриманих в результаті активного експерименту: $y$ - міцність на стиск бетону віком 28 діб; $x_1$ - цементо/водне (Ц/В) відношення бетону М200-М400; $x_2$ - активність цементу $R_ц$, МПа; $x_3$ - модуль крупності $M_{кр}$; $x_4$ - зміст домішок, що відмучуються, $Q_o$.

У таблиці 1 наведено рівні та інтервал зміни факторів.

**Таблиця 1**

Умови планування експерименту

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | | фактор | | | Рівень фактору | | Інтервал |
| 2 | натуральний | кодований | | -1 | 0 | 1 | варіювання |
| 3 | Ц/В | x1 | | 1,4 | 2 | 2,6 | 0,6 |
| 4 | Rц, Мпа | x2 | | 38,8 | 45,3 | 51,8 | 6,5 |
| 5 | Мкр | x3 | | 1,4 | 2,2 | 3 | 0,8 |
| 6 | Qo | x4 | | 1 | 3 | 5 | 2 |

У таблиці 2 наведено матрицю планування експерименту. У ході експерименту кожен дослід повторювали тричі, тому перевіряємо однорідність одержаних вибірок.

Діалогове вікно інструменту «Однофакторний дисперсійний аналіз» представлено на рис. 1, вихідна інформація – у таблиці 3.

**Таблиця 2**

Матриця планування у натуральному вигляді

| | A | B | C | D | E | F | G | H |
|---|---|---|---|---|---|---|---|---|
| 7 | | | Матриця планування | | | | | |
| 8 | x1 | x2 | x3 | x4 | y1 | y2 | y3 | Yср |
| 9 | 2,6 | 51,8 | 3 | 5 | 44,2 | 43 | 43,6 | 43,6 |
| 10 | 2,6 | 51,8 | 3 | 1 | 49 | 49,6 | 47,5 | 48,7 |
| 11 | 2,6 | 51,8 | 1,4 | 5 | 42 | 39,6 | 41,1 | 40,9 |
| 12 | 2,6 | 51,8 | 1,4 | 1 | 45 | 44 | 44,2 | 44,4 |

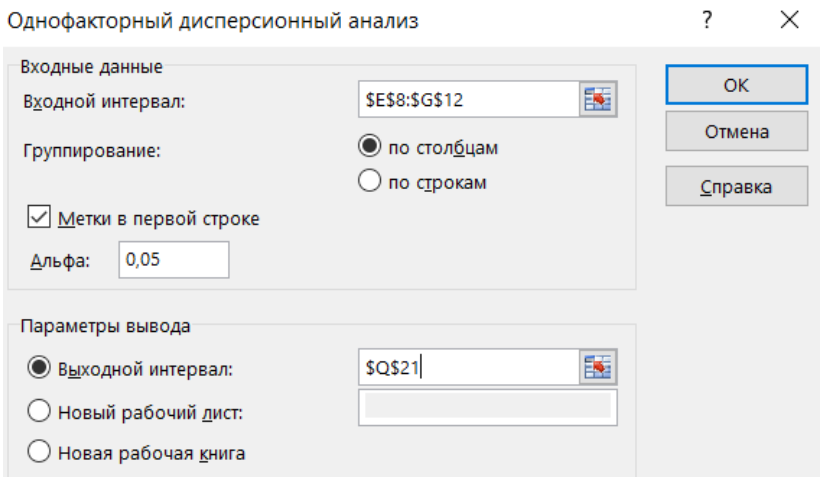



**Рис. 1.** Діалогове вікно інструменту «Однофакторний дисперсійний аналіз»

**Таблиця 3**

Вихідна інформація

| Однофакторный дисперсионный анализ | | | | | | |
|---|---|---|---|---|---|---|
| ИТОГИ | | | | | | |
| *Группы* | *Счет* | *Сумма* | *Среднее* | *Дисперсия* | | |
| y1 | 4 | 180,2 | 45,05 | 8,5433333 | | |
| y2 | 4 | 176,2 | 44,05 | 17,236667 | | |
| y3 | 4 | 176,4 | 44,1 | 6,94 | | |
| Дисперсионный анализ | | | | | | |
| *Источник вариации* | *SS* | *df* | *MS* | *F* | *P-Значение* | *F критическое* |
| Между группами | 2,54 | 2 | 1,27 | 0,1164425 | 0,891400465 | 4,256494729 |
| Внутри групп | 98,16 | 9 | 10,9067 | | | |
| | | | | | | |
| Итого | 100,7 | 11 | | | | |

Оскільки $F \le F_{kp}$, то вибірки однорідні і можна продовжувати досліди. Після завершення всіх дослідів виконуємо кореляційно-регресійний аналіз. В якості $y$ виступає середнє значення міцності бетону на стиск $Y_{cp}$ (табл. 4).

**Таблиця 4**

Матриця планування експерименту

| | K | L | M | N | O | P |
|---|---|---|---|---|---|---|
| 2 | x1 | x2 | x3 | x4 | Ycp | Yx |
| 3 | 2,6 | 51,8 | 3 | 5 | 43,6 | 43,794 |
| 4 | 2,6 | 51,8 | 3 | 1 | 48,7 | 45,781 |
| 5 | 2,6 | 51,8 | 1,4 | 5 | 40,9 | 41,631 |
| 6 | 2,6 | 51,8 | 1,4 | 1 | 44,4 | 43,619 |

Перевіряємо відповідність відгуку до нормального закону розподілу. Діалогове вікно інструменту «Гістограма» наведено на рис. 2, інструменту «Описова статистика» – на рис. 4, гістограма – на рис. 3, результати – у табл. 5.

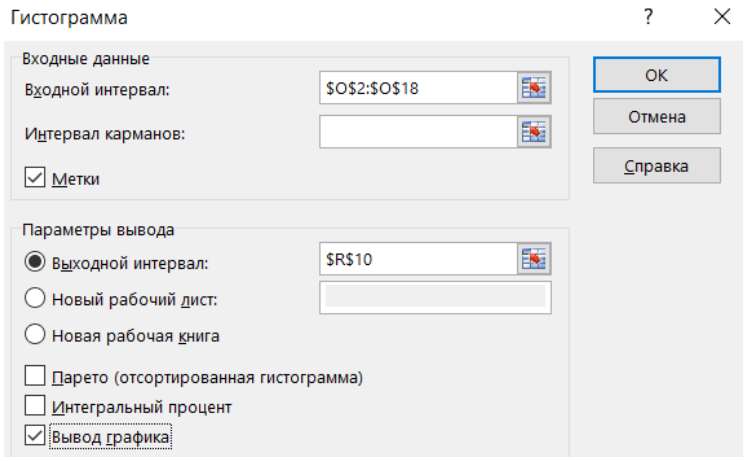



**Рис. 2.** Діалогове вікно інструменту «Гістограма»

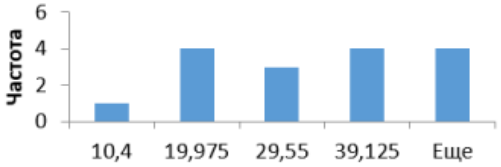



**Рис. 3.** Гістограма міцності бетону на стиск

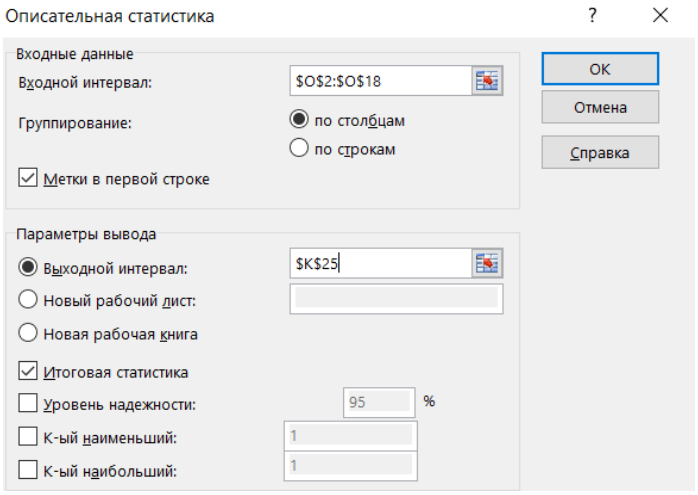



**Рис. 4.** Діалогове вікно інструменту «Описова статистика»

**Таблиця 5**

Статистична обробка вибірки міцності бетону

| | K | L |
|---|---|---|
| 25 | *Ycp* | |
| 26 | | |
| 27 | Среднее | 27,256 |
| 28 | Стандартная ошибка | 3,2237 |
| 29 | Медиана | 26,1 |
| 30 | Мода | #Н/Д |
| 31 | Стандартное отклонение | 12,895 |
| 32 | Дисперсия выборки | 166,28 |
| 33 | Эксцесс | -1,3047 |
| 34 | Асимметричность | 0,1975 |
| 35 | Интервал | 38,3 |
| 36 | Минимум | 10,4 |
| 37 | Максимум | 48,7 |
| 38 | Сумма | 436,1 |
| 39 | Счет | 16 |

На вигляд гістограми і по значенням числових характеристик можна припустити, що міцність бетону на стиск підпорядковується нормальному закону розподілу.

**Кореляційний аналіз даних експерименту.** Матриця коефіцієнтів парної кореляції (табл. 6) отримана за допомогою інструмента «Кореляція», діалогове вікно якого наведено на рис. 5.

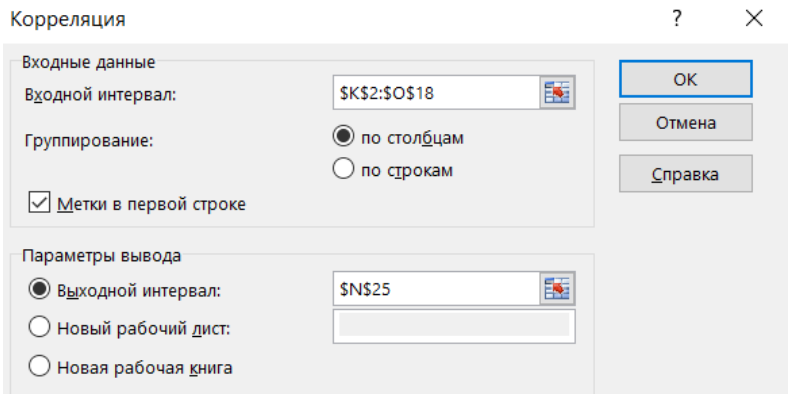



**Рис. 5.** Діалогове вікно інструмента «Кореляція»

**Таблиця 6**

Матриця коефіцієнтів парної кореляції

| | *x1* | *x2* | *x3* | *x4* | *Ycp* |
|---|---|---|---|---|---|
| x1 | 1 | | | | |
| x2 | -1E-17 | 1 | | | |
| x3 | 0 | -4E-17 | 1 | | |
| x4 | 0 | 0 | 0 | 1 | |
| Ycp | 0,8935 | 0,424 | 0,0866016 | -0,0796 | 1 |

Як очевидно з кореляційної матриці фактори незалежні між собою. Між фактором Ц/В і міцністю бетону на стиск спостерігається сильний кореляційний зв'язок, зв'язок міцності з активністю цементу помітний, а з модулем крупності і вмістом домішок, що відмучуються, практично відсутній.

**Регресійний аналіз даних експерименту**. Загальний вигляд моделі багатовимірної лінійної регресії міцності бетону на стиск має вигляд:

$$\overline{Y}_x = a_0 + a_1x_1 + a_2x_2 + a_3x_3 + a_4x_4,$$

де $a_0, a_1, ..., a_4$ - невідомі параметри моделі регресії. Оптимальні значення цих параметрів визначаємо на основі методу найменших квадратів, математичний запис якого:

$$S(a_0, a_1, a_2, a_3, a_4) = \sum_{i=1}^{16}(a_0 + a_1x_{1i} + a_2x_{2i} + a_3x_{3i} + a_4x_{4i} - y_i)^2 \Rightarrow \min.$$

Конкретні значення параметрів моделі регресії та розрахункові значення критеріїв якості визначаємо за допомогою інструмента «Регресія», вихідна інформація якого наведена у таблиці 7, діалогове вікно – на рис. 6.

Отже, модель багатовимірної лінійної регресії має вигляд:

$$\overline{Y}_x = -48{,}35 + 18{,}583x_1 + 0{,}815x_2 + 1{,}359x_3 - 0{,}493x_4.$$

Виконаємо аналіз якості отриманої моделі регресії: $R^2 = 0{,}992$, отже, 99,2% дисперсії значень міцності пояснюється впливом факторів; розрахункове значення критерію Фішера $F$ =343,619; критичне значення критерію Фішера $F_{kp} = 3{,}356$. Отже, рівняння регресії загалом статистично значуще, тобто є хороша відповідність даним експерименту.

**Таблиця 7**

Вихідна інформація інструменту Регресія

| ВЫВОД ИТОГОВ | | | | | | |
|---|---|---|---|---|---|---|
| *Регрессионная статистика* | | | | | | |
| Множественный R | 0,996010332 | | | | | |
| R-квадрат | 0,992036582 | | | | | |
| Нормированный R-к | 0,989140793 | | Fkp | tkp | | |
| Стандартная ошибка | 1,343735465 | | 3,356690021 | 2,20098516 | | |
| Наблюдения | 16 | | | | | |
| Дисперсионный анализ | | | | | | |
| | *df* | *SS* | *MS* | *F* | *Значимость F* | |
| Регрессия | 4 | 2474,2775 | 618,569375 | 342,579093 | 1,84512E-11 | |
| Остаток | 11 | 19,861875 | 1,805625 | | | |
| Итого | 15 | 2494,139375 | | | | |
| | *Коэффициенты* | *Стандартная ошибка* | *t-статистика* | *P-Значение* | *Нижние 95%* | *Верхние 95%* |
| Y-пересечение | -48,30742788 | 2,820520199 | -17,12713417 | 2,8011E-09 | -54,515351 | -42,099505 |
| x1 | 18,59375 | 0,559889777 | 33,20966155 | 2,2042E-12 | 17,36144091 | 19,8260591 |
| x2 | 0,814423077 | 0,051682133 | 15,75830999 | 6,7741E-09 | 0,700671469 | 0,92817469 |
| x3 | 1,3515625 | 0,419917333 | 3,218639466 | 0,00817884 | 0,427330683 | 2,27579432 |
| x4 | -0,496875 | 0,167966933 | -2,958171533 | 0,01301778 | -0,86656773 | -0,1271823 |

Критичне значення $t$ - статистики $t_{kp} = 2{,}2$. Для всіх параметрів $a_j$ моделі регресії розрахункове значення $t$ - статистики більше критичного значення, тобто всі параметри моделі регресії статистично значущі. Про це свідчать: $P$ - значення ( $P$ <0,05) і межі довірчого інтервалу (нижні 95% и верхні 95%) цих параметрів. Отже, нульова гіпотеза про те, що параметри $a_j$ моделі регресії можуть набувати нульових значень, відкидається. Коефіцієнт множинної кореляції дорівнює 0,996. Він є критерієм оцінки точності функції регресії.

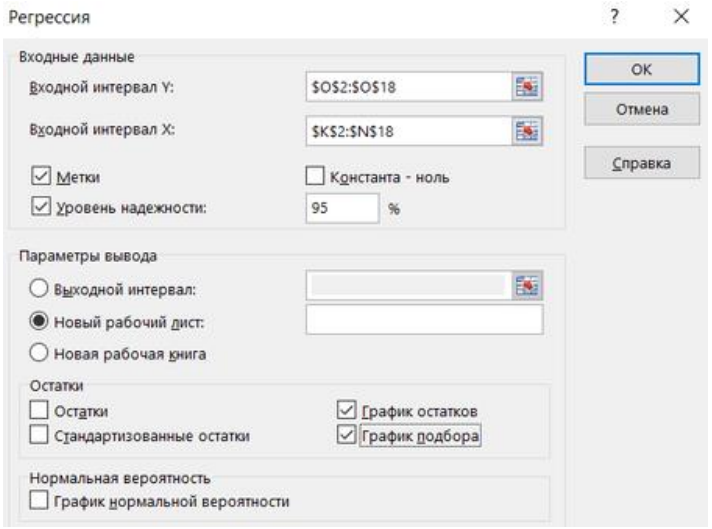



**Рис. 6.** Діалогове вікно інструменту Регресія

Критичні значення F – критерію та t – статистики визначаємо за статистичними функціями F.ОБР.ПХ и СТЬЮДЕНТ.ОБР.2Х майстра функцій.

Для приклада, що розглядається, модель регресії можна використовувати для прийняття рішень і прогнозування.

**Прогнозування на основі рівняння регресії.** У вихідній інформації інструменту "Регресія" видаються значення точкового прогнозу для всієї матриці планування експерименту (див. табл. 4). Максимальна міцність бетону на стиск, що дорівнює 45,788 МПа, отримана у другій точці матриці планування з координатами $x_1 = 2{,}6; x_2 = 51{,}8; x_3 = 3; x_4 = 1$.

**Прогнозування на основі методу лінійного програмування.** Математична модель оптимізації.

Цільова функція

$$\overline{Y}_x = -48{,}35 + 18{,}583x_1 + 0{,}815x_2 + 1{,}359x_3 - 0{,}493x_4 \rightarrow \max$$

Обмеження: $1{,}4 \le x_1 \le 2{,}8;\quad 38{,}8 \le x_2 \le 51{,}8;\quad 1{,}4 \le x_3 \le 3;\quad 1 \le x_4 \le 5.$

Оптимізацію виконуємо за допомогою надбудови "Пошук рішення". Результати оптимізації представлені у таблиці 8.

**Таблиця 8**

Результати оптимізації

| | X | Y | Z | AA | AB |
|---|---|---|---|---|---|
| 20 | x0 | x1 | x2 | x3 | x4 |
| 21 | 1 | 2,6 | 51,8 | 3 | 1 |
| 22 | a0 | a1 | a2 | a3 | a4 |
| 23 | -48,35 | 18,6 | 0,8154 | 1,3594 | -0,494 |
| 24 | | значение целевой функции | | | |
| 25 | | | 45,788 | | |
| 26 | x1n | x2n | x3n | x4n | |
| 27 | 1,4 | 38,8 | 1,4 | 1 | |
| 28 | x1v | x2v | x3v | x4v | |
| 29 | 2,6 | 51,8 | 3 | 5 | |

Точка оптимального плану розташована у вершині багатогранника безлічі допустимих рішень $x_1 = 2{,}6; x_2 = 51{,}8; x_3 = 3; x_4 = 1$. Міцність бетону на стиск дорівнює 45,788 МПа.

**Прогнозування на основі методу покоординатного пошуку**. Метод покоординатного пошуку належить до методів нульового порядку, у яких обчислюються лише значення цільової функції. Відповідно до цього методу напрямок пошуку вибирають паралельно координатним осям. Спочатку здійснюють пошук уздовж першої осі $Ox_1$, потім уздовж другої осі $Ox_2$ і так далі до останньої осі $Ox_n$. Потім цикл повторюють, починаючи з першої змінної.

Таким чином, спочатку задається початкова точка $X^{(0)} = (x_1^0, x_2^0, \ldots, x_n^0)$, в якій визначаються значення цільової функції та обмежень. Потім послідовно змінюються значення змінної $x_1$ на величину кроку, а інші змінні $x_2, x_3, \ldots, x_n$ фіксуються на своїх початкових рівнях $x_2^{(0)}, x_3^{(0)}, \ldots, x_n^{(0)}$. У діапазоні $[x_1^н, x_1^в]$ (нижня і верхня межа) знаходиться точка, що задовольняє обмеженням і має максимальне (мінімальне) значення цільової функції серед точок цієї осі. Відповідне значення змінної $x_1$ фіксують і починають послідовно змінювати значення змінної $x_2$ на величину кроку, доки знайдеться точка цієї осі з найкращим значенням цільової функції. Фіксують значення змінної $x_2$ і роблять аналогічно з третьою змінною і так далі до змінної $x_n$. Потім цикл послідовної зміни змінних проводять заново, починаючи з $x_1$. Ця процедура повторюється доти, доки не буде знайдена точка, зміщення щодо якої зміною будь-якої змінної призводить лише до погіршення результату. Вона береться за точку екстремуму.

Приклад 2. На основі регресійного аналізу, наведеного у прикладі 1, отримана модель багатовимірної лінійної регресії міцності на стиснення бетону у віці 28 діб.

Методом покоординатного пошуку визначити значення факторів, які забезпечують отримання максимального значення міцності бетону на стиск.

Математична модель

Цільова функція $F(X) = -48{,}35 + 18{,}583\,x_1 + 0{,}815x_2 + 1{,}359x_3 - 0{,}493x_4 \to \max$

Обмеження: $1{,}4 \le x_1 \le 2{,}8$; $38{,}8 \le x_2 \le 51{,}8$; $1{,}4 \le x_3 \le 3$; $1 \le x_4 \le 5$.

Розв'язування. Результати розрахунку в середовищі ЕТ наведено у таблиці 9.

**Таблиця 9**

Оптимізація методом покоординатного пошуку

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | x1n | x1v | h1 | x2n | x2v | h2 | |
| 2 | 1,4 | 2,6 | 0,3 | 38,8 | 51,8 | 2 | |
| 3 | x3n | x3v | h3 | x4n | x4v | h4 | |
| 4 | 1,4 | 3 | 0,4 | 1 | 5 | 1 | |
| 5 | | | | | | | |
| 6 | a0 | a1 | a2 | a3 | a4 | | |
| 7 | -48,35 | 18,5833 | 0,8154 | 1,3594 | -0,494 | | |
| 8 | x0 | x1 | x2 | x3 | x4 | Yx | |
| 9 | 1 | 1,4 | 38,8 | 1,4 | 1 | 10,7133 | |
| 10 | 1 | 1,7 | 38,8 | 1,4 | 1 | 16,2883 | |
| 11 | 1 | 2 | 38,8 | 1,4 | 1 | 21,8633 | |
| 12 | 1 | 2,3 | 38,8 | 1,4 | 1 | 27,4383 | |
| 13 | 1 | 2,6 | 38,8 | 1,4 | 1 | 33,0133 | |
| 14 | 1 | 2,6 | 39,8 | 1,4 | 1 | 33,8287 | |
| 15 | 1 | 2,6 | 40,8 | 1,4 | 1 | 34,6441 | |
| 16 | 1 | 2,6 | 41,8 | 1,4 | 1 | 35,4595 | |
| 17 | 1 | 2,6 | 42,8 | 1,4 | 1 | 36,2749 | |
| 18 | 1 | 2,6 | 43,8 | 1,4 | 1 | 37,0903 | |
| 19 | 1 | 2,6 | 44,8 | 1,4 | 1 | 37,9057 | |
| 20 | 1 | 2,6 | 45,8 | 1,4 | 1 | 38,7211 | |
| 21 | 1 | 2,6 | 46,8 | 1,4 | 1 | 39,5365 | |
| 22 | 1 | 2,6 | 47,8 | 1,4 | 1 | 40,3519 | |
| 23 | 1 | 2,6 | 48,8 | 1,4 | 1 | 41,1673 | |
| 24 | 1 | 2,6 | 49,8 | 1,4 | 1 | 41,9827 | |
| 25 | 1 | 2,6 | 50,8 | 1,4 | 1 | 42,7981 | |
| 26 | 1 | 2,6 | 51,8 | 1,4 | 1 | 43,6135 | |
| 27 | 1 | 2,6 | 51,8 | 1,8 | 1 | 44,1572 | |
| 28 | 1 | 2,6 | 51,8 | 2,2 | 1 | 44,701 | |
| 29 | 1 | 2,6 | 51,8 | 2,6 | 1 | 45,2447 | |
| 30 | 1 | 2,6 | 51,8 | 3 | 1 | 45,7885 | максимум |
| 31 | 1 | 2,6 | 51,8 | 3 | 2 | 45,2945 | |

В якості початкової точки виберемо точку, координати якої відповідають нижній межі факторів, тобто $X^{(0)} = (1{,}4;\ 38{,}8;\ 1{,}4;\ 1)$. Визначимо цільову функцію у цій точці:

$$f(X^{(0)}) = -48{,}35 + 18{,}5833x_1^{(0)} + 0{,}8154x_2^{(0)} + 1{,}3594x_3^{(0)} - 0{,}494x_4^{(0)} =$$

$$= -48{,}35 + 18{,}5833 \cdot 1{,}4 + 0{,}8154 \cdot 38{,}8 + 1{,}3594 \cdot 1{,}4 - 0{,}494 \cdot 1 = 10{,}7125.$$

Будемо здійснювати пошук вздовж першої координатної осі, і приймемо довжину кроку в цьому напрямку рівною $h_1 = 0{,}3$, тоді наступна точка матиме координати $X^{(1)} = (1{,}7;\ 38{,}8;\ 1{,}4;\ 1)$. Значення цільової функції у цій точці $f(X^{(1)}) = 16{,}288$, тобто збільшується. Отже, потрібно здійснювати пошук у цьому напрямі. Так як модель лінійна і перед фактором $x_1$ коефіцієнт додатний, значення цільової функції буде з кожним кроком збільшуватися, досягаючи при $x_1 = 2{,}6$ значення 33,013. Зафіксуємо це положення як початкову точку другої координатної осі $X^{(0)} = (2{,}6;\ 38{,}8;\ 1{,}4;\ 1)$ і йдемо вздовж цієї осі, допустимо довжина кроку дорівнює $h_2 = 1$, тобто маємо точку $X^{(1)} = (2{,}6;\ 39{,}8;\ 1{,}4;\ 1)$ і значення цільової функції в цій точці $f(X^{(1)}) = 33{,}828$. Значення цільової функції збільшується $f(X^{(1)}) > f(X^{(0)})$, тобто можна йти у цьому напрямі. Перед фактором $x_2$ цільової функції коефіцієнт додатний і її значення буде з кожним кроком збільшуватися, досягаючи при $x_2 = 51{,}8$ значення 43,613. Зафіксуємо це положення як початкову точку третьої координатної осі $X^{(0)} = (2{,}6;\ 51{,}8;\ 1{,}4;\ 1)$ і йдемо вздовж неї, допустимо довжина кроку дорівнює $h_3 = 0{,}4$, тобто маємо точку $X^{(1)} = (2{,}6;\ 51{,}8;\ 1{,}8;\ 1)$ і значення цільової функції в цій точці $f(X^{(1)}) = 44{,}157$. Значення цільової функції збільшується $f(X^{(1)}) > f(X^{(0)})$, отже, йдемо у цьому напрямі. Перед фактором $x_3$ цільової функції коефіцієнт додатний

і її значення буде з кожним кроком збільшуватися, досягаючи при $x_3 = 3$ значення 45,788. Зафіксуємо це положення як початкову точку четвертої координатної осі $X^{(0)} = (2{,}6;\ 51{,}8;\ 3;\ 1)$ і йдемо вздовж неї, допустимо довжина кроку дорівнює $h_4 = 1$, тобто маємо точку $X^{(1)} = (2{,}6;\ 51{,}8;\ 3;\ 2)$ і значення цільової функції в цій точці $f(X^{(1)}) = 45{,}294$. Значення цільової функції зменшується $f(X^{(1)}) < f(X^{(0)})$, отже, потрібно повернутися на один крок і це буде оптимальним рішенням. Отже, максимальне значення міцності бетону на стиск дорівнює 45,788 МПа і отримано воно при $x_1 = 2{,}6;\ x_2 = 51{,}8;\ x_3 = 3;\ x_4 = 1$. Таким чином, прогноз по рівнянню регресії та методам оптимізації збігається.

**Висновки.** Обробка даних експерименту засобами Excel дозволяє відмовитись від: кодування факторів; рандомізації дослідів; визначення за формулами параметрів та критеріїв оцінки якості моделі регресії; таблиць щодо визначення критичних значень $F$ - критерію та $t$ - статистики; ручної реалізації чисельних методів оптимізації.

Це значно прискорює процеси планування, проведення та обробки результатів експерименту та дає відповіді досліднику на такі питання:

– як потрібно організувати експеримент, щоб якнайкраще вирішити поставлене завдання (у сенсі витрат часу та засобів або точності результатів);

– як слід обробляти результати експерименту, щоб отримати максимальну кількість інформації про досліджуваний об'єкт або явище;

– які обґрунтовані висновки про об'єкт, що досліджується, можна зробити за результатами експерименту.

Використання інструментальних засобів Excel робить активний експеримент привабливим та прискорить його впровадження у наукові дослідження та інженерну практику.

**Список літератури**


1. Барабащук В. И., Креденцер Б. П., Мирошниченко В. И. Планирование эксперимента в технике. К.: Техніка, 1984. 200 с.
2. Вознесенский В. Статистические решения в технологических задачах. Кишинев: Картя Молдовеняскэ, 1969.232 с.
3. Вознесенский В. А., Ляшенко Т. В., Огарков Б. Л. Численные методы решения строительно-технологических задач на ЭВМ. К.: Выща школа, 1989. 328 с.
4. Гарькина И. А., Данилов А. М., Прошин А. П., Бормотов А. Н. Применение математических методов в строительном материаловедении. Пенза: ПГАСА, 1999. 204 с.
5. Гарькина И. А., Данилов А. М., Прошин А. П. Математические методы синтеза строительных материалов. Пенза: ПГАСА,2001. 106 с.
6. Дворкин Л. И., Шамбан И. Б. Проектирование составов бетона с применением математического моделирования. К.: УМК ВО, 1992. 144 с.
7. Ершова Н. М. Реализация в среде электронных таблиц методов корреляционно-регрессионного анализа и прогнозирования. Днепропетровск: ПГАСА, 2002. 50 с.
8. Єршова Н. М. Свідоцтво про реєстрацію авторського права на твір наукового характеру «Алгоритм планирования, проведения и обработки активного эксперимента». №39237 від 19.07.2011.
9. Ершова Н. М., Деревянко В. Н., Тимченко Р. А., Шаповалова О. В. Обработка данных средствами Excel при планировании эксперимента: учеб. пособие для вузов Д.: ПГАСА, 2012. 350 с.
10. Ершова Н. М. Дисперсионный анализ данных наблюдений. Днепропетровск: ПГАСА, 2010. 80 с.

11. Красовский П. С. Исследование и оптимизация свойств строительных материалов с применением элементов математической статистики. Хабаровск: ДВГУПС, 2004. 128 с.
12. Пінчук С. Й. Організація експерименту при моделюванні та оптимізації технічних систем. Д.: Дніпро-VAL, 2009. 289 с.
13. Сивец С. А. Статистические методы в оценке недвижимости и бизнеса. Учебно-практическое пособие по статистике для оценщиков. Запорожье, 2001. 320 с.
14. Бородюк В. П., Вощинин А. П., Иванов А. З. Статистические методы в инженерных исследованиях. /Под ред. Г .К. Круга. М.: Высш. школа, 1983. 216 с.
15. Ферстер Э., Ренц Б. Методы корреляционного и регрессионного анализа: Руководство для экономистов. М.: Финансы и статистика, 1983. 302 с.
16. Горват А. А., Молнар О. О., Мінькович В. В. Методи обробки експериментальних даних з використанням MS EXCEL. Ужгород: Видавництво УжНУ «Говерла», 2019. 160 с.

## OPTIMIZATION METHODS AND INFORMATION TECHNOLOGIES IN ACTIVE EXPERIMENT

N. M. Yershova, L. Yu. Kryvenkova

Prydniprovska State Academy of Civil Engineering and Architecture
st. Chernishevsky, 24, Dnipro, 49000, Ukraine
E-mail: nersova107@gmail.com, lyuk2406@i.ua

Despite the fact that the effectiveness of experiment planning methods, especially when solving applied problems, has been proven more than once, the ideas of a multifactorial experiment are very slowly being introduced into science and engineering practice. The reasons for this are: the complexity of organizing the experiment; the parameters of the systems under study are of a complex dynamic nature and are subject to significant influences of changes in environmental conditions; the apparent complexity of the planning and calculation matrix discourages researchers with insufficient mathematical background; many manuals on the application of a multifactorial experiment are written at a level inaccessible to engineers, given the training in the disciplines of mathematical and computer cycles in technical universities. The experiment occupies the main place among the methods of obtaining information about the internal relationships of phenomena in nature and technology. As the processes and phenomena under study become more complex, the costs of the equipment and the experiment increase. During the tests, a large amount of experimental data is collected that requires processing and analysis. At the same time, the duration of the analysis, comprehension of the test results and their accounting for adjusting the characteristics of new products is very significant. The systems approach involves considering all elements of an active experiment as a single system. From these positions, it is necessary to present the general properties of the experiment as an object of study and give recommendations on the choice of mathematical techniques and methods that the experimenter can use when choosing decisions during the preparation of the experiment, its conduct and processing of the results. It is very important to choose methods and tools for processing experimental data. Despite the fact that, since 2002, many works have been proving the effectiveness of using the Excel analysis package for processing experimental data, the scientific and educational literature still uses coding of variables, randomization of experiments, and the choice of critical values of the criteria for assessing the quality of mathematical models from tables. In addition, optimization methods are not used in the works for forecasting. In this paper, in order to accelerate the process of introducing effective computer processing methods into science and engineering practice, the implementation of the methodology for processing active experiment data using Excel is presented.

**Keywords:** experiment, science, engineering practice, Excel analysis package tools, optimization methods.

DOI 10.15276/imms.v13.no1-2.75
УДК 004.42:004.056.57

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp. 75-86

# СИСТЕМИ АНАЛІЗУ КІБЕРБЕЗПЕКИ ПРОГРАМНО-ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ КОРПОРАТИВНИХ КОМП'ЮТЕРІВ ТА САЙТІВ

В.В. Задоров, Д.О. Фурман, О.А.Стопакевич, А.О.Стопакевич

Національний університет «Одеська політехніка»,
Проспект Шевченка, 1, Одеса, 65044, Україна; E-mail:
stopakevich@gmail.com

Корпоративні комп'ютери та сайти є основними мішенями для атак кіберзлочинців, оскільки вони містять цінну інформацію та забезпечують функціонування бізнес-процесів. Підприємства повинні бути готовими виявляти та запобігати потенційним загрозам, щоб уникнути фінансових втрат, втрати репутації та порушенням конфіденційності даних клієнтів. У цьому контексті системи аналізу кібербезпеки програмно-технічного забезпечення корпоративних комп'ютерів та сайтів виконують важливу роль. Вони дозволяють проводити повноцінний аудит інфраструктури, виявляти потенційні вразливості, оцінювати ризики та розробляти стратегії захисту. Ці системи надають комплексну інформацію про стан безпеки та допомагають приймати обґрунтовані рішення щодо поліпшення безпеки. В статті розглянуто дві програмні системи аналізу кібербезпеки. Перша програмна система забезпечує аналіз апаратного та програмного забезпечення комп'ютерів, виявляючи потенційні проблеми безпеки на рівні окремих пристроїв. Вона надає зручний графічний інтерфейс десктопної програми й дозволяє проводити докладний аналіз комп'ютерів, включаючи оцінку апаратного забезпечення та встановлених програм, і надає звіти про виявлені проблеми та рекомендації щодо поліпшення безпеки. Друга програма, реалізована як телеграм-бот, надає можливість віддалено аналізувати корпоративні сайти та сервери, спрощуючи процес контролю безпеки для спеціалістів з кібербезпеки та системних адміністраторів. Інтерфейс телеграм-бота дозволяє проводити віддалену діагностику довільного сервера без необхідності доступу до нього, встановлення та регулярного оновлення спеціального програмного забезпечення, що робить бот зручним для використання. Бот забезпечує аналіз різних складових сайту та серверу, на якому він розміщений, включаючи версію операційної системи сервера, сервіси сервера, CMS сайту, піддомени сайту, репутацію сервера та сайту та геолокацію сервера. Результати аналізу надаються у звіті, що допомагає виявити потенційні вразливості та надає рекомендації з покращення безпеки.

**Ключові слова:** корпоративна кібербезпека, сайт, сервер, аудит, програма, звіт, телеграм, бот, віддалена діагностика, проблеми безпеки, сервіс, вразливості.

**Вступ.** Актуальність розробки систем автоматизованого аналізу кібербезпеки набуває особливої ваги в сучасному цифровому світі, оскільки технології зламу постійно удосконалюються й стає все важчим слідкувати за змінами програмного та апаратного забезпечення корпоративних комп’ютерів та сайтів [1]. Зловмисники постійно шукають нові способи зламу інформаційних систем, використовуючи вразливості в програмному та апаратному забезпеченні. Новим викликом є розвиток штучного інтелекту, що відкриває нові можливості для створення систем зламу [2].

Стаття розглядає розв’язання двох задач аудиту кібербезпеки: задачі аудиту апаратного та програмного забезпечення корпоративних комп’ютерів та задачі аудиту корпоративних сайтів та серверів, на яких ці сайти розміщені.

Метою розв’язання першої задачі є виявлення потенційних проблеми безпеки на рівні окремих пристроїв. А саме, проблеми з апаратними компонентами, програмним забезпеченням, драйверами, патчами та іншими аспектами комп'ютерної інфраструктури, що дозволяє вчасно вжити заходів для виправлення помилок, забезпечуючи належну продуктивність та функціональність системи. Оскільки, багато компаній підлягають регуляторним вимогам щодо безпеки, захисту персональних даних, фінансової звітності та інших аспектів, то аудит комп'ютерів допомагає встановити відповідність цим вимогам та підтвердити, що комп'ютерна інфраструктура відповідає потрібним стандартам і регуляціям. Також аудит комп'ютерів може допомогти виявити зайве апаратне та програмне забезпечення, неефективне використання ресурсів, недієві процеси та інші фактори, які призводять до надмірних витрат. Аналізуючи результати аудиту, можна вжити не тільки заходи щодо кіберпезпеки, но і заходи по оптимізації забезпечень та зменшенню витрат.

Метою розв’язання другої задачі є виявлення потенційних проблем безпеки корпоративних сайтів. А саме: відсутність актуальних версій програмного забезпечення серверу та сайту, надмірна кількість відкритої інформації про програмне забезпечення серверу, відкритий доступ до служб, які призначені для застосування в локальних мережах, наявність працюючих служб з застарілими протоколами чи таких, які не відповідають цільовому призначенню корпоративних серверів, застосування небезпечних та застарілих CMS, перевантаженість сервера сайтами, відсутність сертифікату чи ненадійний центр сертифікації, низька репутація сервера та сайту, геолокація сервера в недружніх до України країнах й країнах, в яких не дотримуються законодавчих норм щодо доступу до приватної інформації. Наслідки зламу корпоративних сайтів можуть бути дуже неприємними, зокрема: зупинка роботи сайту; крадіжка та розповсюдження персональних даних користувачів, фінансових даних, логінів та паролів; втрата цінної інформації, яка складає корпоративну таємницю. Якщо злам стає відомим, то це, звичайно, має негативний вплив на ділову діяльність та репутацію компанії. Втрата довіри користувачів може вплинути на лояльність клієнтів, зниження трафіку та втрату бізнесу. Крім того, багато країн та регуляторних органів встановлюють обов'язкові вимоги щодо захисту даних, конфіденційності та приватності даних користувачів на веб-сайтах. Невиконання цих вимог може призвести до штрафів, правових санкцій та інших негативних наслідків для підприємства. Регулярний аудит сайтів дозволяє також оцінити причини низької швидкості доступу до сайтів, виявити моральну застарілість застосованого програмістами та системними адміністраторами програмного забезпечення та технологій програмування.

**Система аналізу кібербезпеки програмно-технічного забезпечення корпоративних комп'ютерів**. Вимогою до системи є наявність зручного графічного інтерфейсу, швидкість роботи та адекватний перелік параметрів для порівняння. Програмні інтерфейси ОС Windows не відрізняються високою швидкістю отримання параметрів, що пов’язано з архітектурними особливостями операційної системи (ОС) [3, 4]. Інформація про кожний компонент в певний момент часу не доступна для ОС й має отримуватись через інтерфейси відповідних драйверів, доступ до який знаходиться через відповідні ключі реєстру. Ця інформація також не записується в певній централізованій базі при запуску операційної системи, коли ініціалізуються

драйвери. Тому звичайно програми для отримання системної інформації або відображують тільки базову інформацію або структурують відображення по категоріям та підкатегоріям, щоб користувач міг отримати тільки конкретну інформацію, коли це потрібно. Отримання всієї інформації, яку можливо отримати про комп'ютер за допомогою інтерфейсів ОС, звичайно займає на порівняно сучасному ПК біля 10 хвилин. Тому програма для аудиту має вибирати тільки ті параметри, які відносяться до значимих, не змінюються кожного запуску ОС. Перелік досліджуваних параметрів має бути невеликий й однозначно зрозумілий користувачеві програми.

Для ще більшого спрощення роботи з програмою найбільш типові перевірки кібербезпеки системою мають виконуватись за запитом. Для цього треба реалізувати окрему функцію – автоматичне проведення аудиту. В результаті аналізу отримана інформація (як поточна конфігурація, так і її зміни) аналізується з точки зору кібербезпеки. Аналіз зазначає які результати перевірки відповідають вимогам кібербезпеки ПК, які їх порушують, а які вимагають додаткового ручного аналізу.

Розроблена система, яка відповідає зазначеним вимогам, реалізована як програма мовою Python [5, 6], з орієнтацією на застосування в ОС Windows. Проводиться отримання таких значимих параметрів як:

– апаратне забезпечення комп'ютера (BIOS, процесору, пам'яті, дисків тощо);

– основні параметри операційної системи (номера збірки, дати встановлення, папки встановлення тощо);

– перелік встановленого програмного забезпечення (видалення, додавання, зміна версії);

– перелік програмного забезпечення, яке запускається разом з ОС та при вході в акаунт;

– перелік встановлених оновлень операційної системи (ОС) Windows;

– наявність та активності брандмауера Windows, антивірусів Windows Defender та інших основних виробників, актуальності задіяної бази даних антивірусів.

Система відповідає наступним ключовим технологічним вимогам:

– можливість збереження поточного переліку параметрів комп'ютера, як еталонну конфігурацію в портабельному та зручному для обробки форматі JSON;

– можливість порівняння поточної конфігурації та будь-якої зі збережених конфігурацій в якості еталонної;

– час отримання необхідного переліку параметрів не перевищує 1 хв.;

– проведення автоматичного аудиту з генерацією звіту оновлень операційної системи, наявності та активності брандмауера Windows, антивірусів Windows Defender та інших основних виробників, актуальності задіяної бази даних антивірусів.

При розробці системи використані наступні інструменти програмної інженерії:

– мова програмування Python 3.10 в дистрибутиві Anaconda;

– бібліотека для реалізації графічного інтерфейсу PyTK;

– інтерфейси ОС для доступу до WMI [7], реєстру [8] тощо;

– утиліта autoruns [9], результати якої використовуються для гарантованого знаходження всіх програм в автозапуску ОС (що, при бажанні гарантованого результату, не є простою справою);

– технології інтернаціоналізації програми.

Під час роботи з системою аудитор може зберігати поточну конфігурацію як еталонну й порівнювати в наступний раз зміни в комп'ютері відносно неї. Для збереження конфігурації в системі застосований зручний портабельний формат JSON.

Приклад перевірки конфігурації комп'ютера після оновлення його технічного забезпечення приведений на рис.1.

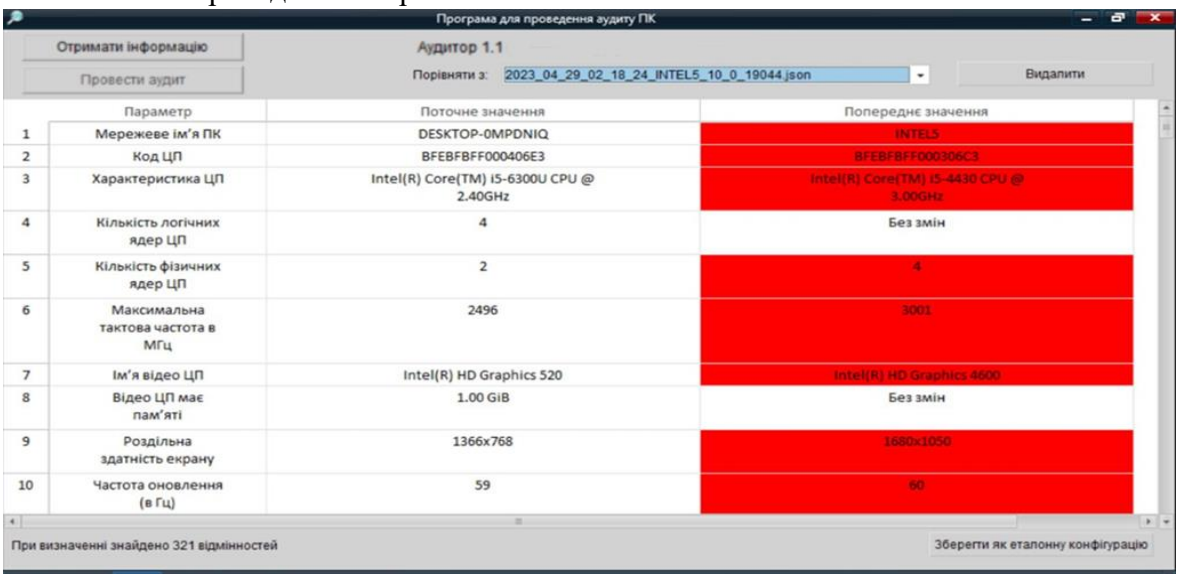



**Рис.1.** Приклад перевірки конфігурації комп'ютера після оновлення його технічного забезпечення

Приклад перевірки конфігурації комп'ютера після оновлення його програмного забезпечення приведений на рис. 2.

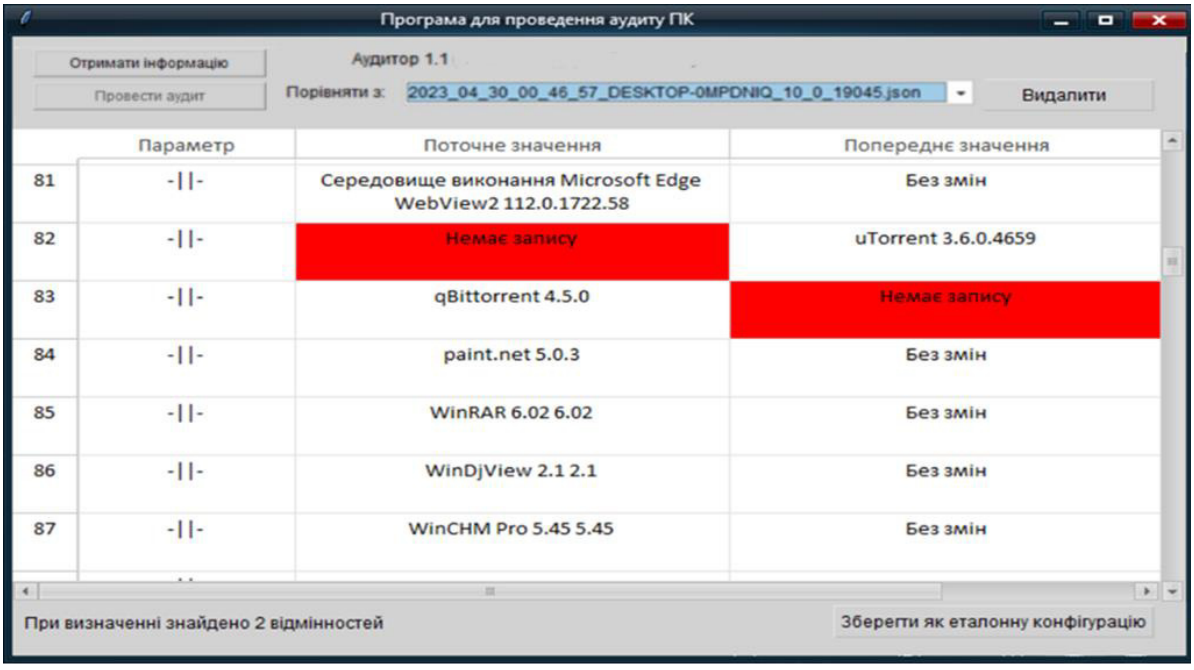



**Рис.2.** Приклад перевірки конфігурації комп'ютера після оновлення його програмного забезпечення

Для спрощення роботи користувачів, система може автоматично згенерувати звіт аудиту. Приклад звіту приведено на рис. 3.

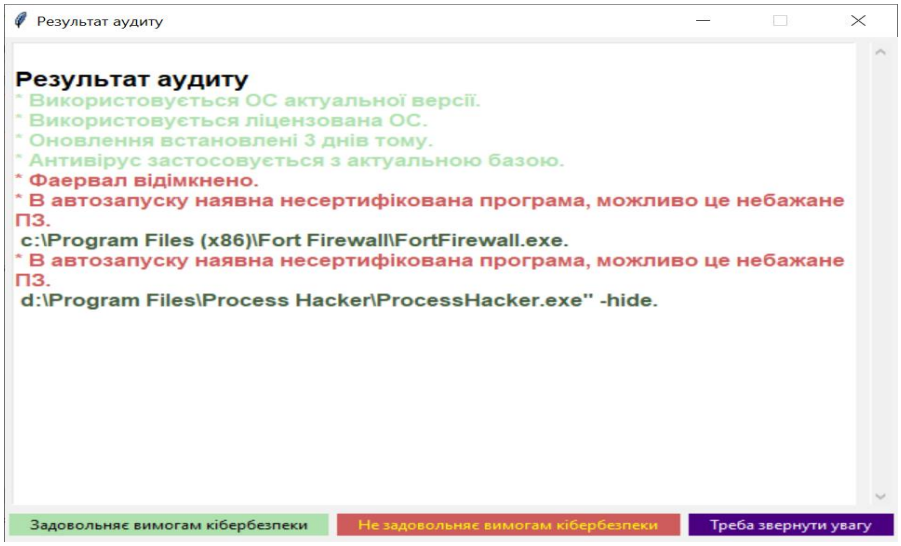



**Рис.3.** Приклад автоматично згенерованого звіту

При проведенні аудиту при умові, що проблем не знайдено, аудитор має зберегти поточну конфігурацію в файл як еталонну, назва якого за замовченням містить мережеве ім'я комп'ютера та дату. В наступний раз аудитор вже звіряє конфігурацію з тою, що збережена. Якщо зміни позитивні, то нова конфігурація зберігається як еталонна.

**Система аналізу кібербезпеки програмного забезпечення корпоративних сайтів**. Вимогою до системи є забезпечити зручний для користувача швидкий аудит сайту й серверу, на якому він розміщений, з метою визначення ступеню захисту й основних можливих проблем. При цьому спеціаліст оцінює сайт та серверне програмне забезпечення (ПЗ) ззовні, тобто аналізує яка інформація може бути доступна зацікавленому хакеру. Визначення стану безпеки ззовні має певні недоліки – отримана інформація може бути не повна чи навіть недостовірна, якщо адміністратор спеціально над цим працював. Однак, істотна перевага такого визначення – незалежність від адміністратора сервера й програмістів сайтів, які можуть й не знати про факт такої перевірки. Оцінку вірності прийняти рішень системним адміністратором сервера та програмістами сайту проводить програма, таким чином виникає можливість погляду на ситуацію збоку.

Критеріями вразливості сайтів при зовнішній діагностиці є наступні [10]:

– Використання застарілої операційної системи на сервері сайту. Як правило, програмне забезпечення (в тому числі для реалізації інтернет-сервісів) для застарілих ОС, особливо в світі UNIX-подібних ОС, не оновлюється, це призводить до того що знайдені вразливості в старших версіях не виправляються й цим можуть скористатись хакери.

– Використання застарілих служб на сервері сайту. Причина така ж сама – служба, версія якої не підтримується може стати джерелом зламу

– Використання на сервері сайту служб, що реалізують застарілі протоколи. Кожна служба, яка доступна користувачу в інтернеті є джерелом потенційної небезпеки. Використовувати служби, які реалізують неактуальний функціонал в загальному випадку непотрібно й краще знаходити сучасні альтернативи

– Надмірна кількість сайтів на сервері. Кожен сайт може мати вразливе ПЗ, через яке можливо потенційно отримати доступ до всього серверу. Чим більше сайтів, ти такий ризик більший. Якщо на корпоративному сайті зберігається важлива інформація, то треба уникати його розміщення на хостингу с сотнями сайтів на одному сервері.

– Надмірна кількість субдоменів. Субдомени виділяються для служб (наприклад, ftp чи smtp) та сайтів. Іноді адміністратори зловживають субдоменами, розміщуючи в інтернет непотрібний контент

– Використання CMS. Використання типового й відомого програмного забезпечення є джерелом небезпеки. Звичайно, різні CMS мають різний ступінь захисту та якість виконання, вимоги до адміністрування тощо. Сайти з CMS звичайно зламуються в першу чергу.

– Репутація сайту. Інтернет служби та пошукові системи мають власні інструменти для визначення факту зламу сайтів, також є спеціальні сервіси, в яких розміщують жалоби користувачів на аномальну активність серверів.

– Географічна локація сайту**.** Хоча вона не є прямим показником кібербезпеки, однак розміщення сайтів в деяких країнах є небезпечним оскільки законодавство там не дуже діє, тому доступ до інформації можуть отримати фізично чи шляхом встановлення спеціальних аналізаторів трафіку за місцем встановлення серверу.

Для проведення зовнішньої діагностики розроблено багато програмних утиліт, тому при розробленні системи не доцільно займатись розробкою нових інструментів, а бажано застосовувати максимальну кількість відпрацьованих рішень. Призначення системи буде в тому, щоб проаналізувати отримані оцінки й представити їх в зрозумілій формі аудитору.

Для визначення операційної системи доцільно застосовувати результати програми nmap [11-13], яка використовує механізм fingerprinting. База даних програми nmap складається з ідентифікаторів різних ОС та відповідних до них маркерів. Обсяг БД складається більше ніж з 7000 тисяч записів. Слід зазначити, що процедура визначення принципово має ймовірнісний характер, тобто, наприклад Linux 6.2 [14, 15] може бути з ймовірністю 90%, а Linux 6.0-6.4 з ймовірністю 89% і т.п. Крім того, адміністратори серверів можуть застосовувати методи фальсифікації показів [16, 17]. В роботі [18] наведено літературний огляд методів підвищення ймовірності визначення. Для підвищення точності оцінювання треба перевірити достовірність оцінки, орієнтуючись на співпадіння з іншими визначеними параметрами (службами ОС, веб-сервером, застосованими серверними мовами програмування тощо). Уточнення версії також досягається за рахунок застосування методів аналізу трафіку HTTPS [19] і DNS.

Для визначення та оцінки сервісів операційної системи також доцільно орієнтуватись в першу чергу на алгоритми програми nmap, які застосовують декілька механізмів визначення переліку працюючих сервісів та програмного забезпечення, що їх реалізує.

Оцінка застосованих сервісів ОС за призначенням доцільно проводити виходячи з наступної класифікації [3, 13, 14, 15]:

1. Сервіси, нормальні для інтернет-сервера, які є актуальні й рекомендовані: ftps, ftps, ftps-data, ftps-data, smtp, http, https, imap, imap4-ssl, imaps, xmpp, xmpp-client, http-proxy, http-alt,https-alt, submission

2. Сервіси, нормальні для інтернет-сервера, протоколи яких застарілі й такі сервіси краще замінити на більш безпечні аналоги (наприклад, telnet на ssh, irc на

мессенджери та ін.): ftp-data, ftp, telnet, rtelnet, tam, nntp, tftp, via-ftp, gopher, irc, talk, conference, rtsp, rsync, telnets, ircs, pop3, pop3s, time, scp, scp-config

3. Сервіси для віддаленого управління сервером, які мають контролюватись брандмаурером чи не бути віддалено бути доступні взагалі: ssh, vnc, vnc-1, vnc-2, vnc-3, vnc-http,, vnc-http-1, vnc-http-2, vnc-http-3,teamviewer, ms-wbt-server, msrpc

4. Стандартні мережеві сервіси операційних систем, сервіси баз даних, X-сервер та ін., все що має не бути доступно через мережу інтернет й має в більшості випадків використовуватись лише в межах локальної мережі: sqlserv, sqlnet, netbios-ns, netbios-dgm, netbios-ssn, snmp, ipx, microsoft-ds, printer, http-rpc-epmap, login, dhcpv6-client, dhcps, nfs, mysql, mysql-proxy, postgresql, ms-olap2, ms-olap1, rsqlserver, rsqlserver, dns, x11, X11, X11:1,X11:2,X11:3,X11:4, X11:5, X11:6, X11:7, X11:8, X11:9

5. Сервіси майнінгу, які не мають бути присутніми на сервері: bitcoin, litecoin.

6. Спеціальні сервіси для розробників, наявність яких на корпоративному інтернет сервері скоріше не бажані: git, cvspserver, cvsup. Для цих сервісів краще виділити окремий сервер, якщо це дозволяє фінансування.

З точки зору безпеки краще, якщо не можливо визначити програмне забезпечення, яке реалізує сервіс чи хоча б версію програми. Для оцінки достовірності оцінки програмного забезпечення необхідно додатково для критичних служб застосовувати додаткові алгоритми. Наприклад, для веб-сервера слід перевірити додатково версії протоколу HTTP, що підтримуються (наприклад, HTTP/2 підтримують далеко на всі веб-сервери). Заголовки веб-серверів в деяких випадках містять інформацію про операційну систему, наприклад в дистрибутиві Ubuntu за замовчуванням в заголовці HTTP сервера міститься повна версія дистрибутиву. Перехресна перевірка дозволяє уточнити точність визначення ОС.

Визначення та оцінка CMS сайту [20, 21] може спрощено проводитись за допомогою аналізу HTML коду. Однак більш точні дані надає сервіс whatcms. Цей сервіс дозволяє визначити CMS з високою долею ймовірністю. Однак все ж таки дані треба перевірити на відповідність з простим алгоритмом оцінки за HTML кодом.

Визначення та оцінка переліку піддоменів може проводитись як за аналізом SAN запису SSL сертифікату [10], так і брутфорс методом за допомогою перебору за словником програмою subbrute [22]. Другий метод є більш надійний, оскільки сертифікат часто видають не на окремі піддомени, а на всі разом. Велика кількість піддоменів звичайно є небезпечною, часто в піддоменах розміщують певні тестові версії сайтів, за якими потім не слідкують.

Визначення переліку інших сайтів, розміщених на сервері та оцінка їх кількості може проводитись за допомогою механізму зворотного запису DNS. Найбільш зручний сервіс визначення надає служба Reverse2IP. Кожний сайт, розміщений на сервері, є джерелом небезпеки. Тому краще для корпоративного сайту, який містить важливі персональні дані не розміщуватись на дешевих хостингах з сотнями інших сайтів.

Визначення геолокації серверу може бути проведена за допомогою безкоштовного сервісу ip2geotools. В деяких країнах не бажано розміщення сайтів підприємств України й перевірка за переліком ненадійних країн є бажаною.

Визначення репутації серверу може бути проведена за допомогою сервісу abusedb. Цей сервіс автоматично проводить оцінку шкідливості та небезпеки сайту за 100 бальною шкалою. Отримання інформації з подібних сервісів дозволяє оцінити чи

був факт зламу серверу чи використання його для незаконної чи недозволеною в мережі діяльності за період в 2-3 останні роки.

З використанням зазначеного вище переліку потенційних вразливостей сайтів та запропонованих методів їх визначення була розроблена система, яка реалізована як Python програма з інтерфейсом чат-боту мессенджера Telegram [23, 24]. Бот проводить віддалену діагностику, не залежить від адміністраторів сайту та не вимагає дозволу для доступу до сервера. Для використання боту обов'язкова реєстрація користувачів та робиться детальний звіт всіх дії користувачів. Дані отримуються за допомогою зовнішніх інструментів та різна інформація про конкретний сайт оновлюється з різними інтервалами з метою захисту від DDoS атак.

Для розробки бота використана версія системи програмування Python 3.10 з дистрибутива Anaconda. Використано Aiogram – сучасний та повністю асинхронний фреймворк для API Telegram Bot, написаний на Python за допомогою інтерфейсу asyncio. Для збереження інформації о користувачах використана локальна СКБД SqLite.

Інтерфейс головного вікна програми приведений на рис.4.

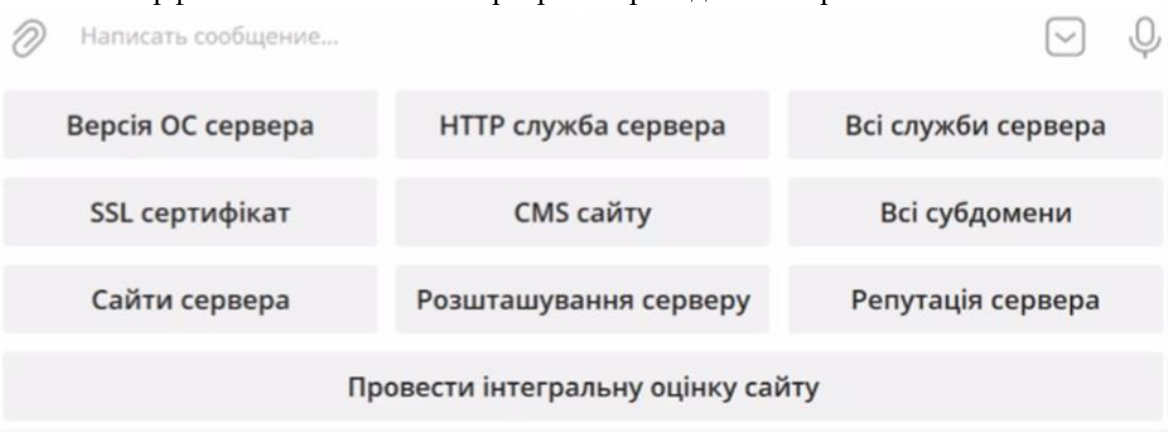



**Рис.4.** Меню інтерфейсу чат-боту

Деякі фрагменти з результатів аналізу системи приведені нижче.

1. **Аналіз версії ОС сервера де розміщено сайт ukr.net**

Операційна система сервера
Ймовірність присутності ОС:
Linux 5.0 - 5.4 - 87 процентів
Якщо ця версія вірна: Використовується актуальна версія ядра Linux. Це безпечно.

2. **Аналіз служб сервера де розміщено сайт ukr.net**

Всі сервіси сервера
Знайдені служби:
TCP 80: http Cloudflare http proxy – Це безпечна для інтернет-сервера служба.
TCP 443: https cloudflare – Це безпечна для інтернет-сервера служба.
TCP 8080: http Cloudflare http proxy – Це безпечна для інтернет-сервера служба.
TCP 8443: https-alt cloudflare – Це безпечна для інтернет-сервера служба.
Використовуються дані програми nmap

3. **Аналіз SSL сертифікату сайту ukr.net**

Криптографічний захист сайту
Сертифікат працюючий. На сайт можна зайти за допомогою HTTPS
Результати детальної перевірки сертифікату

Сертифікат дійсний в період : 2023-02-07 00:00:00-2024-02-07 23:59:59
Спільне ім'я сертифікату: ukr.net
SAN: *.ukr.net, ukr.net
Сертифікат видано: Cloudflare Inc ECC CA-3
Це безпечно

**4. Аналіз CMS сайту ukr.net**

CMS сайту
Результати перевірки за допомогою аналізу HTML коду – CMS не знайдена
Інформація від сервісу whatcms: – CMS не знайдена
Це безпечно

**5. Аналіз всіх субдоменів сайту ukr.net**

Піддомени сайту
За SSL сертифікатами – *.ukr.net, ukr.net
Спеціальний агресивний метод за словником зі 100 слів (програма subbrute):
ukr.net, www.ukr.net, mail.ukr.net, ftp.ukr.net, localhost.ukr.net, webmail.ukr.net, smtp.ukr.net, pop.ukr.net, ns1.ukr.net, …. , sms.ukr.net, office.ukr.net, exchange.ukr.net. ipv.ukr.net
Кількість піддоменів завелика, можливо деякі з них непотрібні для зовнішніх користувачів.
Не зовсім безпечно

**6. Аналіз наявності інших сайтів на сервері де розташовано сайт ukr.net**

Знайдено сайтів:21:
a678ff.com, acromegalie.nl, … , ukr.net, ukr.net.ua, ukrnet.net.ua, vismasolutions.com, www.aussiebeeflamb.com, vn.cdn.cloudflare.net, www.leatest.site, xodi.bet, yslbeauty.sa
Чим більше сайтів на сервері, тим ймовірніше його злам через програмне забезпечення. Кількість сайтів представляє небезпеку. Це небезпечно. Рекомендується змінити сервер.

**7. Аналіз розташування сервера, де розташовано сайт ukr.net**

Географічна локація сервера
Місто – Toronto, Регіон: Ontario, Країна: CA, Широта: 43.6534817, Довгота: -79.3839347
IP2Geotools
Це не дуже безпечно для сайту, який призначений для обслуговування переважно України

Фрагмент інтегральної оцінки сайту op.edu.ua приведено на рис. 5.

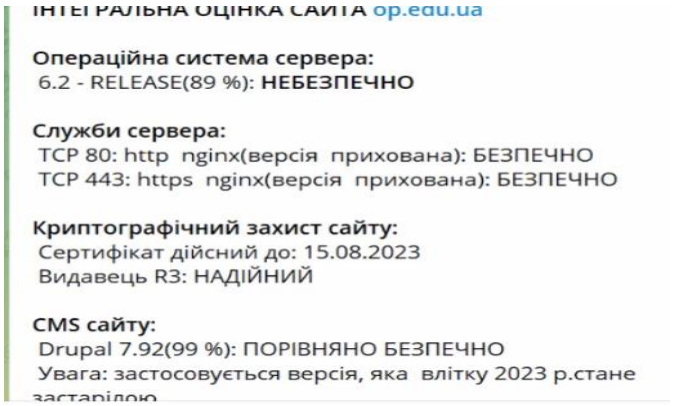



**Рис.5.** Фрагмент інтегральної оцінки сайту op.edu.ua

Таким чином, Демонстрація роботи програми при діагностиці різних сайтів показує успішність вибраних інструментів та алгоритмів оцінки складових безпеки корпоративних сайтів.

**Висновки.** Розглянуті дві задачі аудиту кібербезпеки: аналіз апаратного та програмного забезпечення комп'ютерів і аналіз корпоративних сайтів та серверів.

Метою першої задачі було виявлення потенційних проблем безпеки на рівні окремих комп'ютерних пристроїв, що дозволяє вжити заходів для виправлення помилок і забезпечити безпеку і надійність комп'ютерної інфраструктури. Виконуючи аудит комп'ютерів, можна встановити відповідність регуляторним вимогам, забезпечити захист персональних даних, фінансової звітності та інших аспектів, а також знизити витрати шляхом оптимізації ресурсів та процесів.

Перша задача розв’язана шляхом розробки програми для аудиту корпоративних комп’ютерів, основними відмінностями якої є швидкість роботи та автоматичний аналіз зміни конфігурації та її змін з точки зору спеціаліста з кібербезпеки.

Друга задача полягала у виявленні потенційних проблем безпеки корпоративних сайтів та серверів. Аудит сайтів дозволяє виявити такі проблеми, як застаріле програмне забезпечення серверів, недотримання законодавчих вимог, відкритий доступ до служб, використання небезпечних та застарілих CMS і багато інших. Злам корпоративних сайтів може мати серйозні наслідки, включаючи втрату конфіденційної інформації, зупинку роботи сайту та негативний вплив на репутацію компанії. Аналізуючи результати аудиту, можна прийняти необхідні заходи для забезпечення безпеки сайту та виконання вимог щодо захисту даних користувачів.

Друга задача розв’язана шляхом розробки програми для аудиту корпоративних сайтів, основними відмінностями якої є визначення наявності потенційних вразливостей шляхом віддаленої діагностики. Програма реалізована як чат-бот. Особливістю програми є використання алгоритмів оцінки достовірності отриманих з різних джерел даних та наявність зручної для спеціаліста з кібербезпеки інтегральної оцінки безпеки сайту, яка не вимагає наявності доступу до сервера, на якому він розміщений.

Розробка систем автоматизованого аналізу кібербезпеки має велике значення в сучасному цифровому світі, де зловмисники постійно шукають нові способи зламу інформаційних систем. Застосування таких систем дозволяє вчасно виявляти потенційні загрози та вразливості, вживати відповідні заходи для їх усунення і забезпечувати безпеку та надійність комп'ютерних систем і корпоративних сайтів. Запровадження аудиту кібербезпеки є необхідним кроком для захисту важливих даних, забезпечення дотримання регуляторних вимог і зміцнення довіри до організацій в сфері кібербезпеки.

**Список літератури**

1. Boskamp E. 30 crucial cybersecurity statistics [2023]: data, trends and more. Zippia. URL: https://www.zippia.com/advice/cybersecurity-statistics/
2. Meah J. AI in Cybersecurity: The Future of Hacking is Here. Technopedia. https://www.techopedia.com/ai-in-cybersecurity-the-future-of-hacking-is-here/2/34520
3. Panek C. Windows Server Administration Fundamentals. Hoboken, NJ: Willey & Sons, 2020

4. Yosifovich P., Ionesku A., Russinovich M. E., Solomon D. A. Windows internals. Part1: System architecture, processes, threads, memory management, and more. Microsoft Press, 2017.
5. Васильєв О. Програмування мовою Python. Київ: Навчальна книга – Богдан, 2019.
6. Руденко В., Жугастров О. Основи алгоритмізації та програмування мовою Python. Київ: Ранок, 2019.
7. Lissoir A. Leveraging WMI Scripting: Using Windows Management Instrumentation to Solve Windows Management Problems. Digital Press, 2003.
8. Palne L. The Defender's Guide to the Windows Registry. URL: https://posts.specterops.io/the-defenders-guide-to-the-windows-registry-febe241abc7
9. Systntelnals autoruns. URL : https://download.sysinternals.com/files/Autoruns.zip
10. Dalziel H. How to Attack and Defend Your Website. N.Y.: Elsevier, 2015
11. Chauhan A.S. Practical Network Scanning: Capture network vulnerabilities using standard tools such as Nmap and Nessus. Packt Publishing, 2018
12. Brown N. Nmap 7: From Beginner to Pro. Packt Publishing, 2021
13. Calderon P. Nmap Network Exploration and Security Auditing Cookbook. Packt Publishing, 2021
14. Calcatinge A., Balog J. Mastering Linux Administration: A Comprehensive Guide to Installing, Configuring, and Maintaining Linux Systems in the Modern Data Center. Packt Publishers, 2021
15. Рамський Ю., Олексюк В., Балик А. Адміністрування комп'ютерних мереж та систем. Тернопіль: Богдан, 2010.
16. Kalia S., Singh M. Masking approach to secure systems from operating system fingerprinting. *TENCON*. 2005. Vol.1. No.6. P.21-24.
17. Greenwald L.G., Thomas T.J. Understanding and preventing network device fingerprinting. *Bell Lab Tech J*. 2007. Vol. 3. P. 149–166.
18. Kumar A., Soni I., Kumar, M. Operating System Fingerprinting Using Machine Learning. Algorithms for Intelligent Systems. Singapore: Springer, 2022. P. 157-161. URL: http://doi.org/10.1007/978-981-16-7136-4_13
19. Lastovicka M., Spacek S., Velan P., Celeda P. Using TLS Fingerprints for OS Identification in Encrypted Traffic. IEEE/IFIP Network Operations and Management Symposium. 2020. P.1–6. URL: https://doi.org/10.1109/NOMS47738.2020.9110319
20. Barker D. Web Content Management: Systems, Features, and Best Practices. Sebastopol, CA: O'Reilly, 2016
21. Jain N. WordPress Website Security Guide. Oxford: IP, 2019
22. Prakhar P. Mastering modern Web penetration testing. Birmingham, U.K.: Packt Publishing, 2016.
23. Modrzyk N. Building Telegram Bots: Develop Bots in 12 Programming Languages Using the Telegram Bot API. Tokyo: Apress, 2019
24. Демиденко А. Telegram Bot. Руководство по созданию бота в мессенджере Telegram. SelfPub, 2023

В.В. Задоров, Д.О. Фурман, О.А.Стопакевич, А.О.Стопакевич

# CYBER SECURITY ANALYSIS SYSTEMS OF SOFTWARE AND TECHNICAL SUPPORT OF ENTERPRISE COMPUTERS AND SITES

V.V.Zadorov, D.O.Furman, O.A.Stopakevych, A.O.Stopakevych

National Odesa Polytechnic University, Shevchenko str., 1, Odesa, 65044, Ukraine
stopakevich@gmail.com

Enterprise computers and websites are prime targets for cybercriminals because they contain sensitive information and provide business processes. Businesses must be prepared to spot and prevent potential threats to avoid financial losses, reputational damage, and violations of customer data privacy. In this context, cybersecurity analysis systems for software and hardware of enterprise computers and websites play an important role. They allow to conduct a full-fledged infrastructure audit, discover potential vulnerabilities, assess risks, and develop protection strategies. These systems provide comprehensive information about the security status and help to make reasonable decisions on how to improve security. The article discusses two software systems for cybersecurity analysis. The first software system analyzes computer hardware and software, identifying potential security issues at the level of individual devices. It provides a user-friendly graphical interface of a desktop program and allows to perform a detailed analysis of computers, including an assessment of hardware and installed programs, and provides reports on the detected problems and recommendations for improving security. The second program, implemented as a Telegram bot, allows remote analysis of corporate websites and servers, simplifying the security control process for cybersecurity specialists and system administrators. The Telegram bot interface allows remote diagnostics of an arbitrary server without the need to access it, install and regularly update special software, which makes the bot comfortable to use. The bot analyzes various components of the website and the server on which it is hosted, including the server operating system version, server services, website CMS, website subdomains, server and website reputation, and server geolocation. The analysis results are presented in a report that helps identify potential vulnerabilities and provides recommendations for improving security.

**Keywords**: corporate cyber security, site, server, audit, software, telegram boat, remote diagnostics, security problem, service, vulnerability.

DOI 10.15276/imms.v13.no1-2.87
УДК 004.42:004.056.57

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp. 87-96

# СИСТЕМИ АВТОМАТИЗОВАНОГО ВИБОРУ СКЛАДОВИХ ПРОГРАМНОГО ТА АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ КІБЕРБЕЗПЕКИ КОРПОРАТИВНИХ КОМП'ЮТЕРІВ

А.В.Князєв, Р. І.Назаренко, О.А.Стопакевич, А.О.Стопакевич

Національний університет «Одеська політехніка»,
Проспект Шевченка, 1, Одеса, 65044, Україна; E-mail:
stopakevich@gmail.com

Забезпечення ефективної кібербезпеки для корпоративних комп'ютерів є надзвичайно важливою та складною задачею, яка вимагає ретельного вибору програмного та апаратного забезпечення. У даній статті розглядаються ключові компоненти програмного та апаратного забезпечення систем кібербезпеки – антивірусні програми та апаратні файрволи (мережеві екрани). З огляду на швидкий розвиток кіберзагроз, стає зрозумілим, що порівняння окремих характеристик вже не дозволяє зробити коректний вибір. Наприклад, коли мова йде про антивірусне програмне забезпечення, вже не можна просто орієнтуватися на розмір бази вірусів, оскільки їхня кількість сягає мільярда одиниць, і навіть найпотужнішому комп'ютеру не вистачить, щоб постійно ретельно перевіряти на збіг кожен виконавчий файл. Аналогічні виклики виникають із вибором апаратного забезпечення. Сучасні апаратні файрволи більше не обмежуються простим аналізом трафіку за заголовками пакетів та відкиданням небажаних адрес або портів. Вони повинні фільтрувати складний за структурою трафік, у тому числі зашифрований, і забезпечувати надійний захист, одночасно забезпечуючи високу продуктивність для задоволення потреб користувачів. Також варто зазначити, що вибір апаратного забезпечення має враховувати архітектуру мережі, масштабування та планування майбутніх потреб. Оскільки зробити відповідний вибір програмного та апаратного забезпечення кібербезпеки стає все складніше, автори статті пропонують автоматизувати вибір з використання досягнень в сучасній теорії прийняття рішень. Наведені алгоритми вибору інструментів, орієнтовані на персональні потреби та на думки експертної групи. Продемонстрована ефективність вказаних алгоритмів для розв'язку реальних задач вибору в галузі за допомогою спеціально розробленого авторами програмного забезпечення для автоматизованого вибору складових програмного та апаратного забезпечення системи кібербезпеки корпоративних комп'ютерів.

**Ключові слова:** вибір антивіруса, важливість альтернатив, вибір файрвола, експертний метод, програмне забезпечення, кібербезпека.

**Вступ.** Вибір складових програмного та апаратного забезпечення системи кібербезпеки корпоративних комп'ютерів є відповідальною й одночасно достатньо творчою задачею, кількість можливих розв'язків якої дуже велика. Забезпечення кібербезпеки включає широкий спектр компонентів, які мають бути відповідним чином вибрані та налаштовані для забезпечення захисту інформації в мережах корпоративних комп'ютерів. Складовими програмного забезпечення систем кібербезпеки в першу чергу є антивірусні та антишпигунські програми, також це системи виявлення вторгнень, програмні файрволи, парольні менеджери, криптографічне програмне забезпечення. Складовими апаратного забезпечення систем в першу чергу є апаратні файрволи, а також пристрої ідентифікації та фізичні засоби збереження даних. Тенденція розвитку апаратного й програмного забезпечення

полягає в постійному ускладненні інтелектуальних алгоритмів, тому порівняння за окремими характеристиками втрачає сенс й тільки вводить в оману.

Так, наприклад, задача функціонування сучасного антивірусу – це вже давно не задача пошуку фрагментів у виконавчих файлах. Експерти стверджують, що кількість створених вірусів сягнуло відмітки одного мільярда одиниць – таку перевірку кожного виконавчого файлу не витримає найпотужніший комп'ютер [1]. Виходячи з цього такий відомий критерій як розмір бази вірусів давно вже перестав бути показником. Сучасний антивірус в ідеалі має виявити шкідливу програму за її поведінкою, однак чим більше ми орієнтуємось на поведінку, а не на відомі фрагменти коду, тим більшим стає ризик виникнення хибних спрацьовувань антивірусу при перевірці системного програмного забезпечення, програм зі спеціальними алгоритмами захисту від піратства тощо. Чим складніший алгоритм антивірусу, тим більше він вимагає ресурсів, теж саме можливо сказати й про недосконалий алгоритм [2]. Відсутність простих критеріїв для порівняння антивірусів компенсується можливістю переходу до порівняння за результатами синтетичних тестів. Кількість відомих антивірусних програм становить більше ніж 200, звісно, серед них можна знайти узагальнено якісно виконані й узагальнено неякісно виконані антивіруси. Однак, власне кажучи, серед найкращих на ринку антивірусів абсолютно найкращий знайти складно, кожен з них відповідає певному критерію ефективності.

Аналогічна проблема виникає й з апаратним програмним забезпеченням. Сучасний апаратний файрвол – це не проста система, яка аналізує трафік за заголовками пакетів й відкидає такі, які належать до переліку певних адрес чи портів. Напроти, це звичайно дуже коштовна система, яка має фільтрувати зміст складного за структурою трафіку. Класичний параметр – кількість з'єднань, за якими визначають лінійки апаратних файрволів з ускладненням програмних алгоритмів, з впровадженням шифрування й мультимедіа трафіку, вже втратив себе як надійний показник для порівняння. Фактично продуктивність файрвола залежить від типу трафіку, а значить якість роботи з ним залежить як від архітектури мікропроцесора, обсягу пам'яті, так і від складності алгоритмів. Синтетичні тести можуть також створити перевірку продуктивності за певними узагальненими типами трафіку, однак, як і в випадку антивірусів, серед топових продуктів не можна знайти певного абсолютного лідера для всіх задач.

Виходячи зі сказаного, необхідно переходити до застосування автоматизованих інструментів вибору, які дозволяють обробити результати синтетичних тестів й знайти оптимум для конкретних потреб спеціалістів. Саме розробці таких інструментів й присвячена ця стаття.

**Розв'язок задачі вибору найкращого антивірусу.** Задача сучасного антивірусу – визначення та блокування активності різноманітного, за метою своєї розробки, способом поширення та принциповими алгоритмами програмного забезпечення [1-7].

Антивірус має виявити та блокувати активність наступних типів програмного забезпечення .

Класичні віруси – це тип шкідливого ПЗ яке вбудовується в інші виконувані файли або програми. Віруси можуть саморозповсюджуватися шляхом впровадження свого коду в інші файли та інфікувати інші системи.

Шкідливе програмне забезпечення можна поділити на наступні категорії [3]:

Вимагацьке ПЗ – намагається вимагати гроші у жертв, блокуючи їхні файли або пристрої, доки вони не заплатять кіберзловмиснику суму викупу.

Шпигунське ПЗ – дозволяє зловмисникам віддалено відстежувати активність на зараженому комп'ютері без відома або згоди користувача.

Трояни – замасковане під законне програмне забезпечення, які надають віддалений доступ до системи користувача для зловмисних цілей, таких як крадіжка конфіденційної інформації.

Хробаки – самовідтворюваний код, призначений для поширення мережами і спричинення збоїв; можуть використовуватися для різних шкідливих дій, таких як видалення файлів, поширення інших шкідливих програм або навіть захоплення контролю над комп'ютерами.

Рекламне ПЗ – принципово так ПЗ може бути не шкідливим, але постійно спричиняє незручності користувачу.

Перша проблема вибору антивірусів – це проблема вибору критеріїв для оцінки. Ця проблема не є тривіальною. Так, наприклад, в роботі [5] проведено аналіз різних критеріїв та зроблено висновок, що задача вибору має ставитись як принципово багатокритеріальна. Певні орієнтовні критерії вибору це: функціональні можливості, ефективність виявлення і блокування загроз, вплив на продуктивність операційної системи, сумісність, вартість, юзабіліті. В роботі [5] запропоновано класифікувати всі критерії у дві великі категорії: безпека та операційність.

Друга проблема вибору антивірусів – це проблема вибору методу багатокритеріальної оцінки. Відзначимо роботу [7], в якій запропоновано застосовувати метод діаграм Рея для вибору найкращого антивірусу. Недоліками роботи є те, що порівняння виконано лише для чотирьох антивірусів та й критерії, за якими виконано порівняння не зовсім обґрунтовані.

Розв’язок першої проблеми став можливим з впровадженням технології проведення синтетичних тестів, аналогічно до тестів, які проводяться щодо апаратного забезпечення комп’ютерів. Дослідження, що проводились з початку століття [8], дозволили створити технологію тестування антивірусів, полягає в оцінці за спеціальними узагальненими критеріями. На базі застосування цієї технології була сформована незалежна організація Av-Comparatives [9]. AV-Comparatives розробляє сценарії, що відтворюють реальні умови використання антивірусів, а також застосовує тестові файли та шкідливі програми для оцінки рівня виявлення і блокування. Організація також проводить тестування продуктивності, перевіряючи вплив антивірусного програмного забезпечення на швидкодію системи. AV-Comparatives отримала визнання у сфері безпеки і премії і відзнаки за свою роботу. Зокрема це премія "За видатні досягнення в галузі тестування безпеки" на конференції EICAR (European Institute for Computer Antivirus Research) й відзнака AV-Comparatives як лабораторії ІТ-тестування, яка заслуговує на довіру на цій же конференції.

Для антивірусів під ОС Windows компанія застосовує наступні критерії: Advanced Threat Protection (ATP), Mailware Protection (MWP), Perfomance, Real World Protection, False Alarms.

Advanced Threat Protection (ATP). Оцінює якість інтелектуального алгоритму для виявлення потенційно шкідливої активності програмного забезпечення, що працює на комп’ютері, зважаючи на якість виявлення вірусів, які не відомі програмі.

Malware Protection (MWP). Оцінює якість захисту користувача від запуску шкідливих програм, які розміщуються на локальних, мережевих та хмарних сховищах.

Perfomance. Оцінює швидкість роботи антивірусу. Критерій визначається експериментально за часом перевірки значної за обсягом бази.

Real World Protection (RWP). Оцінює ймовірність зараження вірусом через браузер при відвідуванні шкідливих сайтів.

Результати оцінювання критеріїв, отримані в результаті тестів, нормуються від 0 (найгірше значення) до 3 (найкраще значення).

Як основні ненормовані показники якості антивірусів застосовані такі показники.

False Alarms (FA). Оцінює не тільки кількість правильних спрацювань, але й кількість хибних. Визначається за великою тестовою базою даних. Кількісні показники адекватні для порівняння в межах тільки використаної БД вірусів.

Price. Оцінює вартість програми. Чим вища величина параметра, тим менш ймовірне зараження вірусом через браузер при відвідуванні шкідливих сайтів.

Оцінювання критеріїв якості для вибору антивірусів десктопної версії ОС Windows за результатами тестування у 2023 році приведені в табл. 1.

**Таблиця 1**

Результати оцінювання критеріїв якості для вибору антивірусів [9]

| **Антивірус** | **ATP** | **MWP** | **Perfomance** | **RWP** | **ATP** | **FA** | **Ціна за рік** |
|---|---|---|---|---|---|---|---|
| Bitdefender | 3 | 3 | 3 | 3 | 3 | 6 | 50 |
| Avast | 2 | 3 | 3 | 3 | 2 | 2 | 0 |
| AVG | 2 | 3 | 3 | 3 | 2 | 2 | 0 |
| G Data | 2 | 3 | 3 | 3 | 2 | 2 | 50 |
| Avira | 0 | 3 | 3 | 3 | 0 | 1 | 100 |
| McAfee | 0 | 3 | 3 | 3 | 0 | 9 | 86 |
| ESET | 3 | 2 | 3 | 1 | 3 | 0 | 35 |
| VIPRE | 0 | 3 | 2 | 3 | 0 | 6 | 40 |
| NornonLifeLock | 0 | 3 | 3 | 2 | 0 | 3 | 100 |
| Microsoft | 2 | 2 | 1 | 3 | 2 | 32 | 0 |
| K7 | 0 | 1 | 3 | 3 | 0 | 67 | 26 |
| Total Defense | 0 | 3 | 1 | 2 | 0 | 6 | 57 |
| TotalAV | 0 | 3 | 2 | 2 | 0 | 0 | 119 |
| Panda | 0 | 0 | 3 | 2 | 0 | 102 | 0 |
| Trend Micro | 0 | 0 | 3 | 1 | 0 | 10 | 50 |
| Mailwarebytes | 0 | 0 | 3 | 0 | 0 | 25 | 40 |

Ціна за рік не є результатом тесту, а мається на увазі вартість річної підписки для персонального використання за рік в доларах США.

Далі розв’яжемо другу проблему, що дозволить знайти комплексний розв’язок задачі. Пропонуємо застосовувати метод прийняття рішень з урахуванням важливості альтернатив [10]. Цей метод дозволяє користувачеві задати свої переваги між кожною парою альтернатив. Кількість РС альтернатив формується виходячи з формули перестановок $PC = 0.5 \cdot C!/(C\text{-}2)!$, де $C$ – кількість критеріїв вибору. Аналізуючи вихідні дані (табл. 1) треба зробити висновок, що критерії вибору дещо відрізняються за характером оцінки. Так, для перших чотирьох критеріїв порівняння в межах критерію однозначно просто – чим більше, тим краще. Для критеріїв FA і Price – чим більше, тим гірше й результати мають бути безумовно масштабовані.

Система автоматизованого вибору антивірусу реалізується як програма з графічним інтерфейсом, яка забезпечує можливість зручної заміни величин результатів синтетичних тестів, які регулярно оновлюються Av-Comparatives. Вибір пріоритетів реалізований в вигляді повзунків, які дозволяють графічно й інтуїтивно зрозуміло до користувача відобразити обраний пріоритет. Користувач має чітко визначитись з кожним пріоритетом. Якщо будь-який параметр йому не потрібний, то користувач має можливість його просто відімкнути й тоді кількість альтернатив, які необхідно обрати, зменшується. Також введено додатковий фільтр, який дозволяє, за

необхідності, чітко відокремити безплатні антивіруси від таких, які вимагають придбання комерційної ліцензії. Основні результати відображуються за допомогою гістограми, нормованої до 100% шкали. Також, для користувача графічно зображено фактичні ваги пріоритетів в вигляді кругової діаграми. Для реалізації програми обрано мову системи MATLAB, яка орієнтована на математичні розрахунки. Алгоритм вибору, закладений в програмі, застосовує матриці – саме на цей тип даних орієнтована мова. Також застосовано графічне середовище App Designer для швидкої розробки програм з графічним інтерфейсів, яке має графічний конструктор форм та розвинуті засоби візуалізації даних. Вікно екрану розробленої програми для введення відносної важливості критеріїв якості показано на рис. 1.

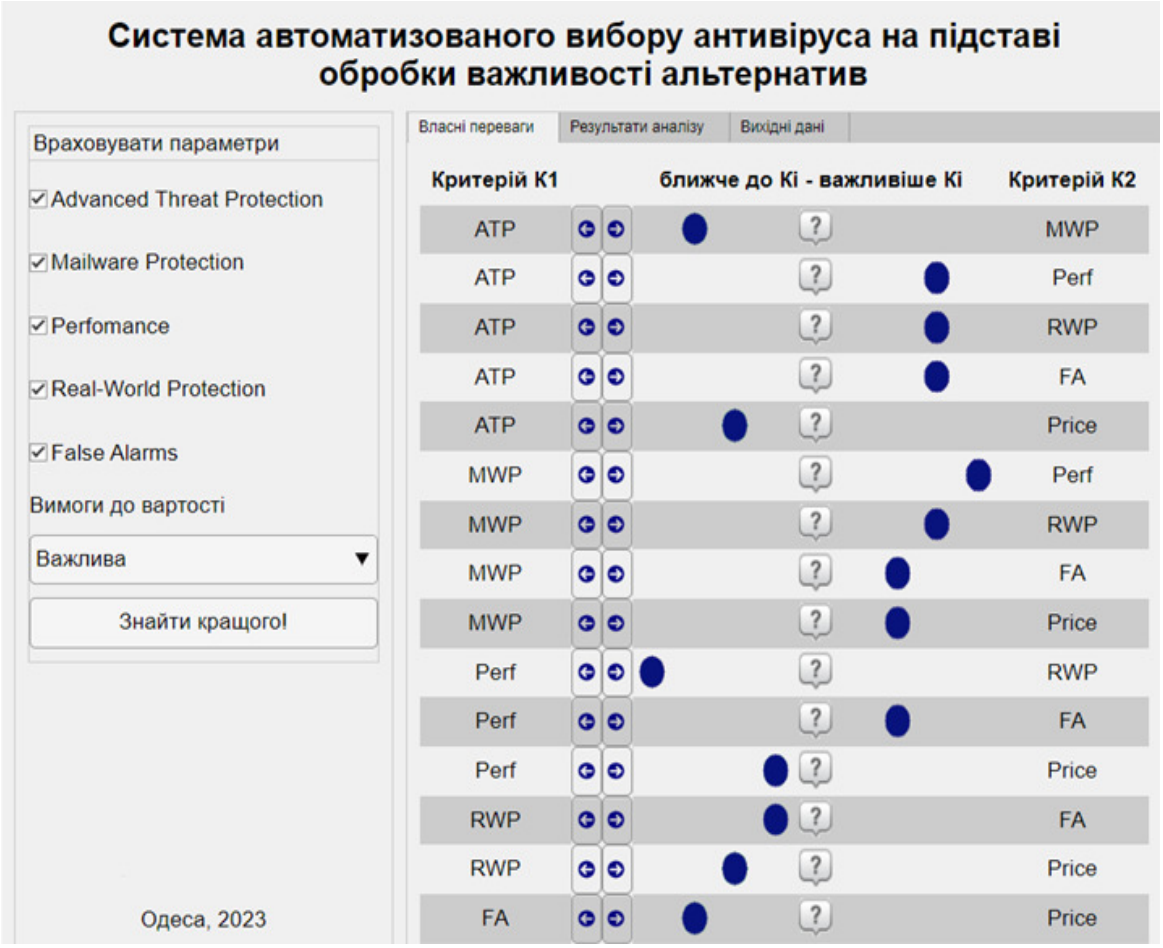



**Рис.1.** Вікно екрану програми для введення відносної важливості критеріїв якості

Вікно екрану розробленої програми з результатами розрахунку якостей антивірусів приведено на рис. 2.

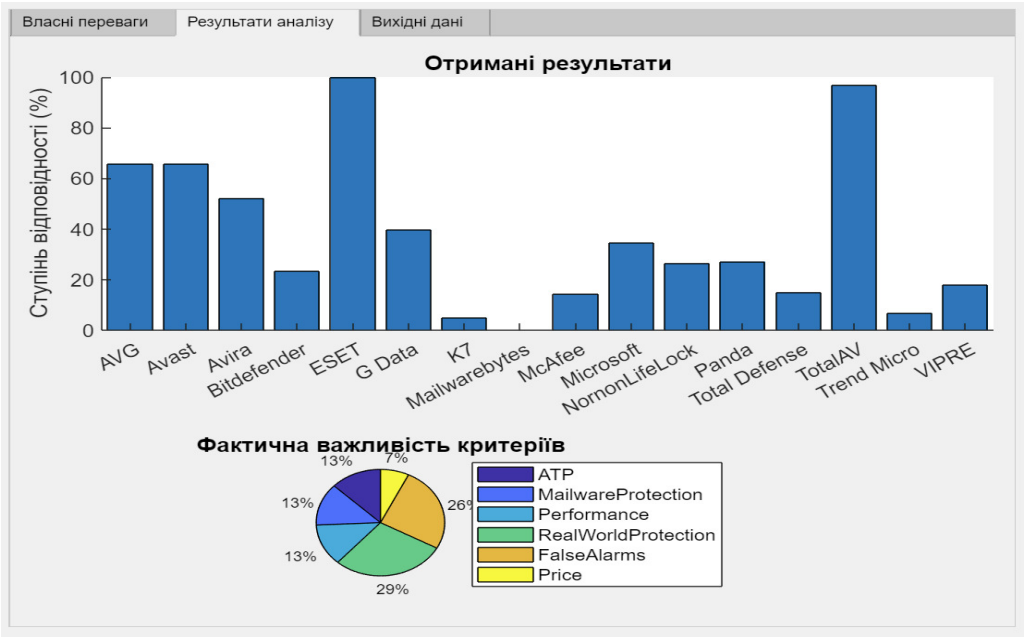



**Рис.2.** Вікно екрану програми з результатами розрахунку якостей антивірусів

Таким чином, бачимо що програма розв'язує поставлену задачу й є зручною для користування.

**Розв'язок задачі вибору найкращого апаратного файрвола.** Як і попередня задача, ця задача має дві проблеми.

Перша проблема – це проблема критерію вибору. Класичним основним критерієм вибору для апаратних файрволів є кількість можливих підключень. Виходячи з цього критерію, апаратні файрволи випускаються в межах лінійок, градація в яких проводиться по оціненій виробниками кількості можливих підключень. Файрволи для малого бізнесу призначені для обслуговування 50-200 користувачів; для середнього бізнесу – від 200 до 500 користувачів; для великого бізнесу – від 500 до 3000; для датацентрів - 3000 й більше. Однак, вибір за цим критерієм не має сенсу автоматизувати, він однозначно визначається масштабом підприємства. Після визначення лінійки файрволів основним стає критерій продуктивності обробки трафіку. Класифікація видів трафіку, який обробляє файрвол, показана на рис. 3.


**Рис. 3**. Класифікація трафіку, який обробляє файрвол

Таким чином, бачимо, що трафік має дві схеми класифікації, тому ми можемо виділити 6 типів трафіку з точки зору роботи файрвола. Зазвичай продуктивність по простому й мультимедіа трафіку відрізняється в 1.5-4 рази в залежності від типу пристрою. Продуктивність по нефільтрованому трафіку близька до продуктивності портів й мережевих пристроїв файрволу. Продуктивність по фільтрованому трафіку є меншою, кратність продуктивності фільтрованого до нефільтрованого трафіку залежить від апаратних та програмних можливостей пристрою, й становить звичайно від 0.3 до 0.9 раз. Продуктивність по трафіку протоколів зв'язку IPSec. В першу чергу це VPN. Продуктивність по IPSec трафіку є ще меншою, кратність продуктивності фільтрованого до нефільтрованого трафіку дуже залежить від апаратних та програмних можливостей пристрою (важливим є математичний сопроцесор, якість, ефективність та масштабованість криптографічних алгоритмів й відповідно програмного забезпечення), й становить звичайно від 0.1 до 0.8 раз до нефільтрованого трафіку.

Позначимо введені критерії якості файрволів наступним чином:

–FWD-B - максимальна здатність перенаправлення простого трафіку (Мбіт/с);

–FW-B - максимальна здатність фільтрації простого трафіку (Мбіт/с);

– IPS-B - максимальна здатність фільтрації простого VPN трафіку за технологією IpSEC (Мбіт/с);

–FWD-C - максимальна здатність перенаправлення мультимедіа (та іншого складного) трафіку (Мбіт/с);

–FW-C - максимальна здатність фільтрації мультимедіа (та іншого складного) трафіку (Мбіт/с);

–IPS-C - максимальна здатність фільтрації мультимедіа (та іншого складного) VPN трафіку за технологією IpSEC (Мбіт/с).

Також додамо критерій не пов'язаний з трафіком – CPU, тобто тактову частоту процесора файрвола (ГГц).

Вибір файрвола будемо проводити серед лінійки апаратних файрволів, яка відповідає потребам середнього бізнесу Відносні значення продуктивності 10 обраних файрволів різних виробників з близькою загальною продуктивністю приведені на рис. 4.

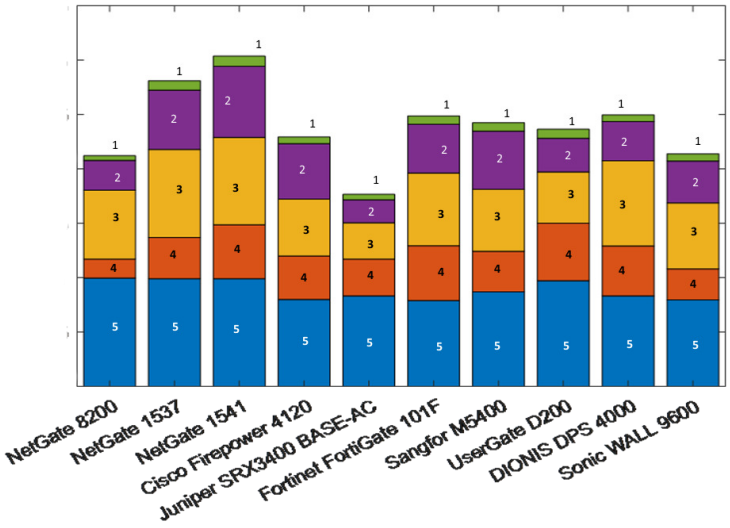



**Рис.4**. Відносні значення продуктивності файрволів

На рис. 4 висота стовпця характеризує величину FWD-B. Висота підстовпців характеризує нормоване відношення: 1. IPS-C/FWD-B; 2. FW-C/FWD-B; 3. FWD-C/FWD-B; 4.IPS-B/FWD-B; 5. FW-B/FWD-B).

Друга проблема – проблема методу вибору. Аналізуючи рис. 4, бачимо, що розподіл характеристик доволі нерівномірний. Таким чином, для підприємств з різним характером трафіку оптимальним буде вибір різних файрволів. На відміну від задачі вибору антивірусу, задача вибору файрволу для підприємства середнього бізнесу є задачею колективного вибору, яка має базуватись на розумінні специфіки роботи різних підрозділів підприємства. Характер трафіку в підприємстві (поточний й прогнозований) можуть знати експерти – системні адміністратори, мережевики. Отже, вибір має орієнтуватись на їх думки. Однак, щоб виключити суб'єктивність, думки експертів мають надаватись анонімно й оброблятись за спеціальним алгоритмом [10]. При цьому експертна група має формуватися таким чином, щоб думки експертів були узгодженими. Узгодженість визначається коефіцієнтом конкордації, який повинен бути не меншим за 0.7.

Система автоматизованого вибору апаратного файрвола реалізується як програма з графічним інтерфейсом, яка забезпечує можливість швидкого введення та модифікації бази параметрів пристроїв та думок експертів. Діалогове вікно програми має вигляд, приведений на рис. 5. Програму реалізовано як класичну десктоп-програму. Для цієї задачі також підходить MATLAB та App Designer. Зміна елементів таблиць думок експертів та технічних характеристик проводиться аналогічно редакторам електронних таблиць. Кожен рядок таблиць має опцію відмічення. Відмічені рядки враховуються при виборі найкращої моделі, не відмічені – не враховуються. Додавання елементів реалізовано шляхом заповнення спеціальних діалогів, введення параметрів в яких перевіряється. Видалення проводиться для елементів, які не відмічені. При зміні елементів таблиць перевіряються вхідні дані, а зміна думок експертів проводиться шляхом вибору з можливих варіантів оцінки. При зміні думок оцінки перераховуються та відображається новий коефіцієнт узгодженості. Значення таблиць зберігаються у текстових csv файлах. Таким чином, зміна кожного параметру зберігається автоматично.

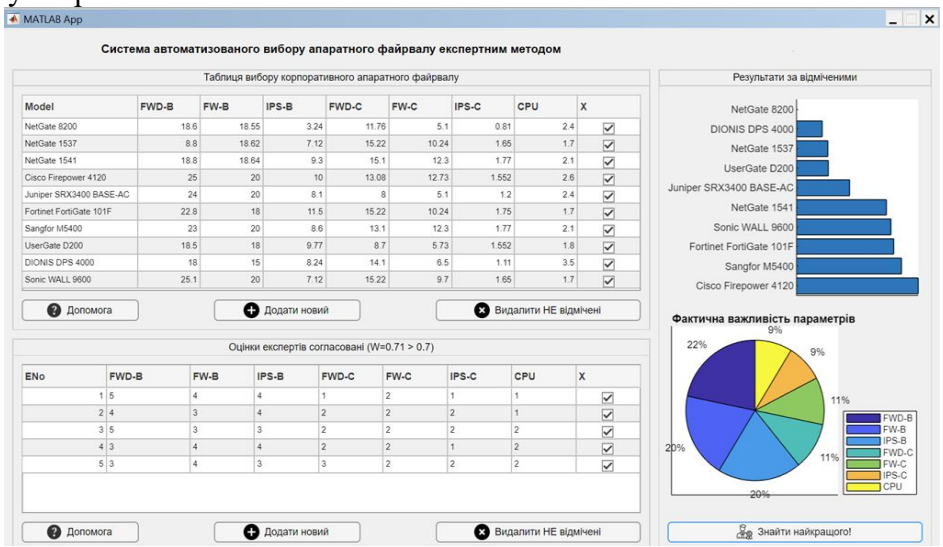



**Рис.5.** Діалогове вікно програми вибору файрвола

Розрахунок має проводитись лише при відповідному коефіцієнті узгодженості (конкордації) та наявності коректних вхідних даних. Основні результати розрахунку

представляються в вигляді двох графіків. Перший графік – гістограма, нормована до 100% шкали з відсортованими результатами, яка показує «ефективність моделі». Другий графік виконаний в вигляді кругової діаграми, в якій зображуються пріоритети параметрів.

**Висновки**. Розроблені системи автоматизованого вибору складових програмного та апаратного забезпечення системи кібербезпеки корпоративних комп'ютерів.

Запропоновано замість класичних критеріїв порівняння антивірусів застосовувати результати синтетичних тестів. Вибір проводиться на базі суб'єктивних переваг користувача для кожного з обраних критеріїв. Об'єктивні значення критеріїв беруться з результатів періодичних тестів AV-Comparatives.

Запропоновано замість класичного критерію порівняння апаратних файрволів застосовувати 7 критеріїв, 6 з яких належать до продуктивності апаратного файрвола за різними типами трафіку. Вибір проводиться на базі обробки думок експертної групи, яка має узгодженість. Значення критеріїв беруться з документації файрволів або результатів незалежних тестів.

Розроблені системи автоматизованого вибору антивірусу та файрвола як програми з графічним інтерфейсом. Продемонстровані функціональні можливості програм показують, що вони зручно та ефективно виконують усі поставлені задачі.

Розроблені системи автоматизованого вибору рекомендуються для застосування спеціалістами з кібербезпеки та системними адміністраторами.

**Список літератури**


1 Jovanovic B.A. Not-So-Common Cold: Malware Statistics in 2023. URL: https://dataprot.net/statistics/malware-statistics/

2 Fahad A. The Evolution of Antivirus Software to Face Modern Threats in 2023. URL: https://securityintelligence.com/posts/antivirus-evolution-to-face-modern-threats/

3 Alenezi M.N., Alabdulrazzaq H.K., Alshaher A.A., Alkharang M.M. Evolution of Malware Threats and Techniques: a Review. *International Journal of Communication Networks and Information Security (IJCNIS)*, 2020. Vol. 12. No.3. P.326-337. URL: https://doi.org/10.17762/ijcnis.v12i3.4723

4 Leka C., Ntantogian C., Karagiannis S., Magkos E., Verykios V. A Comparative Analysis of VirusTotal and Desktop Antivirus Detection Capabilities. *13th International Conference on Information, Intelligence, Systems & Applications (IISA)*, 2022. P.1-6. URL: https://doi.org/10.1109/IISA56318.2022.9904382.

5 Alharbi B., Asseri A., Alzahrani H., Taramisi K. Anti-Malware Efficiency Evaluation Framework. Anti-Malware Efficiency Evaluation Framework. *2nd International Conference on Computer and Information Sciences (ICCIS),* 2020. P. 1-4. URL: http://doi.org/10.1109/ICCIS49240.2020.9257637

6 Al-Saleh M., Hamdan H. Precise Performance Characterization of Antivirus on the File System Operations. *Journal of Universal Computer Science,* 2019. Vol. 25. P.1089-1108. URL: https://doi.org/10.3217/jucs-025-09-1089

7 Gorshkov A.V., Lokhvitskii V.A., Khomonenko A.D., Rybakova E.A., Gorshkov V.N. Multi-criteria choice of Antivirus tools with using the Ray diagrams. *Intellectual technologies on transport*, 2016. No. 2. P. 23-29.

8 Marx A., Rautenstrauch C. Systematisches Testen von Anti-Viren-Software. Wirtschaftsinf 45, 435–443 (2003). https://doi.org/10.1007/BF03250908

9 AV Comparatives. URL: https://www.av-comparatives.org/

10 Стопакевич О.А. Теорія прийняття рішень: конспект лекцій. Одеса: НУОП, 2021.

А.В.Князєв, Р. І.Назаренко, О.А.Стопакевич, А.О.Стопакевич

# AUTOMATED SOFTWARE AND HARDWARE SELECTION SYSTEMS FOR ENTERPRISE COMPUTERS' CYBERSECURITY

A.V.Knyazev, R.I.Nazarenko, O.A.Stopakevych, A.O.Stopakevych

National Odesa Polytechnic University, Shevchenko str., 1, Odesa, 65044, Ukraine
stopakevich@gmail.com

Providing effective cybersecurity for corporate computers is an extremely important and complex task that requires careful choice of software and hardware. This article discusses the key components of cybersecurity software and hardware - antivirus programs and hardware firewalls. Considering the rapid growth of cyber threats, it becomes clear that comparing individual characteristics no longer allows users to make the right choice. For example, when it comes to antivirus software, it is no longer possible to simply focus on the size of the virus database, as the number of viruses reaches a billion, and even the most powerful computer is not enough to scan every executable file for a match. Similar challenges arise with the choice of hardware. Modern hardware firewalls are no longer limited to simply analyzing traffic by packet headers and dropping unwanted addresses or ports. They must filter complex traffic, including encrypted traffic, and provide reliable protection while delivering high performance to meet user needs. It is also worth noting that the choice of hardware should take into account the network architecture, scalability, and planning for future needs. Since it is becoming increasingly difficult to make the appropriate choice of cybersecurity software and hardware, the authors of the article propose to automate the choice using advances in modern decision theory. The authors present algorithms for choosing tools based on personal needs and the opinions of an expert group. The authors demonstrate the effectiveness of these algorithms for solving real-world selection problems in the industry with the help of software specially developed by the authors for automated selection of software and hardware components of the corporate computer cybersecurity system.

**Keywords:** choice of antivirus, importance of alternatives, choice of firewall, expert method, software, cybersecurity

DOI 10.15276/imms.v13.no1-2.97
УДК 519.972.5:004.942.001.57

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp. 97-103

# РОЗРОБКА ТА ЧИСЛОВА РЕАЛІЗАЦІЯ МАТЕМАТИЧНОЇ МОДЕЛІ ГРАВІТАЦІЙНОЇ ХВИЛІ НА ГРАНИЦІ ПОДІЛУ ДВОШАРОВОЇ РІДИННОЇ СИСТЕМИ

Д. А. Лись, А. Ю. Прокофьєв

Національний університет «Одеська політехніка»,
Проспект Шевченка, 1, Одеса, 65044, Україна;E-mail:
loreal338@gmail.com, fallbrick1985@gmail.com

Хвильова динаміка двофазних систем — новий розділ механіки гетерогенних систем та теплофізики, який стрімко розвивається у останні роки. Дослідження у цій галузі потребують залучення сучасних результатів нелінійної хвильової динаміки, розробки нових способів врахування міжфазної взаємодії дослідження сучасних концепцій хвильових рухів (плинів), таких як «кінематичні хвилі», «динамічні хвилі» та «багато хвильові» системи. Складність фізичної постановки зазначених задач потребує відповідних адекватних математичних моделей (ММ) цих процесів, а також конструктивних методів числової реалізації створених ММ. Крім того, актуальність таких досліджень зумовлена тим, що двофазні потоки у переважній більшості виникають у робочих режимах в технологічних установках енергетичної, хімічної, металургійної та інших важливих галузях народного господарства. Робочі процеси в нафтодобувній та нафтопереробній промисловостях, в апаратах кріогенної техніки супроводжуються утворенням особливого типу двофазних систем — пароріднинних сумішей. Відомо, що більшість двофазних систем характеризуються властивістю значного стискання (тобто швидкість звуку в такій системі мала), не лінійністю і тому для розрахунку динаміки таких середовищ, які рухаються з відносно невисокими швидкостями, необхідно застосування особливих газодинамічних методів. Також рух двофазних систем супроводжується процесами між фазного теплообміну , які спричиняють сильну дисипацію середовища, а інерційні властивості газових включень породжують залежність швидкості звуку від частоти — дисперсію швидкості звуку. Тому методи, та пов’язані з ними ММ традиційної пароріднинної динаміки не відповідають специфіці двофазних потоків та дають незадовільні результати при розрахунках. Іншими словами, ці ММ та методи їх числової реалізації не є адекватними складним фізичним досліджуваним процесам. В чинній роботі поставлено задачу побудови конструктивних ММ гравітаційних хвиль, які утворюються на границі поділу фаз двошарової рідинної системи. Запропоновані ММ якісно відображають суть фізики динамічних процесів у двофазній рідинній системі і являють собою модельні канонічні рівняння, сформульовані в умовах прийнятих припущень щодо перебігу досліджуваних процесів. Проведені числові дослідження показали, що коректна формалізація особливостей фізичних явищ в рамках запропонованих моделей, дозволяє при постановці реальних прикладних експериментів з достатньої для інженерної практики розкрити природу закономірностей гідрогазодинамічних плинів.

**Ключові слова**: двофазна система; гравітаційна хвиля; двошарова рідинна система, математична модель; числовий експеримент.

**Вступ.** На практиці досить поширеним є випадок, коли рідинна система являє собою суміш рідин з відмінними фізико-хімічними властивостями, зокрема: з різними густинами, з різними температурами кипіння, з наявністю емульсованих домішок, з несхожою в’язкою (реологічною) поведінкою тощо. Типовими прикладами сумішей таких рідин можуть бути:

– природні рідкі вуглеводні, які складаються з фракцій з різними густинами та температурами кипіння;

– водо-оливні емульсії, які можуть, при зміні динамічного стану (швидкості плину), розділятися на рідини, що не змішуються та мають різні густини;

– колоїдні та полімерні розчини, складові яких мають різну реологічну поведінку, тобто виявляють характер «ньютонівської» (такої, що не стискається, чи, інакше, такої, для якої густина не залежить від швидкості плину) або «ненютонівської» (такої, що стискається, чи, інакше, такої, для якої густина залежить від швидкості плину) рідини.

При цьому зазначимо, що окремим явищем динаміки парорідинної двофазної системи слід розглядати наявність у її рідинній складові двох шарів рідин з різними фізико-хімічними властивостями. Специфіка плину таких парорідинних двофазних систем полягає [1, 2] у наявності вільної границі, на якій може відбуватися розвиток нестійкості та хвилеутворення, а також процеси теплообміну та тертя з паровою фазою.

**Мета роботи**. Мета роботи полягає у розробці конструктивних математичних моделей (ММ) гравітаційних хвиль, що утворюються на границі поділу окремих фаз двошарової рідинної системи та доведення адекватності даних ММ шляхом числового дослідження.

**Основна частина.** Розглянемо розповсюдження гравітаційних хвиль на границі поділу двох шарів рідин різної густини ( $\rho_1 < \rho_2$ ), які не змішуються, за умов для двофазної рідинної системи: обмеження знизу горизонтальним дном, а з верху — наявності вільної границі. Уявімо, що хвилі є *довгими* та мають *малу* амплітуду. Відзначимо, що задача, яка розглядається, не тільки має важливі технічні за стосунки, але і є найпростішою моделлю океану, що враховує стратифікацію. Тобто можна говорити про технічну та загально природну *актуальність* задачі, яка розглядається. Зазначимо, що така двошарова модель містить дві моди коливань — *бароклинну* та *баротропну*. Баротропна мода — швидка, яка відповідає коливанням системи як цілого або синхронним коливанням двох шарів; бароклинна мода — повільна, та відповідає коливанням шарів, зсунутих по фазі на $\pi$. Певний досвід із розв'язання подібних задач описано в роботах [3, 4].

Запишемо рівняння нерозривності та $x$-компоненти рівнянь руху для всієї системи загалом і окремо для нижнього шару:

$$\frac{\partial\left(\langle\rho\rangle\delta\right)}{\partial t}+\frac{\partial\left(\langle u\rangle\langle\rho\rangle\delta\right)}{\partial x}=0;\ \frac{\partial\delta_2}{\partial t}+\frac{\partial\left(\langle u_2\rangle\right)\delta_2}{\partial x}=0; \tag{1}$$

$$\left.\begin{aligned}&\frac{\partial\left(\langle u\rangle\langle\rho\rangle\delta\right)}{\partial t}+\rho_1\frac{\partial\left(\langle u_1^2\rangle\,\delta_1\right)}{\partial x}+\rho_2\frac{\partial\left(\langle u_2^2\rangle\,\delta_2\right)}{\partial x}+\frac{\partial\left(\langle P\rangle\,\delta\right)}{\partial x}=0;\\&\frac{\partial\left(\langle u_2\rangle\,\delta_2\right)}{\partial t}+\frac{\partial\left(\langle u_2^2\rangle\,\delta_2\right)}{\partial x}+\frac{1}{\rho_2}\left[\frac{\partial\left(\langle P_2\rangle\,\delta_2\right)}{\partial x}-P_{\text{г}}\frac{\partial\delta_2}{\partial x}\right]=0.\end{aligned}\right\} \tag{2}$$

тут $\delta$ — загальна глибина двох шарів рідини; $P_{\text{г}}$ — тиск на границі двох рідин; $u$ — середня швидкість хвилі для двофазної рідинної системи; $P$ — середнє значення тиску для двофазної рідинної системи; індексом 1 відмічено величини, які відносяться до верхньої рідини, а індексом 2 — до нижньої. В'язкості рідин не враховуються.

Після диференціювання рівнянь (1) за часом, а рівнянь (2) — по координаті $x$, їх різниця запишеться наступним чином:

$$\begin{aligned}
&\frac{\partial^2\left(\langle\rho\rangle\delta\right)}{\partial t^2}-\rho_1\frac{\partial^2\left(\left\langle u_1^2\right\rangle\delta_1\right)}{\partial x^2}-\rho_2\frac{\partial^2\left(\left\langle u_2^2\right\rangle\delta_2\right)}{\partial x^2}-\frac{\partial^2\left(\langle P\rangle\delta_2\right)}{\partial x^2}=0;\\
&\frac{\partial^2\delta_2}{\partial t^2}-\frac{\partial^2\left(\left\langle u_2^2\right\rangle\delta_2\right)}{\partial x^2}-\frac{1}{\rho_2}\left[\frac{\partial^2\left(\langle P_2\rangle\delta_2\right)}{\partial x^2}-\frac{\partial}{\partial x}\left(P_{\text{г}}\frac{\partial\delta_2}{\partial x}\right)\right]=0.
\end{aligned}\tag{3}$$

Наближення довгих хвиль дозволяє вважати, що профіль горизонтальної складової швидкості рідини є «заповненим», тобто не залежить від координати $y$, а профіль вертикальної складової — лінійним:

$$v_1=\frac{y-\delta_2}{\delta_1}\left(\frac{\partial\delta}{\partial t}-\frac{\partial\delta_2}{\partial t}\right)+\frac{\partial\delta_2}{\partial t};\ \ v_2=\frac{y}{\delta_2}\frac{\partial\delta_2}{\partial t}.$$

Підстановка цих виразів в $y$-компоненти рівнянь руху верхньої та нижньої рідин $\frac{\partial v_1}{\partial t}+\frac{1}{\rho_1}\frac{\partial P_1}{\partial y}+g=0$; $\frac{\partial v_2}{\partial t}+\frac{1}{\rho_2}\frac{\partial P_2}{\partial y}+g=0$ разом з граничними умовами $P_1=0$ при $y=\delta$ та $P_1=P_2=P_{\text{г}}$ при $y=\delta_2$ дає можливість виразити середні значення тиску через глибини шарів.

Спочатку розглянемо «швидку» моду. Малість амплітуди збуджень дозволяє у квадратичних членах рівнянь (3) покласти з точністю до членів другого порядку малості $\left\langle u_1^2\right\rangle=\left[c_1^0\left(\delta'-\delta_2'\right)/\delta_1^0\right]$, $\left\langle u_2^2\right\rangle=\left(c_1^0\delta_2'/\delta_2^0\right)^2$, де $\delta'=\delta-\delta_0$, $\delta_0$ — рівно вагове значення глибини, а $c_1^0$ — граничне довгохвильове значення швидкості розповсюдження безкінечно малих збуджень швидкої моди:

$$\left(c_1^0\right)^2=g\delta^0\left[1+\sqrt{1-4\left(\delta_2^0/\delta^0\right)\left(1-\delta_2^0/\delta^0\right)\Delta\overline{\rho}}\right]/2,$$
$$\Delta\overline{\rho}=1-\overline{\rho};\ \overline{\rho}=\rho_1/\rho_2.$$

В результаті рівняння (3) можна перетворити до вигляду

$$\begin{aligned}
&\frac{\partial^2\delta}{\partial t^2}-g\frac{\partial^2}{\partial x^2}\left[\left(\delta^0-\Delta\overline{\rho}\delta_2^0\right)\delta+\Delta\overline{\rho}\delta_2^0\delta_2\right]-\left(c_1^0\right)^2\left[\frac{\left(\delta'-\delta_2'\right)^2}{\delta_1^0}+\frac{\left(\delta_2'\right)^2}{\delta_2^0}\right]-\\
&-g\frac{\partial}{\partial x}\left[\delta'\frac{\partial\delta}{\partial x}-\Delta\overline{\rho}\delta_2'\frac{\partial\left(\delta-\delta_2\right)}{\partial x}\right]-\left[\frac{\left(\delta_1^0\right)^2}{3}+\overline{\rho}\frac{\delta_1^0\delta_2^0}{2}\right]\times\\
&\times\frac{\partial^4\delta_2}{\partial t^2\,\partial x^2}\left\{1-\left[\frac{\left(\delta_1^0\right)^2}{6}+\frac{\left(\delta_1^0\right)^2}{3}+\overline{\rho}\frac{\delta_1^0\delta_2^0}{2}\right]\right\}=0;\\
&\frac{\partial^2\delta_2}{\partial t^2}-g\delta_2^0\frac{\partial^2}{\partial x^2}\left(\overline{\rho}\delta-\Delta\overline{\rho}\delta_2\right)-\left(c_1^0\right)^2\frac{\partial^2}{\partial t^2}\left[\frac{\left(\delta_2'\right)^2}{\delta_2^0}\right]-g\frac{\partial}{\partial x}\left[\delta_2'\frac{\partial}{\partial x}\left(\overline{\rho}\delta+\Delta\overline{\rho}\delta_2\right)\right]-\\
&-\overline{\rho}\frac{\delta_1^0\delta_2^0}{2}\frac{\partial^4\left(\delta+\delta_2\right)}{\partial t^2\,\partial x^2}-\frac{\left(\delta_2^0\right)^2}{3}\frac{\partial^4\delta_2}{\partial t^2\,\partial x^2}=0.
\end{aligned}\tag{4}$$

Таким чином, для швидкої моди систему рівнянь гідродинаміки вдалося звести до двох рівнянь хвильового типу, які враховують нелінійність збуджень та інерцію шарів рідини. Одначе, в силу складності рівнянь (4), віднайти аналітичний розв'язок даної системи не є можливим. В такій задачі виявляється плідним узагальнення «простих» та «квазіпростих» хвиль, викладене в роботах [5 — 7]. Традиційно поняття простих» та «квазіпростих» хвиль застосовується до систем рівнянь газової динаміки, що являють собою диференційні рівняння першого порядку. Рівняння (4) мають більш високий порядок. Одначе, дотримуючись ідеям Рімана та Карпмана [8], зробимо додаткові припущення, які полягають в тому, що між збудженнями вільної та внутрішньої границь існує наступний зв'язок:

$$\delta_2' = f_1(\delta') + f_1'(\delta'). \tag{5}$$

Тут $f_1(\delta')$ — розв'язок системи рівнянь (5.73) без нелінійних та дисперсійних членів, тобто

$$f_1(\delta') = f_1^0 \delta' = \left\{1 - \left[\delta^0 / \delta_2^0 - \left(c_1^0\right)^2 / \left(g\delta_2^0\right)\right]\Delta\bar{\rho}^{-1}\right\} \delta',$$

а функція $f_1'(\delta')$ має другий порядок малості, оскільки пропорційна нелінійним та дисперсійним членам. Для віднаходження функції $f_1'(\delta')$ необхідно підставити вираз (5) в рівняння (4), знехтувати членами третього порядку малості та вважати, що $\partial^2\left[f_1'(\delta')\right]/\partial t^2 \approx \left(c_1^0\right)^2 \partial^2\left[f_1'(\delta')\right]/\partial x^2$ . У підсумку перше з рівнянь (5.73) приймає вид модифікованого рівняння Буссінеска

$$\frac{\partial^2\delta}{\partial t^2} - \left(c_1^0\right)^2 \frac{\partial^2\delta}{\partial x^2} - \alpha_1 \frac{\partial^2(\delta')^2}{\partial x^2} - \beta_1 \frac{\partial^4\delta}{\partial t^2\,\partial x^2} = 0, \tag{6}$$

де коефіцієнти при нелінійному і дисперсійному членах є константами та визначаються формулами

$$\alpha_1 = g\left[f_1^0 + q_1\Delta\bar{\rho}\left(1 - f_1^0\right)\right]/2 + \left(c_1^0\right)^2\left[q_1/\delta_2^0 + r_1\left(1 - f_1^0\right)/\delta_1^0\right];$$

$$\beta_1 = q_1\left[\left(\delta_2^0\right)^2/3 + \bar{\rho}\delta_1^0\delta_2^0\left(1 + f_1^0\right)/2\right] + r_1\left\{\left(\delta_1^0\right)^2\left(1/2 + f_1^0\right)/\left[3\left(1 - f_1^0\right)\right]\right\};$$

$$q_1 = 1 + r_1;\ r_1 = g\,\delta_2^0\bar{\rho}\left(1 - f_1^0\right)/\left[2\left(c_2^0\right)^2 - g\delta^0\right];$$

$$f_1^0 = 1 + \left[\left(c_2^0\right)^2/\left(g\delta_2^0\right) - 1\right]/\bar{\rho}\ .$$

Розглянемо тепер «повільну» моду, для якої з точністю до членів другого порядку малості $\left\langle u_1^2\right\rangle = \left[c_2^0\left(\delta - \delta_2'\right)/\delta_1^0\right]^2$, а $\left\langle u_2^2\right\rangle = \left(c_2^0\delta_2'/\delta_2^0\right)^2$. Тут $c_2^0 = \sqrt{g\delta^0 - \left(c_2^0\right)^2}$ — граничне довгохвильове значення швидкості розповсюдження безкінечно малих збуджень «повільної» моди. Подібно тому, як це було зроблено для «швидкої» моди, отримаємо наступне рівняння:

$$\frac{\partial^2\delta_2}{\partial t^2} - \left(c_2^0\right)^2 \frac{\partial^2\delta_2}{\partial x^2} - \alpha_2 \frac{\partial^2(\delta_2')^2}{\partial x^2} - \beta_2 \frac{\partial^4\delta_2}{\partial t^2\,\partial x^2} = 0, \tag{7}$$

де

$$\alpha_2 = g\left[f_2^0 + q_2\Delta\bar{\rho}\left(1 - f_2^0\right)\right]/2 + \left(c_2^0\right)^2\left[q_2/\delta_2^0 + r_2\left(1 - f_2^0\right)/\delta_2^0\right];$$

$$\beta_2 = q_2\left[\left(\delta_2^0\right)^2/3 + \overline{\rho}\delta_1^0\delta_2^0\left(1+f_2^0\right)/2\right] + r_2\left\{\left(\delta_2^0\right)^2\left(1/2+f_2^0\right)/\left[3\left(1-f_2^0\right)\right]\right\};$$

$$q_2 = 1 + r_2;\ r_2 = g\,\delta_2^0\overline{\rho}\left(1-f_2^0\right)/\left[2\left(c_2^0\right)^2 - g\delta^0\right];$$

$$f_2^0 = 1 + \left[\left(c_2^0\right)^2/\left(g\delta_2^0\right) - 1\right]/\overline{\rho}\ .$$

Рівняння (6) та (7) мають стаціонарні розв'язки у вигляді рівнянь хвиль, зокрема, аналогічно роботі [9].

Випишемо розв'язок для усамітненої хвилі «повільної» моди на границі розділу двошарової рідини:

$$\delta_2 = \delta_2^0 + \Delta\delta_2 \operatorname{sch}^2\left(\frac{x - V_2 t}{l_2}\right);$$

$$V_2 = c_2^0\sqrt{1 + \frac{2\alpha_2\Delta\delta_2}{3\left(c_2^0\right)^2}};\ l_2 = \sqrt{\frac{6\beta_2 U_2^2}{\alpha_2\Delta\delta_2}}\,. \qquad (8)$$

При $\alpha_2 < 0$ та $\beta_2 > 0$ має місце усамітнена хвиля типи «впадина» ($\delta_2' < 0$). Очевидно, що з рівнянь (6) та (7) можна отримати еволюційні рівняння, перейшовши до розгляду хвиль, які розповсюджуються у один бік. Так, для збудження границі двошарової рідини з (7) випливає еволюційне рівняння типу Кортевега — де Вріза

$$\frac{\partial\delta_2}{\partial t} + c_2^0\frac{\partial\delta_2}{\partial x} - \alpha_2\frac{\delta_2'}{c_2^0}\frac{\partial\delta_2}{\partial x} + \frac{c_2^0}{2}\beta_2\frac{\partial^3\delta_2}{\partial x^3} = 0, \qquad (9)$$

Рівняння (9) має солітонний (у вигляді скупчень хвиль) розв'язок виду (8), одначе тут швидкість $V_2 = c_2^0 + \dfrac{\alpha_2\Delta\delta_2^0}{3c_2^0}$ та ширина $l_2 = \sqrt{\dfrac{6\beta_2 c_2^0}{\alpha_2\Delta\delta_2}}$. Таким чином, швидкість соліона хвильового рівняння менше за швидкість усамітненого збудження еволюційного рівняння (при одних і тих самих амплітудах), а ширина, навпаки, більше.

Проведене числове дослідження (рис. 1, *а*, *б*) де $\overline{V}_2 = V_2/c_2^0$; $\Delta\overline{\delta}_2 = \Delta\delta_2/\delta_2^0$; $\overline{x}_2 = x/\delta_2^0$ показало хороше співпадіння з експериментальними даними [10]. Видно, що розв'язок хвильового рівняння краще узгоджується з експериментальними точками, ніж розв'язок «усамітненої» хвилі, отриманого з відповідного еволюційного рівняння.

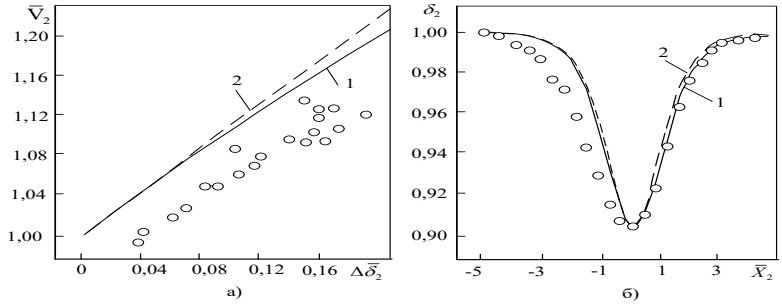



**Рис. 1**. Залежність швидкості солітонів від їх амплітуд (*а*) і форма одного з усамітнених збуджень (*б*) при $\overline{\rho} = 0{,}8$ та $\delta_1^0/\delta_2^0 = 0{,}36$

*1 — розрахунок по модифікованому рівнянню Буссінеска (5.76);*
*2 — розрахунок по рівнянню Кортевега — де Вріза (5.78)*

Для перевірки правомірності застосування техніки «кваліпростих» хвиль до системи хвильових рівнянь порівняємо дисперсійні криві системи лінеаризованих рівнянь (4), (6) та аналогічних рівнянь для «повільної» моди. На рис. 2, *а*, *б* наведено результати розрахунків для $\overline{\rho} = 0{,}8$ та $\delta_1^0 / \delta_2^0 = 0{,}36$. Фазові швидкості $\overline{c}_1 = c_1 / c_1^0$; $\overline{c}_2 = c_2 / c_2^0$; хвильові числа $\overline{k}_1 = k\delta_1^0$; $\overline{k}_2 = k\delta_2^0$. Видно, що дисперсійні криві практично співпадають у широкому інтервалі довжин хвиль.

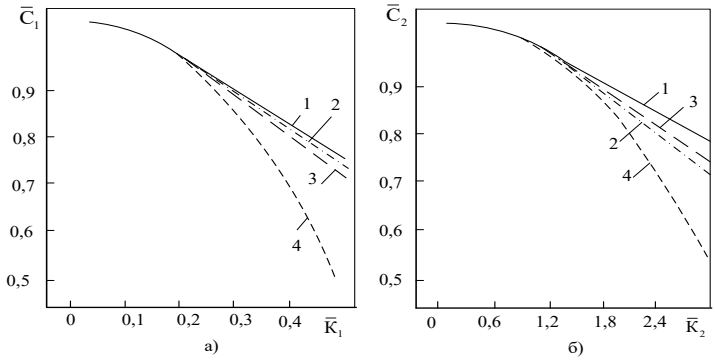



**Рис. 2**. Залежність фазової швидкості «швидкої» (*а*) та «повільної» (*б*) мод хвильового числа:

*1 — точний розв'язок повної лінійної задачі; 2 — розрахунок за формулою (5.73); 3 — розрахунок за формулами (5.75) та (5.76); розрахунок за формулою (5.78)*

Крім того, поведінка фазових швидкостей, віднайдена з отриманих вище рівнянь, слабко відрізняється від точного розв'язку повної лінійної задачі.

Наявність тертя рідин об дно призводить до того, що в модифікованому рівнянні Буссінеска виду (6) з'являється додатковий дисипативний член типу інтегралу Дюамеля [11]. В результаті швидкість розповсюдження гравітаційних хвиль зменшується, а ширина солітонів — збільшується.

**Висновки.** На основі рівняння нерозривності та рівнянь руху для двофазної рідинної системи, що перебуває у нестаціонарному стані, отримано ММ гравітаційної хвилі на границі поділу компонент двошарової структури, причому остання визначається різницею густин рідин, які її утворюють. ММ гравітаційної хвилі отримано у вигляді модифікованого рівняння Буссінеска, яке відрізняється від класичного меншим порядком, одначе відбиває всі якісні властивості фізичної картини явищ хвиле утворення.

Проведені числові дослідження запропонованої ММ гравітаційного хвилеутворення для двошарової рідинної системи, яка не змішується, показали її зручність при розв'язуванні практичних задач, а також цілком придатну точність (з похибкою, що не перевищує $(3...5)\%$ у порівнянні з натурними експериментами) для інженерних розрахунків.

**Список літератури**

1. Yang Z., Xu D., Xiang L. Exponential p-stability of impulsive stochastic differential equations with delays. *Phys. Lett*. 2016. A 359. P.129–137.
2. Wang J., Anisimov M.A. Nature of vapor-liquid asymmetry in fluid criticality. *Phys. Rev*. 2007. E75. P. 58-72.
3. Krasnov G. Air bubble movement in pulsating liquid. *Journal of Mining Science*. 2006. V. 42, №. 5. P. 500–505.
4. Yukhnovskii I. R. Phase Transitions in a Vicinity of the Vapor-Liquid Critical Point *Ukrainian Journal of Physics*. 2019. V.10(1), №33. URL: https://ujp.bitp.kiev.ua/index.php/ujp/article/view/2019662

5. Константінов Ю.М., Гіжа О.О. Технічна механіка рідини і газу. К.: Вища школа, 2002. 277 с.
6. Chorin A.J., Marsden J.E. A mathematical introduction to fluid mechanics [New York: Springer-Verlag, 2000. 169 p.
7. Pozrikidis C. Fluid dynamics: theory, computation, and numerical simulation. Boston: Kluwer Academic Publishers, 2001. 685 p.
8. Shaughnessy E.J., Shaughnessy J., Ira M., Katz J.P. Schaffer Introduction to fluid mechanics. New York, Oxford: Oxford University Press, 2005. P. 1018-1026.
9. Митько Л.О. Положаєнко С.А., Сєрбов М.Г. Моделювання процесу хвилеутворення при донних зрушеннях в мілководих акваторіях. *Математичне та комп'ютерне моделювання. Серія: Технічні науки.* Кам'янець-Подільський: Кам'янець-Подільськ. нац. ун-т, 2009. Вип. 2. С. 95-104.
10. Марценюк О.С., Немирович П.М., Віценко О.М., Пастушенко І.М. Методика гідравлічного розрахунку газорідинних циркуляційних труб. *Наукові праці НУХТ*. 2012. № 44. С. 44–50.
11. Корн Г., Корн Т. Справочник по математике. М.: Наука, 1977. 831с.

## DEVELOPMENT AND NUMERICAL IMPLEMENTATION OF THE MATHEMATICAL MODEL OF A GRAVITY WAVE AT THE BOUNDARY OF SEPARATION OF A TWO-LAYER LIQUID SYSTEM

D.A. Lys, A.Yu. Prokofiev

National Odesa Polytechnic University, Shevchenko str., 1, Odesa, 65044, Ukraine
loreal338@gmail.com, fallbrick1985@gmail.com

Wave dynamics of two-phase systems is a new branch of the mechanics of heterogeneous systems and thermal physics, which has been developing rapidly in recent years. Research in this field requires the involvement of modern results of nonlinear wave dynamics, the development of new ways of taking into account interphase interaction, the study of modern concepts of wave movements (flows), such as «kinematic waves», «dynamic waves» and «multi-wave» systems. The complexity of the physical formulation of these problems requires appropriate and adequate mathematical models (MM) of these processes, as well as constructive methods of numerical implementation of the created MM. In addition, the relevance of such studies is due to the fact that two-phase flows in the vast majority occur in operating modes in technological installations of energy, chemical, metallurgical and other important sectors of the national economy. Work processes in the oil-mining and oil-refining industries, in cryogenic equipment are accompanied by the formation of a special type of two-phase systems - vapor-liquid mixtures. It is known that most two-phase systems are characterized by the property of significant compression (that is, the speed of sound in such a system is low), not by linearity, and therefore to calculate the dynamics of such media that move at relatively low speeds, it is necessary to use special gas-dynamic methods. Also, the movement of two-phase systems is accompanied by interphase heat exchange processes, which cause strong dissipation of the medium, and the inertial properties of gas inclusions give rise to the dependence of the speed of sound on the frequency - the dispersion of the sound speed. Therefore, the methods and associated MM of traditional vapor-liquid dynamics do not meet the specifics of two-phase flows and give unsatisfactory results in calculations. In other words, these MM and methods of their numerical implementation are not adequate to the complex physical processes under study. In the current work, the task of constructing constructive MM gravity waves, which are formed at the phase separation boundary of a two-layer liquid system, is set. The proposed MM qualitatively reflect the essence of the physics of dynamic processes in a two-phase liquid system and represent model canonical equations formulated under the conditions of accepted assumptions regarding the course of the studied processes. The conducted numerical studies showed that the correct formalization of the features of physical phenomena within the framework of the proposed models allows revealing the nature of the regularities of hydro-gas-dynamic flows when setting up real applied experiments with sufficient engineering practice.

**Keywords**: two-phase system; gravitational wave; two-layer liquid system, mathematical model; numerical experiment.

DOI 10.15276/imms.v13.no1-2.104
УДК 621.396

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp. 104-111

# АНАЛІЗ ЗАСТОСУВАННЯ МЕТОДІВ КЛАСТЕРИЗАЦІЇ НА ОСНОВІ КОМБІНОВАНОЇ ВАГИ В РАДІОМЕРЕЖАХ КЛАСУ MANET

К.В.Лукіна, С.О.Клімович

Військовий інституту телекомунікацій та інформатизації імені Героїв Крут, вул. Князів Острозьких, 45/1, Київ, 01011, Україна, e-mail: kateryna.lukina@viti.edu.ua

Використання мереж з децентралізованим управлінням, представниками яких є мобільні радіомережі класу MANET (Mobile Ad-Hoc Networks,) є одним з напрямків застосування бездротових технологій в телекомунікаційних системах. Для радіомереж класу MANET характерна відсутність фіксованих маршрутів передачі інформації, відсутність стаціонарних базових станцій, відсутність фіксованої мережевої інфраструктури. Такі характерні риси роблять можливим застосування радіомереж даного типу в тактичній ланці управління, що дозволить забезпечувати обмін інформацією в інтересах підрозділів високої мобільності. Вимоги до системи зв'язку з боку мобільних підрозділів потребують нових підходів до організації системи управління мережею. Варіантом системи управління мобільною радіомережою класу MANET, при застосуванні в тактичній ланці управління, є розбиття мережі на зони та створення сукупності локальних центрів управління зонами у вигляді вузлів (вузлів-координаторів). При цьому виникає задача вибору вузлів зон, які будуть виконувати функції управління іншими вузлами та взаємодіяти між собою, тобто – вузлів координаторів. Одним із шляхів вибору вузлів-координаторів є застосування методів кластеризації (алгоритмів вибору головного вузла кластера), приймаючи зону, як кластер. Проведено аналіз алгоритмів вибору головного вузла кластера в методах кластеризації на основі комбінованої ваги. Аналіз проведено з метою визначення переваг та недоліків існуючих алгоритмів та можливості їх застосування для визначення головного вузла кластеру в мобільних радіомережах класу MANET. В результаті аналізу зроблено висновок, що алгоритм FWCA (Forecast weight based clustering algorithm) є найбільш перспективним для застосування в розробці методів вибору вузла-координатора в радіомережіх класу MANET.

**Ключові слова:** кластеризація, MANET, вузол-координатор, комбінована вага, координація.

**Вступ.** Створення та використання мереж з децентралізованим управлінням є одним з напрямків використання безпроводових технологій в телекомунікаційних системах передачі даних. Представником цих технологій є мобільні радіомережі (МР) класу MANET (Mobile Ad-Hoc Networks) - радіомережі з динамічною архітектурою. В таких радіомережах передбачена відсутність базових станцій, фіксованої мережевої інфраструктури та фіксованих маршрутів передачі інформації. Всі вузли мережі мобільні і здійснюють обмін інформацією безпосередньо між собою або ретранслюють пакети, що передаються [1].

Сферою застосування мобільних радіомереж класу MANET є, в тому числі, аварійні мережі, які розгортаються в умовах надзвичайних ситуацій. Характерні риси мереж даного класу роблять можливим застосування їх в тактичній ланці управління, і дозволяють забезпечувати обмін інформацією в інтересах всіх військ [2]. При цьому, МР тактичної ланки управління мають наступні особливості: значну розмірність мережі, наявність вузлів з різною мобільністю, потужністю, часто низька пропускна спроможність радіоканалів та ін.

Необхідність вирішення задач управління мобільною радіомережою тактичної ланки управління потребує створення ефективної системи управління,

для координації роботи вузлів мережі та забезпечення інформаційного обміну з заданою якістю. Варіантом системи управління мобільною радіомережею є розбиття мережі на зони та створення сукупності локальних центрів управління зонами у вигляді вузлів [3].

При цьому, задача управління зоною може бути вирішена за допомогою виділення серед мобільних вузлів цієї зони головного вузла, якій буде виконувати функції управління іншими вузлами та взаємодіяти з головними вузлами інших зон, тобто - визначення вузла-координатора. Вузол-координатор, крім функцій прийому, передачі та ретрансляції інформації, буде виконувати додаткову задачу - створення оптимальних умов для виконання цілей управління всіма вузлами зони і всієї мобільної радіомережі.

Вибір вузла-координатора є важливим етапом процесу створення системи управління як окремою зоною, так і всієї мобільною радіомережею. Тому, актуальною є задача розробки методів визначення вузла - координатора (ВК) з урахуванням особливостей мобільних радіомереж класу MANET тактичної ланки управління [5].

**Мета статті та постановка задачі.** Метою статті є аналіз алгоритмів пошуку вузла-координатора в МР класу MANET, які застосовуються в методах кластеризації на основі зваженої ваги. Для досягнення мети в роботі проводиться аналіз існуючих алгоритмів вибору головного вузла кластера. В основу визначення методу з найкращим алгоритмом вибору головного вузла покладено низку критеріїв: кількість вузлів, мобільність вузлів, потужність, заряд батареї, пропускна спроможність радіоканалу. Найкращим вважитимемо той метод, алгоритм вибору головного вузла якого враховує крім поточного ще й попередні значення ваги вузла. Об’єктом дослідження є процес кластеризації з визначенням головного вузла кластеру. Предмет – алгоритми вибору головного вузла кластеру.

**Основна частина.** Головний вузол або вузол-координатор визначається серед інших вузлів залежно від характеристик, таких як апаратне оснащення, розташування в топології радіомережі, кількість сусідів та інше. Крім звичайних функцій, які виконує будь-який вузол, таких як прийом, передача та ретрансляція інформації в мережі, головний вузол має завдання створення умов для досягнення цілей управління.

Під час планування мережі вузол-координатор може бути визначений організаційним шляхом. Проте, під час функціонування МР можливі випадки, коли організаційно призначений вузол стає непридатним для подальшої роботи щодо створення умов для досягнення цілей управління. Це може бути пов’язано з різними причинами, такими як вихід з ладу обладнання вузла, втрата зв’язку між вузлами через збільшення допустимої відстані зв’язку або через властивості рельєфу місцевості та ін.

У зв’язку з цим, виникає необхідність у виборі нового вузла-координатора. Для обґрунтованого вибору необхідна наявність наступної інформації: дані про структуру мережі; характеристики кожного вузла мережі; ознаки (критерії), значення яких є пріоритетними при здійсненні вибору вузла-координатора. У зв’язку з динамічним характером мобільних радіомереж класу MANET та стохастичною природою їх функціонування, в [6] запропоновано розв’язувати задачу пошуку вузла-координатора за допомогою методів кластеризації. Кластеризація – це процес розбиття множини об’єктів на групи, де об’єкти всередині кожної групи подібні між собою, а між об’єктами з різних груп є значні відмінності [7]. Кожен кластер складається з одного головного вузла (Cluster Head або CH), декількох вузлів нижчого рівня, які є членами кластера (Cluster Member або CM), та шлюзів (Gateway), які можуть сприймати два або більше головних вузлів кластерів (рис.1). Головний вузол кластера (далі - ГВК) відповідає за

координацію всіх інших вузлів у кластері. До особливостей МР тактичної ланки управління відносяться: висока динаміка топології, значна розмірність, низька пропускна спроможність радіоканалу, неоднорідність вузлів (за продуктивністю, потужністю, мобільністю) та ін.

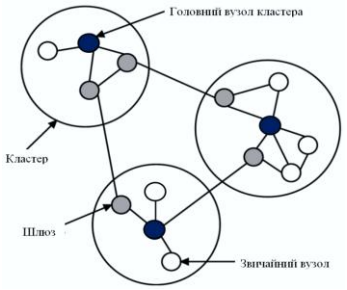



**Рис. 2.** Структура кластера

Щоб забезпечити кластеризацію, вибір головного вузла кластера має бути здійснений із врахуванням властивостей мережі.

Визначення вузла-координатора можна розділити на два етапи: перший – розділення мережі (або зони) на кластери, другий – вибір вузла-координатора серед інших вузлів, що належать до обраного кластера.

Алгоритм вибору вузла повинен бути підпорядкований властивостям мережі, таким як територіальний розподіл вузлів мережі (зони) , характеристики та взаємне розташуванням вузлів мережі (зони), їх параметрами. В [7, 8] приведено класифікацію методів кластеризації на основі яких ґрунтується вибір головного вузла кластеру.

В результаті аналізу груп алгоритмів виявлена перевага алгоритму на основі комбінованої ваги (головний вузол кластеру обирається за результатами оцінки ваги кожного з вузлів мережи). Під вагою розуміємо набір певних показників, характеристик, можливостей. Тому, пропонується розглянути алгоритми саме цієї групи.

В [7-10] розглядається алгоритм **WCA** *(Weighted Cluster Algorithm*), який здійснює вибір головного вузла кластеру у відповідності з: кількістю вузлів, які він може обслуговувати; відстанню між вузлами, мобільністю вузлів; потужністю передавачів; потужністю акумулятора кожного вузла. Цей алгоритм не є періодичним, процедура вибору головного вузла здійснюється або при переміщенні вузла, або при виході вузла з ладу. Для застосовують заздалегідь визначену межу (поріг), який відображує оптимальну кількість вузлів. Вага вузла $v$ визначається з виразу:

$$W_v = w_1 \Delta v + w_2 D_v + w_3 M_v + w_4 P_v \quad (1)$$

де $W_v$ – вага вузла $v$; $w_1, w_2, w_3, w_4, w_5$ – вагові коефіцієнти відповідних параметрів; $\Delta v$ - різниця між граничною кількістю вузлів, які може обслуговувати головний вузол кластеру та кількістю вузлів; $D_v$ – сума відстаней від вузла $v$ до всіх його сусідів; $M_v$ – міра мобільності (рухливості); $P_v$ – загальний час перебування вузла головним вузлом кластеру.

Вагові коефіцієнти обираються таким чином, щоб задовільнити умову:

$$w_1 + w_2 + w_3 + w_4 = 1 \quad (2)$$

Головою кластеру обирається вузол з найменшою вагою.

Алгоритм **WBACA** (*Weight-based adaptive clustering algorithm*) розглядається в [7] – це адаптований зважений алгоритм кластеризації. Недоліком алгоритму WCA є те, що кожен вузол повинен знати ваги всіх інших вузлів ще до початку процесу кластеризації. Цей процес потребує багато часу. Підхід, запропонований в

алгоритмі WBACA ґрунтується на залученні глобальної системи навігації GPS (Global Position System) для отримання інформації про місцезнаходження вузлів.

В WBACA для визначення головного вузла кластеру враховуються параметри: потужність передачі, швидкість передачі, мобільність, потужність батареї та пропускна спроможність радіоканалу. Вузол з найменшою вагою обирається головним. Вага вузла *v* визначається як:

$$W_v = w_1 M + w_2 B + w_3 T_x + w_4 D + {}^{w_5}\!/_{T_R} \quad (3)$$

де $w_1, w_2, w_3, w_4, w_5$ – вагові коефіцієнти для відповідних параметрів; $M$ – мобільність (рухливість) вузла; $B$ – ємність акумуляторної батареї; $T_x$ – потужність передачі; $D$ – сума відстаней від вузла *v* до всіх його сусідів, $T_R$ – пропускна спроможність радіоканалу.

Головні вузли кластерів, які перетинаються, зв'язуються один з одним через шлюз. Звичайні вузли знаходяться на відстані 1 крок від голови кластеру.

Алгоритм **EWBCA** (*An Efficient Weight-based clustering algorithm*) розглядається в [11]. Алгоритм розроблений для покращення використання таких обмежених ресурсів, як пропускна здатність та енергія, шляхом створення стабільних кластерів, мінімізації накладних витрат на маршрутизацію та збільшення пропускної здатності мережі. В алгоритмі кожен вузол має певну вагу (якість), яка визначає його придатність бути головним вузлом кластера. Вага обчислюється чотирма параметрами: кількість сусідів, залишковий заряд акумулятора, стабільність та відстані до усіх сусідів [9].

Алгоритм **iWCA** (*Improved Weight Clustering Algorithm*) проаналізовано в [12, 13]. Модифікований (вдосконалений) зважений алгоритм кластеризації ваги. Вдосконаленням цього алгоритму є можливість застосування його в сенсорних мережах, з урахуванням конкретних обмежень. До формули оцінки додається фактор оцінки характеристики вузла [10]. Таким чином вузли, що обираються в якості головного вузла кластера, можуть мати ефективнішу поведінку в неоднорідних сенсорних мережах, ніж ті, що не мають додаткового фактора.

Обчислюється загальна вага для кожного вузла V за формулою:

$$W_v = w_1 \Delta v + w_2 D_v + w_3 M_v + w_4 T_v + w_5 C v \quad (4)$$

В якості головного вузла кластеру обирається вузол з мінімальною вагою $W_v$. Обрані головні вузли кластерів діятимуть як вузли додатків у бездротовій мережі і можуть змінюватися через різні часові інтервали. Через фіксований інтервал часу цей алгоритм повторно запускається знову, щоб додати нові вузли додатків, так що очікується, що тривалість життя системи триватиме довше.

Алгоритм **FWCA** (*Forecast weight based clustering algorithm*) – прогнозований зважений алгоритм кластеризації ґрунтується на основі WCA, який розраховує вагу кожного вузла та обирає в якості головного вузла кластера вузол з найменшою вагою, розглянуто в [11, 14]. У випадку використання WCA, головний вузол кластера обирається лише за миттєвим значенням ваги. У разі виникнення факторів, які можуть вплинути на можливість вузла надіслати миттєве значення ваги, наприклад, через високе навантаження мережі, цей вузол не буде врахований під час вибору кластера, хоча міг би бути обраним навіть головним вузлом. Отже, може статися неправильний вибір головного вузла кластера. Щоб запобігти виникненню такої ситуації, був запропонований алгоритм FWCA. На відміну від WCA, FWCA враховує крім поточного ще й попереднє значення ваги вузла. Це призводить до більш зваженого вибору головного вузла кластера.

Розрахунок прогнозованої ваги здійснюється за допомогою обчислення експоненційного ковзного середнього (EMA – exponential moving average) [15]. Вибір EMA обумовлений тим, що його обчислення не вимагає всіх попередніх

даних, залежить лише від поточного значення зважування та попереднього значення прогнозованої ваги. Вузли кластера транслюють лише свої значення ваги.

Вага розраховується за виразом:

$$FW = aW_{current} + (1-a)FW_{previous} \quad (5)$$

де $a$ – коефіцієнт згладжування, регульований параметр, має значення від 0 до 1.

Вага кожного вузла розраховується методом WCA згідно формул (2, 3).

В методі FWCA прогнозовану вагу розраховують згідно формули:

$$FW_{i(t+1)} = a\sum_{k=0}^{t-1}(1-a)^k FW_{i(t-k)} + (1-a)^t W_i \quad (6)$$

де $FW_{i(t+1)}$ – прогнозована вага за період (*t+1*) виміряна в час *t*; $W_i$ – поточне значення в час *t*, $FW_{i(t-k)}$ – прогнозована вага за попередній період часу *(t-1)*.

В [7] розглядається алгоритм **VBCA** (*Vote-based clustering algorithm*) - алгоритм кластеризації на основі голосування, оцінюється за двома факторами: кількістю сусідів та залишком заряду батареї кожного вузла) [7]. Кожен мобільний вузол має унікальний ідентифікатор (*ID*), який є цілим числом. Основною інформацією всередині мережі є *hello*-повідомлення, які передаються від вузлів. Використовуючи цю інформацію та інформацію про заряд батареї, алгоритм представляє концепцію «голосування». Формат hello-повідомлення наведений нижче (рис. 3).

| MH_ID | CH_ID | Vote | Option |
|---|---|---|---|

**Рис. 3.** Формат hello-повідомлення

MH_ID – власний ідентифікатор мобільного вузла, CH_ID – ідентифікатор мобільного вузла, що виконує функції головного вузла кластера, Vote – кількість голосів, тобто зважена сума кількості сусідів та залишку заряду батареї, розраховується за формулою (7). Елемент опції використовується для реалізації балансу навантаження кластера:

$$Vote = w_1 \cdot (\frac{n}{N}) + w_2 \cdot (\frac{m}{M}) \quad (7)$$

де $w_1, w_2$ – зважені коефіцієнти кількості сусідів та заряду батареї відповідно, $n$ – кількість сусідів, $N$ – розмір мережі або максимальна кількість вузлів в кластері, $m$ – залишок заряду батареї, $M$ – максимальний залишок заряду батареї.

Кожен мобільний вузол надсилає hello-повідомлення протягом періоду встановлення з'єднання. Якщо мобільний вузол є новим в мережі, то він надсилає відповідь типу: «CH_ID» яка означає, що вузол не був раніше в жодному кластері і не має інформації про своїх сусідів. Кожен мобільний вузол рахує скільки hello-повідомлень він отримав протягом періоду встановлення з'єднання та присвоює собі їх кількість, як його власний id. Кожен вузол надсилає іншому ще одне hello-повідомлення, в якому для здійснення "ранжування" встановлюється власне значення рангу, яке отримується з рівняння. Формуючи привітальне повідомлення під час другого циклу привітання, кожен вузол знає, що відправник, з найбільшим рангом не належить жодному існуючому кластеру, є головним вузлом цього кластеру. Він відправляє в наступному привітальному повідомленні позначку "ch_id" та значення ідентифікатора голови кластера. У випадку коли два або більше мобільних вузла отримають однакове число привітальних повідомлень, в пріоритеті буде той, хто має нижчий id. Дослідження показали, що кожен мобільний вузол знає головний вузол кластера після двох періодів привітальних повідомлень. Алгоритм **PMW** (*Power, Mobility and Workload*) розглянуто в [16] – зважений алгоритм кластеризації, де вага кожного вузла обчислюється за трьома

параметрами: потужністю, мобільністю та навантаженням. Цей алгоритм формує стабільні кластери та має високу масштабованість [9].

Порівняльна характеристика розглянутих алгоритмів представлена в таб.1.

Загальною перевагою алгоритмів кластеризації на основі зваженої ваги є їх здатність враховувати кілька показників одночасно, у зв'язку з тим, що показники, їх кількість, їх вагу можливо встановлювати в залежності від вимог до вибору вузла-координатора для певної мережі, для певних вимог.

Загальним недоліком всіх цих алгоритмів кластеризації є обчислювальна складність, яка підвищується при збільшенні рухомості вузлів, ймовірність якої при застосуванні в тактичній ланці управління висока. Тобто, вузол буде витрачати час на визначення вузла-координатора при кожній зміні місцеположення вузла. Тому, доцільно застосовувати алгоритми кластеризації на основі зваженої ваги для мобільних мереж, де рухомість вузлів не дуже висока або для таких мобільних мереж, де обмін службовою інформацією є більш важливим, чим передача інформації. Для таких випадків серед алгоритмів кластеризації на основі зваженої ваги найкращим є алгоритм FWCA, у зв'язку з тим, що дозволяє запобігати ситуаціям, коли нові вузли кластеру не враховуються при виборі головного вузла.

**Таблиця 1**

Порівняльна характеристика алгоритмів кластеризації на основі комбінованої ваги

| Характеристики / Алгоритм | Параметри ваги, що враховуються | Принцип вибору ГВК | Переваги | Недоліки |
|---|---|---|---|---|
| WCA | кількість вузлів в зоні, мобільність, потужність передачі, заряд батареї | Вузол з найменшою вагою | Запобігає перевантаженню вузлів | Необхідність знання ваги вузлів. |
| WBACA | потужність передачі, швидкість передачі, мобільність, потужність батареї | Вузол з найменшою вагою | Залучення GPS | Потребує багато часу |
| EWBCA | кількість сусідів, залишковий заряд батареї, стабільність, відстані до сусідів | Вузол з найменшою вагою | Мінімальні витрати ресурсів | - |
| IWCA | кількість вузлів, мобільність, потужність передачі, потужність батареї, поводження в сенсорних мережах | Вузол з найменшою вагою | Можливість застосування в сенсорних мережах | - |
| FWCA | кількість вузлів, мобільність, потужність передачі, потужність батареї | Вузол з найменшою вагою | Застосування EMA | - |
| VBCA | кількість сусідів, залишок заряду батареї | Вузол з найменшим id | Швидкість вибору ГВК | - |
| PMW | потужність, мобільність, навантаження | Вузол з найменшою вагою | Збільшує тривалість життя мережі | - |

**Висновки.** В статті проведено аналіз методів кластеризації відповідно вимог застосування їх в радіомережах класу MANET. Проведено аналіз алгоритмів на основі комбінованої ваги, як перспективних щодо застосування в радіомережах обраного класу. Аналіз показав, що застосування приведеної групи алгоритмів забезпечує необхідну в радіомережах класу MANET мобільність та мінімальність споживання енергії для забезпечення тривалості життя мережі. Серед алгоритмів на основі комбінованої ваги, алгоритм FWCA є найбільш перспективним, у зв'язку

з тим, що дозволяє запобігати ситуаціям, коли нові вузли кластеру не враховуються при виборі головного вузла.

Водночас, ці методи мають високу обчислювальну складність, у зв'язку з тим, що вузли в мережі переміщуються. Із збільшенням мобільності більше часу, трафіку та потужності буде витрачатися на процес перерахунку головного вузла. Тобто, доцільно застосовувати методи на основі зваженої ваги в мобільних мережах, де рухомість вузлів низька.

Напрямком подальших досліджень є розробка та наукове обґрунтування механізму застосування алгоритмів кластеризації на основі комбінованої ваги на практиці, в умовах активних бойових дій.

**Список літератури**


1. Сова О.Я., Романюк В.А., Міночкін Д.А., Романюк А.В. Методи обробки знань про ситуацію в мобільних радіомережах класу MANET для побудови вузлових інтелектуальних систем управління. *Збірник наукових праць ВІТІ ДУТ.* 2014. С. 97-110.
2. Романюк В.А. Еволюція тактичних радіомереж. *Пріоритетні напрямки розвитку телекомунікаційних систем та мереж спеціального призначення:* тези доп. VI наук.-практ. семінару (м. Київ 20 жовтня 2011 року). Київ, 2014. С. 45-52.
3. Романюк В.А., Сова О.Я., Жук О.В. Архітектура системи управління мережами MANET. *Проблеми телекомунікацій – 2011*: тези доп. V міжнар. конф., (м. Київ, 19-22 квітня 2011р.). Київ, 2011. С. 58–60.
4. Романюк В., Стемпковська Я., Симоненко О., Сова О. Координація цільових функцій інтелектуальних систем управління тактичними радіомережами класу MANET. *Збірник наукових праць ХУПС*. 2014. №3 (40). С. 85–92.
5. Романюк В.А. Інтелектуальні мобільні радіомережі. *Пріоритетні напрямки розвитку телекомунікаційних систем та мереж спеціального призначення*: зб. мат. V наук.-тех. конф., (м. Київ, 20-21 жовтня 2010 року). Київ, 2010. С. 28–36.
6. Сова О., Лукіна К., Олексенко В., Шаповал О. Аналіз методів кластеризації для визначення вузла-координатора в мобільних радіомережах класу MANET. *Збірник наукових праць ВІТІ НТУУ*. 2017. №4. С. 121-128.
7. Agarwal R., Motwani M.. Survey of clustering algorithms for MANET. *International Journal on Computer Science and Engineering.* 2009. Vol. 1(2). P. 98-104.
8. Лукіна К., Сова О., Марилів О., Олексенко В. Аналіз вибору головного вузла кластера в радіомережах класу MANET. *Збірник наукових праць ВІТІ*. 2018. №3. С. 38-48.
9. 9. Malhotra P, Dureja A. A Survey of Weight-Based Clustering Algorithms in MANET *IOSR Journal of Computer Engineering (IOSR-JCE)*. 2013. Vol. 9. Issue 6. P.34-40. URL: www.iosrjournals.org.
10. Goriya N., Rajput I.J., Mehta Mihir A survey paper on cluster head selection techniques for Mobile ad-hoc network *IOSR Journal of Computer Engineering (IOSR-JCE)*. 2015. Vol. 17. Issue 1. P. 34-39. URL: www.iosrjournals.org.
11. Bentaleb A., Boibetra A., Harous S. Survey of Clustering Schemes in Mobile ad-hoc networks *Communications and Network*. 2013. 5. P.8-14
12. Gupta K.D., Prakash J. Cluster head selection algorithms in MANET: a survey. *International Journal of Advance Research in Science and Engineering*. 2017. Vol. 6. Issue 02. P.342-347.
13. Tzung-Pei Hong, Cheng-Hsi Wu An Improved Weighted Clustering Algorithm for Determination of Application Nodes in Heterogeneous Sensor Networks. *Journal*

*of Information Hiding and Multimedia Signal Processing*. 2011.Vol. 2, №2. P.173-184
14. Vijayalakshmi J., Prabu K. A Survey of various weighted based clustering algorithm for MANET. *International Journal of Data Mining Techniques and Applications*. 2018. Vol. 07. Issue 01, P. 146-153.
15. Експоненційне ковзне середнє URL: https://wiki.tntu.edu.ua.
16. Poonam Thakur Clustering sachems in wireless sensor networks and mobile adhoc network: classification and comparison. *International Journal of computer networks and wireless communications*. 2012. Vol.2, №6.

## ANALYSIS OF THE APPLICATION OF CLUSTERING ALGORITHMS BASED ON COMBINED WEIGHT IN MANET CLASS RADIO NETWORKS.

K.V. Lukina, S.O. Klimovych

Military Institute of Telecommunications and Informatization Technologies named after Heroes of Kruty 45/1, Kniaziv Ostrozkyh St, Kyiv, 01011, Ukraine, e-mail: kateryna.lukina@viti.edu.ua

The use of networks with decentralized management, represented by mobile radio networks of the MANET class (Mobile Ad-Hoc Networks), is one of the areas of application of wireless technologies in telecommunication communication systems. Radio networks of the MANET class are characterized by the absence of fixed information transmission routes, the absence of stationary base stations, and the absence of a fixed network infrastructure. Such characteristic features make it possible to use radio networks of this type in the tactical chain of command, which will ensure the exchange of information in the interests of all troops. This requires new approaches to the organization of the network management system. A variant of the MANET-class mobile radio network control system, when applied in the tactical control chain, is the division of the network into zones and the creation of a set of local zone control centers in the form of nodes (coordinator nodes). At the same time, there is a problem of selecting zone nodes that will perform the functions of controlling other nodes and interacting with each other, that is, coordinator nodes. One of the ways to select coordinator nodes is to use clustering methods (algorithms for selecting the main node of the cluster), taking the zone as a cluster. The article analyzes algorithms for selecting the main cluster node in clustering methods based on combined weight. The analysis was carried out in order to determine the advantages and disadvantages of the existing algorithms and the possibility of their application to determine the main node of the cluster in mobile radio networks of the MANET class. As a result of the analysis, it was concluded that the FWCA algorithm (Forecast weight based clustering algorithm) is the most promising for use in the development of methods for selecting a coordinator node in MANET class radio networks.

**Keywords:** clustering, MANET, coordinator node, combined weight, coordination.

DOI 10.15276/imms.v13.no1-2.112
УДК 004.42:004.056.57

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp. 112-118

# РОЗРОБКА ПРОГРАМНОГО ДОДАТКУ ДЛЯ РОБОТИ ЗІ СТАНДАРТАМИ КІБЕРБЕЗПЕКИ

О.О. Лановська, О.Ю. Лебедєва

Національний університет «Одеська політехніка»
пр. Шевченко 1, Одеса, 65044, Україна
e-mails: lanovska.8088987@stud.op.edu.ua, o.y.lebedieva@op.edu.ua

Розроблено програмний додаток, спрямований на роботу із стандартами кібербезпеки та файлами XML-формату. Мета розробки цього програмного додатку полягає у спрощенні процесу роботи зі стандартами кібербезпеки у форматі XML. Додаток буде мати інтуїтивно зрозумілий інтерфейс, що дозволить користувачам зручно створювати, редагувати та переглядати XML-файли, пов'язані з кібербезпекою. Міжнародна організація зі стандартизації ISO займається розробкою та публікацією стандартів, що включають рекомендації та передові практики для забезпечення кібербезпеки. Ці стандарти відображають світові норми та найкращі практики у сфері кібербезпеки і широко використовуються в бізнесі та інших секторах з метою забезпечення безпеки в Інтернеті. Основними задачами такого програмного продукту є забезпечення комфортної та зручної взаємодії користувача із вмістом стандартів кібербезпеки, створення платформи для пошуку необхідних користувачеві визначень в стандарті та взаємодії із текстом, а також забезпечення можливості формування за визначеною схемою XML-файлів на основі існуючих файлів стандартів кібербезпеки у PDF форматі. XML є розповсюдженим форматом для зберігання та передачі даних та найчастіше використовується для зберігання технічної документації, тому у контексті зберігання стандартів цей формат є доволі перспективним і відкриває більше різнопланових можливостей для взаємодії із ними. У роботі було проведено аналіз предметної області та сучасних існуючих аналогів – у результаті аналізу виявлено, що усі аналоги не забезпечують виконання необхідних задач, а також було розроблено перелік тегів та атрибутів для внутрішньої структуризації XML-файлів стандартів кібербезпеки. Результати даної роботи можуть бути використані під час навчального процесу з метою швидкого доступу як для викладачів, так і для студентів, до потрібних стандартів і визначень, а також порівняння визначень термінів з різних стандартів. Практична цінність цього програмного продукту полягає в можливості активного використання його всіма учасниками навчального процесу, а також зацікавленими особами.

**Ключові слова:** кібербезпеки, інформаційна безпека, стандарт, ISO, XML-файл, теги, атрибути, додаток-ґелпер.

**Вступ.** Стандарти кібербезпеки (інформаційної безпеки) є набором відкритих методів, призначених для захисту користувачів та організацій у кіберсередовищі. Використання цих стандартів спрямоване на зниження ризиків, особливо пов'язаних з інформаційною безпекою.

Дотримуючись рекомендацій, що містяться у стандартах кібербезпеки, організація або підприємство може забезпечити надійний захист інформації, а також інформаційних систем, які використовуються під час робочого процесу. З огляду на значне зростання дистанційної роботи в бізнесі та постійну загрозу в онлайн-середовищі, наявність сертифікації організації з відповідністю певним стандартам безпеки стає все важливішою на ринку. Ця важливість зростає кожен рік.

В кінці 2017 року, близько 40 000 дійсних сертифікатів були зафіксовані ISO (міжнародна організація зі стандартизації) від 160 країн. Це вказує на середній темп

зростання 19,4% протягом 12-річного періоду з 2006 по 2017 рік. Крім того, ці дані свідчать про те, що близько 60% цих сертифікатів належать до IT-сектора [1].

Таким чином необхідність роботи із стандартами, забезпечення їх зручного перегляду та зберігання з кожним роком тільки зростає. І зростає актуальність створення таких програмних рішень, які б спростили дану роботу.

**Мета та задачі роботи.** Метою роботи постає розробка такого програмного рішення, яке б дозволило спростити та зробити більш комфортною роботу зі стандартами кібербезпеки.

В процесі досягнення мети виконуються наступні задачі:

— аналіз предметної області та існуючих аналогів;

— аналіз технологій розробки додатку для роботи із xml-файлами;

— розробка алгоритму додатку для роботи із стандартами кібербезпеки та визначення основного функціоналу;

— розробка додатку для роботи із стандартами кібербезпеки;

— забезпечення в додатку необхідних функцій для роботи із стандартами кібербезпеки.

**Основна частина.** На даний момент єдиною платформою для роботи зі стандартами, в тому числі і стандартами інформаційної безпеки та кібербезпеки є сайт Міжнародної організації стандартизації – ISO.

ISO є незалежною неурядовою міжнародною організацією зі стандартизації, яка включає в себе 168 національних органів стандартизації. Шляхом співпраці зі своїми членами, вона збирає експертів з метою обміну знаннями та розробки добровільних міжнародних стандартів, які базуються на консенсусі та відповідають потребам ринку. Ці стандарти підтримують інновації та надають рішення для глобальних проблем [2].

Дана платформа зберігає усі створені нею стандарти та забезпечує можливості перегляду певних їх частин та пошуку за певними критеріями (кодом, назвою, більш детальною інформацією). Але дана платформа охоплює занадто велику сферу стандартизації та спеціалізується на роботі саме із стандартами, що були розроблені власними спеціалістами – що також означає те, що вона орієнтована на документи, представлені лише англійською мовою.

Також на сайті ISO існує можливість придбання необхідного стандарту у форматі PDF. Взагалі зберігання стандартів у PDF форматі є найбільш поширеною практикою, що відкриває величезний простір для розробки програмного забезпечення, яке б орієнтувалося на роботу з іншими форматами файлів.

Найбільш цікавим та перспективним для подальшого розвитку у сфері зберігання та роботи зі стандартами представляється формат XML.

XML (Extensible Markup Language) – це розширювана мова розмітки, що використовується для опису даних. Стандарт XML – це гнучкий спосіб створення інформаційних форматів і електронного обміну структурованими даними через загальнодоступний Інтернет, а також через корпоративні мережі. Основною функцією XML є створення форматів для даних, які використовуються для кодування інформації для документації, записів бази даних, транзакцій і багатьох інших типів даних [3].

Файл XML містить код XML і закінчується розширенням файлу «.xml». Він містить теги, які визначають не лише те, як має бути структурований документ, але й те, як його слід зберігати та транспортувати через Інтернет [4].

Саме файли XML формату найчастіше використовуються для зберігання технічної документації. Тому ідея зберігати стандарти кібербезпеки в такому форматі є доволі перспективною.

Сучасні додатки та онлайн платформи для роботи з файлами XML формату не є бездоганними. Їх можна розділити на дві окремі групи: ті, що виводять тільки

набір тегів; та ті, що виводять сам зміст даних тегів у вигляді звичайного суцільного тексту.

Прикладами першої групи можуть слугувати такі найбільш популярні онлайн платформи, як Code Beautify, JSON Formatter, Tutorialspoint, а також настільний додаток XML NOTEPAD. Проте ці продукти не забезпечують зручного меню для пересування за тегами та можливості побудови ієрархічного древа змісту – в принципі формат їх роботи не є комфортним та зрозумілим для користувача. До другої ж групи можна віднести, наприклад, таку онлайн платформу як GROUPDOCS (XML аналізатор), або також звичайний Microsoft Word – проте знову ж таки дані продукти не забезпечують якісного відображення змісту файлу, так як вони орієнтовані тільки на вміст тегів, а не їх специфіку та атрибути.

Отже, постає необхідність створення нового програмного продукту, який би забезпечив більш зручну та інтерактивну комунікацію із даним форматом файлів, а також був би орієнтований під більш чітку та конкретну внутрішню схему даних файлів.

Таким чином постає питання розробки певного набору тегів та атрибутів, необхідних для формування та забезпечення деякої стандартизованої структури файлів стандартів під час зберігання та/або переведення їх у форматі XML, а також розробки такого програмного забезпечення, яке допомагало б швидко та зручно переводити стандарти із PDF формату в XML з відповідною базовою структурою.

В роботі запропоновано використовувати наступні правила для структуризації XML-файлів стандартів:

— відкриваючий рядок файлу представлений у форматі <?xml version="1.0" encoding="utf-8" ?>;

— визначається кореневий тег файлу – <standart> (атрибути тегу: code – для коду стандарту, name – для повної назви стандарту);

— тег <header>, що позначає кожен новий заголовок (атрибути тегу: level – рівень в ієрархії, number – порядковий номер, name – назва, nameEng – англійська назва);

— тег <content>, що позначає вміст заголовку;

— тег <picture>, що позначає зображення в тексті (атрибути тегу: number – номер зображення, name – назва зображення);

— тег <definition>, що позначає визначення в тексті (атрибути тегу: name – назва терміну, abbreviation – абревіатура).

Приклад готового XML-файлу стандарту з використанням запропонованих тегів представлено на рис. 1.

```
<?xml version="1.0" encoding="UTF-8"?>
- <standart year="2002" nameEng="Information security, cybersecurity and privacy protection — Evaluation criteria for IT security — Part 2: Security functional components" name="Інформаційна безпека, кібербезпека та захист конфіденційності. Критерії оцінки ІТ-безпеки. Частина 2. Функціональні компоненти безпеки." code="ДСТУ ISO/IEC 15408-2.2">
  - <header nameEng="Foreword" name="Передмова" number="" level="">
      <content> ISO (Міжнародна організація зі стандартизації) та IEC (Міжнародна електротехнічна комісія) утворюють спеціалізовану систему всесвітньої стандартизації. Національні органи, які є членами ISO або IEC, беруть участь у розробці міжнародних стандартів через технічні комітети, створені відповідною організацією для роботи з певними сферами технічної діяльності. Технічні комітети ISO та IEC співпрацюють у сферах взаємного інтересу. </content>
  </header>
  - <header nameEng="Paradigm of functional requirements" name="Парадигма функціональних вимог" number="1.3" level="2">
    - <content>
        На рисунках 1.1 та 1.2 показано деякі ключові поняття парадигми. Описані й інші, які не показані на малюнках, ключові поняття. Ключові поняття, що розглядаються
        <picture nameEng="Key concepts of functional security requirements (single OO)" name="Ключові поняття функціональних вимог безпеки (єдиний ОО)" number="1.1"> ISO_15408_pic_1_1.jpg
        </picture>
        Рисунок 1.1 - Ключові поняття функціональних вимог безпеки (єдиний ОО)
        <picture nameEng="Security features in a distributed assessment facility" name="Функції безпеки в розподіленому ОО" number="1.2"> ISO_15408_pic_1_2.jpg </picture>
        Рисунок 1.2 - Функції безпеки в розподіленому ОО виділено напівжирним курсивом. Визначення термінів, наведені у словнику в розділі 2 ДСТУ ISO/IEC 15408-1, в цьому підрозділі не змінюються і не перевизначаються. Цей стандарт містить каталог функціональних вимог безпеки, які можуть бути пред'явлені до об'єкта оцінки (ОО).
        <definition name="Об'єкт оцінки" URL="" abbreviation="ОО"> Об'єкт оцінки (ОО) — це продукт або система ІТ (разом з керівництвом адміністратора та користувача), що містять ресурси типу електронних носіїв даних (таких як диски), периферійних пристроїв (таких як принтери) та обчислювальних можливостей (таких, як процесорний час), які можуть використовуватися для обробки та зберігання інформації та є предметом оцінки. </definition>
```

**Рис. 1.** Приклад XML-документу

Проте робота із XML-файлами може бути зручною на програмному рівні, але для користувача вони є важкими для читання та розуміння. Таким чином постає необхідність розробки такого програмного продукту, який би допоміг відображати дані файли у максимально зручному та зрозумілому для користувача вигляді. А також щоб він забезпечити не тільки можливість перегляду файлу, а й деякі найнеобхідніші функції взаємодії із вмістом файлу.

Такий програмний продукт представляє собою програму-ґелпер, що створюється як спеціальний допоміжний додаток для основної програми – тієї, що займається переведенням стандартів у формалізований XML формат.

Терміни «допоміжна програма» і «додаток-ґелпер» (або «програма-ґелпер») відносяться до програмного забезпечення, яке призначене для користувача і може читати та обробляти файли конкретного формату. Компанія Netscape використовує термін «допоміжна програма», щоб описати таке програмне забезпечення, тоді як Microsoft Internet Explorer використовує термін «переглядач» [5].

Основним завданням такого додатку є полегшення для користувача роботи із стандартами кібербезпеки. Додаток складається з двох незалежних частин. В першій частині користувачеві надається можливість форматувати файл стандарту у XML формат використовуючи запропоновані в роботі теги. Друга частина працює з файлами XML зазначеної структури.

Завдяки роботі із XML-файлами, додаток має змогу аналізувати не увесь текст, а лише певні теги, що значно полегшує його роботу. Отже, додаток повинен виконувати наступні задачі:

— форматувати файл стандарту у XML формат використовуючи запропоновані в роботі теги;

— читання тексту стандартів та перегляду інших його складових;

— формування ієрархічного змісту стандарту (що в даному випадку базується на атрибутах, а не тегах, а тому звичайні XML-парсери та XML-в'юєри не спроможні вивести його у необхідному вигляді);

— формування списку наявних в стандарті зображень;

— забезпечення можливості пошуку визначень у обраному стандарті;

— забезпечення можливості пошуку визначення в усіх наявних стандартах;

— забезпечення можливості пошуку за словом або декількома словами.

Додаток розроблено з використанням середовища Visual Studio, мови програмування C# та технології інтерфейсу Windows Forms. Це дозволяє створити настільний додаток, який надає максимальний комфорт при використанні в операційній системі Windows. Крім того, додаток має зрозумілий інтерфейс, що забезпечує простоту використання та доступ до необхідних функцій.

Розглянемо детальніше роботу з кожною частиною розробленого додатку.

В першій частині користувачеві надається можливість форматувати файл стандарту. Після запуска цієї частини з'явиться вікно, як продемонстровано на рис.2.

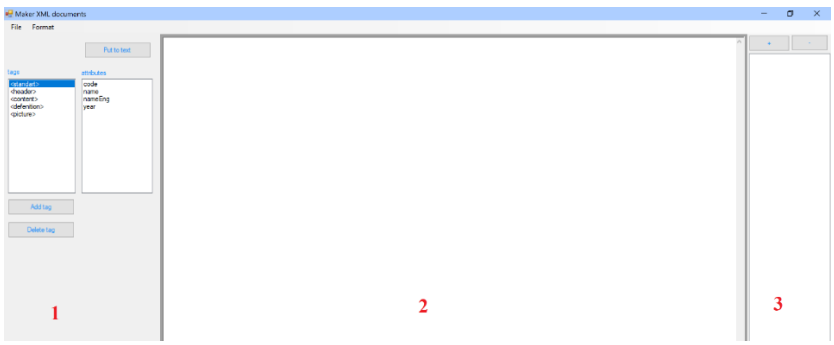



**Рис. 2.** Вікно програми для форматування файлу стандарту

Вікно складається із трьох панелей. На першій панелі розміщені списки для вибору XML-тегу, який вставлятиметься в текст стандарту, а також представлені можливі атрибути вибраного тегу. На другій панелі розташоване текстове вікно, в якому відображатиметься текст стандарту. У цьому вікні можна видаляти, додавати та змінювати елементи тексту. На третій панелі з'явиться список зображень, якщо у вибраному стандарті кібербезпеки є малюнки і вони не закриті від копіювання.

Процедура додавання тегу та його атрибутів передбачає виділення потрібного тегу, виділення частини тексту, що відноситься до цього тегу та натискання кнопки «Put to text». В результаті з'явиться вікно в якому вже додано обраний тег та надається можливість визначити з тексту, що є атрибутами обраного тегу (рис. 3).

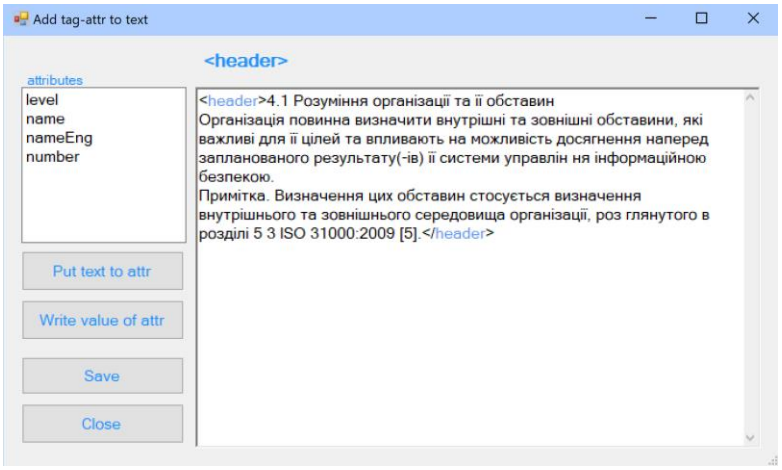



**Рис. 3.** Додавання атрибутів до обраного тегу

Після запуску другої частини у вікні відображається ієрархічне древо змісту стандарту та список зображень, а також повідомлення про наявність можливих небазових тегів. Дана вкладка також дозволяє повністю переглядати вміст стандарту у комфортному та зрозумілому для користувача вигляді (рис. 4).

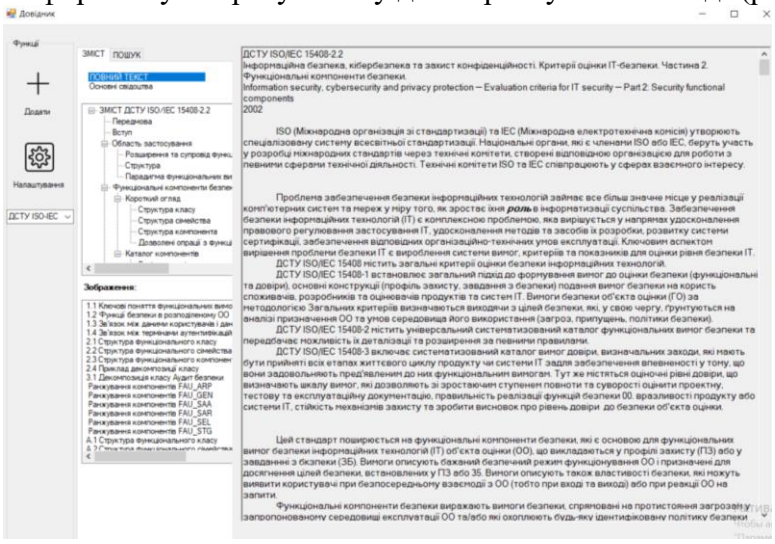


**Рис. 4.** Вікно для перегляду змісту стандарту

На етапі пошуку програма дозволяє проводити пошук необхідної інформації, базуючись в першу чергу на тегах, що містять визначення. Тобто абсолютно вся інформація, якою оперує даний додаток отримується саме із тегів вхідних XML-файлів. Визначення в стандартах є одними із ключових елементів розуміння усього його змісту, тому легкий їх пошук стає важливою задачею. Також забезпечення можливості пошуку визначення одного і того ж терміну у різних

стандартах, стає важливим етапом у, наприклад, проведенні порівняльного аналізу стандартів та вибору найбільш підходящого для тих чи інших цілей. Забезпечення ж пошуку за словом також є необхідною функцією, що значно спрощує орієнтування в додатку та пошуку необхідного місця в тексті за доволі короткий строк – значно швидший за ручне гортання тексту, особливо у великих файлах.

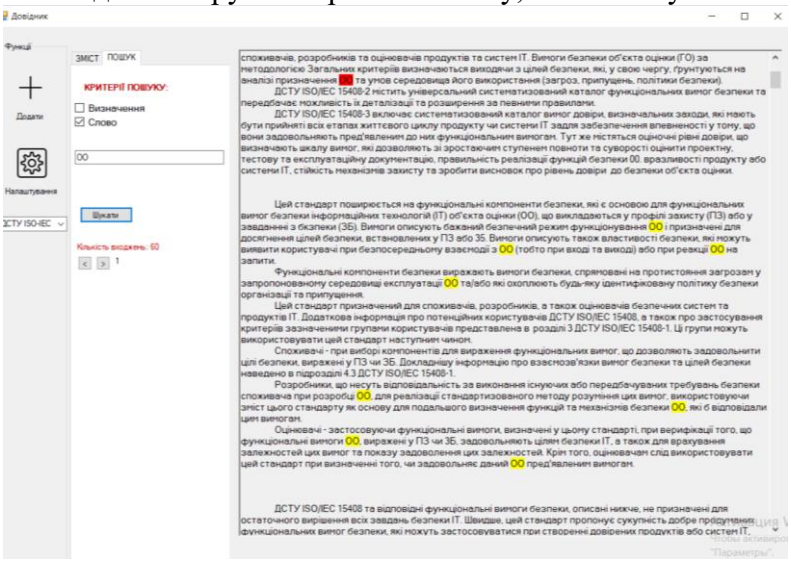



**Рис. 5.** Вікно для пошуку інформації в тексті стандарту

Також в додатку передбачені реакції на різні помилкові дії користувача, такі як неправильні натискання на позиції, помилково зазначені дані тощо – це робить даний додаток-помічник дружнім до користувача.

Отже, тандем із даних двох програмних продуктів відкриває новий спосіб взаємодії із стандартами кібербезпеки, як на програмному рівні (тобто на етапі їх зберігання), так і на рівні користувача. Таким чином виконуються одразу дві масштабні задачі – зберігання стандартів у більш зручному вигляді, що також забезпечує економію пам'яті на швидку роботу програми; а також забезпечення зручного та зрозумілого для користувача способу взаємодії із даними файлами стандартів. Дані програмні продукти можуть в перспективі популяризувати процедуру стандартизації в Україні, а також поширити обізнаність а даній сфері, в тому числі при використанні їх в навчальному процесі та під час підготовки до іспитів.

## Список літератури

1. International Diffusion of the Information Security Management System Standard ISO/IEC 27001: Exploring the Role of Culture. URL: https://aisel.aisnet.org/ecis2020_rp/88
2. ISO (International Organization for Standardization). URL:https://www.iso.org/about-us.html
3. XML (Extensible Markup Language) URL: https://www.techtarget.com/whatis/definition/XML-Extensible-Markup-Language
4. XML Files: What They Are & How to Open Them URL: https://blog.hubspot.com/website/what-is-xml-file
5. What Are Helper Applications and Viewers? URL: https://www.microfocus.com/documentation/extend-acucobol/925/BKPIPIVIEWS001.html

О.О. Лановська, О.Ю. Лебедєва

# DEVELOPMENT OF A SOFTWARE APPLICATION FOR WORKING WITH CYBER SECURITY STANDARDS

O. Lanovska, O. Lebedeva

National Odesa Polytechnic University

1, Shevchenko Ave., Odesa, 65044, Ukraine;
e-mails: lanovska.8088987@stud.op.edu.ua, o.y.lebedieva@op.edu.ua

The work developed a software application aimed at working with cyber security standards and XML format files. The goal of this software application is to simplify the process of working with cybersecurity standards in XML format. The application will have an intuitive interface that will allow users to conveniently create, edit and view XML files related to cyber security. The International Organization for Standardization (ISO) develops and publishes standards that include recommendations and best practices for cybersecurity. These standards reflect global norms and best practices in the field of cyber security and are widely used in business and other sectors to ensure safety on the Internet. The main tasks of such a software product are to ensure comfortable and convenient interaction of the user with the content of cyber security standards, to create a platform for searching for the definitions required by the user in the standard and to interact with the text, as well as to provide the possibility of forming XML files according to a defined scheme based on existing files of cyber security standards in PDF format XML is a widespread format for storing and transmitting data and is most often used for storing technical documentation, therefore, in the context of storing standards, this format is quite promising and opens up more versatile opportunities for interaction with them. In the work, an analysis of the subject area and modern existing analogues was carried out - as a result of the analysis, it was found that all analogues do not provide the necessary tasks, and a list of tags and attributes was developed for the internal structuring of XML files of cyber security standards. The results of this work can be used during the educational process for the purpose of quick access for both teachers and students to the necessary standards and definitions, as well as comparison of definitions of terms from different standards. The practical value of this software product lies in the possibility of its active use by all participants of the educational process, as well as interested persons.

**Keywords:** cyber security, information security, standard, ISO, XML file, tags, attributes, helper application.

DOI 10.15276/imms.v13.no1-2.119
УДК 004.896

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp. 119-129

# ПІДХІД ДО УСУНЕННЯ КОНФЛІКТІВ У МИЛЬТИАГЕНТНИХ СИСТЕМАХ НА ОСНОВІ АЛГОРИТМУ ДЕЙКСТРИ

В.Г.Пенко, О.В.Пенко, В.В.Коган

Одеський національний університет імені І. І. Мечнікова,
вул.Дворянська,2, м.Одеса, Україна,
e-mails: vpenko@onu.edu.ua , odael.odes@gmail.com, vladislav.kogan@gmail.com

Завдання пошуку агентом маршруту є одним із базових завдань і має велике практичне значення. Як правило, це оптимізаційне завдання, яке вирішується у просторі графа станів з метою мінімізації довжини маршруту. Таке завдання стає складнішим і набуває ще більшої популярності, якщо воно вирішується на основі одного графового простору декількома агентами. В цьому випадку з'являється ще один практично важливий параметр оптимізації, який виражається кількістю конфліктів між агентами. Така абстрактна постановка завдання має численні варіанти формалізації та велику кількість підходів до її вирішення. У цієї роботі основним елементом підходу є застосування класичного алгоритму Дейкстри, що виконується послідовно кожним агентом. Під час знаходження найкоротшого маршруту кожен агент модифікує графовий простір таким чином, щоб зменшити ймовірність конфліктів з наступними агентами. Цей похід є оригінальним і для перевірки був виконаний ряд обчислювальних експериментів за допомогою розробленого для цієї мети програмного забезпечення. Експерименти, що були проведено, демонструють адекватну поведінку основного алгоритму. У роботі проаналізовано обмеження, властиві запропонованому підходу у рамках абстрактної постановки завдання. Визначено напрями подальшого розвитку базового підходу.

**Ключові слова**: кооперативна поведінка агентів, алгоритм Дейкстри, мультиагентні системи, конфлікти.

**Вступ та огляд існуючих підходів.** Розробка та дослідження мультиагентних систем (МАС) є перспективним сучасним науково-практичним напрямом, який можна віднести до категорії міждисциплінарних на стику наступних галузей: інформатика (особливо штучний інтелект), психологія (особливо соціальна психологія) та філософія. Цей список можна розширити, але й у такому вигляді зрозуміло, що від вдалих проектів МАС слід очікувати синергетичного ефекту не тільки завдяки синхронній участі кількох агентів, але й ефекту використання методів цих наукових напрямів, що їх взаємодоповнює.

Мабуть, ключовий фактор, що дозволяє досягти позитивного синергетичного ефекту – це здатність системи організувати взаємодію агентів так, щоб зменшити або навіть мінімізувати конфлікти між ними. В іншому випадку ми ризикуємо отримати зворотний ефект, коли агенти діють непродуктивно, заважаючи один одному. Звернемо увагу, що словосполучення «здатність системи» не має на увазі тут конкретної децентралізованої архітектури системи. Більш того, найбільшого ефекту слід очікувати від систем значною мірою гомогенних.

Область досліджень, якій присвячена дана робота, має досить усталене позначення – Multi-Agent Path Finding (MAPF). З цього погляду поведінка агента зводиться до знаходження маршруту у просторі станів.

Основною метою даної роботи є розробка механізму, який дозволить

знаходити баланс між двома взаємносуперечливими показниками – кількість конфліктів та довжина маршруту. Іншими словами, ми хочемо знайти субоптимальне рішення одночасно для цих двох показників. Незважаючи на присутність показника довжини маршруту, спектр предметних областей не обмежений середовищами, в яких явно присутній метричний простір (одночасне переміщення автомобілів з автопілотами, авіаційних транспортних засобів, роботів, що виконують сервісні функції у приміщенні чи будівлі). При більш загальної інтерпретації предметна область може мати не просторового аспекту і бути системою споживання ресурсів обмеженого обсягу, які агенти використовують рои виконанні своїх завдань. Таким чином, важливим аспектом запропонованого дослідження є розробка методики, що дозволяє перетворювати концепції вихідної предметної галузі у концепції формального походу. Зазначимо також, що в цій роботі ми намагаємося запропонувати підхід, який дозволить уникнути конфлікту, а не усувати його після факту його виникнення. У термінах MAC цей метод значною мірою відноситься до розділу планування [1].

Аналіз джерел, присвячених проблемам MAPF демонструє різноманітність методів їх вирішення. Природно, значна частина цих методів пов'язана з побудовою графа станів для розв'язуваної задачі та застосування різних різновидів алгоритмів пошуку на графах. Для підвищення ефективності часто використовують різновиди алгоритму A*. Наприклад, у роботі [2] пропонується модифікація EPEA*, у якій у процесі пошуку оптимізується кількість вузлів графа, що розкриваються на кожному кроці. У цьому напрямі робота [3] пропонує використовувати дворівневий пошуковий алгоритм, спроектований у тому, щоб перевершити традиційний алгоритм A*. Двохрівнева декомпозиція на думку авторів має забезпечити оптимальність рішення, зберігаючи при цьому прийнятні показники ефективності. Автори стверджують, що запропонований підхід забезпечує оптимальність рішення за прийнятний час у деяких випадках його застосування. Багато з пропонованих у цьому напрямі підходів реалізують багатоагентність у формі групового агента (joint agent), у результаті вузли графового простору подають інформацію фактично про безліч агентів.

Хоча робота [4] також у основі формальної моделі використовує графові алгоритми, але у значно іншій манері. Автори зосередили зусилля на безконфліктному русі транспортних засобів на ділянках із некерованими перехрестями. Математичним підґрунтям цього підходу є алгоритм мінімального розміру клікового покриття графа, а інструментарієм дослідження – імітаційне моделювання з подальшим аналізом результатів експериментів.

Інша методика вирішення проблем MAPF пропонується у [5]. Тут як основу пошукового алгоритму автори використовують генетичний алгоритм. Недоліком генетичного алгоритму, як і інших традиційних методів штучного інтелекту, є його непрозорість, яка не дозволяє користувачеві критично аналізувати отримані рішення, спираючись на зрозумілу логіку алгоритму. З іншого боку, прикладний напрямок запропонованого автором підходу пов'язаний з управлінням авіаперевезеннями (Air Transportation Management - ATM), де такий критичний аналіз є важливим. Тому, крім самого механізму пошуку маршрутів, автори пропонують використовувати методи візуалізації для аналізу рішень.

Поданий тут невеликий перелік підходів не претендує на повноту, але дозволяє зрозуміти, що для вирішення завдання організації ефективної безконфліктної поведінки безлічі агентів застосовуються дуже різноманітні у математичних основах методи. Досить серйозною проблемою є також різноманітність формулювань розв'язуваних задач, що ускладнює порівняльний аналіз різних підходів.

**Базове завдання кооперативної поведінки агентів та основний підхід до її вирішення.** Предметна область даної роботи – мультиагентне середовище, у якому агенти досягають своїх різних цілей, намагаючись мінімізувати взаємні конфлікти. У цій роботі така ситуація не буде формалізована таким чином, щоб формулювання виявилося би певною мірою універсальним. Спроби зробити таке формулювання міститься в [6]. Однак зворотною стороною такої універсальності є складність здійснити на її основі реалізацію представницької множини корисних прикладних випадків.

Розглянемо наступну версію постановки завдання. Деякі агенти існують у просторі, заданому орієнтованим зваженим графом. Вагу дуги називатимемо довжиною дуги. Кожен із агентів на початку знаходиться в одному із вузлів графа (стартовий вузол). Наприкінці процесу агент повинен опинитися у вузлі, який називатимемо цільовим вузлом. Може існувати кілька маршрутів, які приводять агента зі стартового вузла до цільової. Кількість маршрутів може бути навіть нескінченною через наявність циклів у графі. Конфліктом між двома агентами у початковій постановці завдання є присутність у їх маршрутах однакової дуги. Кількість конфліктів – це кількість таких збігів. Метою завдання є знаходження таких маршрутів для всіх агентів, які одночасно мінімізують довжини маршрутів та кількість конфліктів для всієї сукупності агентів. Не завжди вдається досягти нульової кількості конфліктів за мінімізації довжини маршрутів. І тут рішення приймається виходячи з балансу між цими суперечливими показниками у конкретній задачі.

У такій постановці завдання немає поняття часу. Тому поняття конфлікту визначено безвідносно послідовності дуг у маршруті. Таку постановку завдання далі називатиме базовим завданням кооперативної поведінки.

У такому трактуванні завдання його не можна безпосередньо сприймати як модель руху матеріальних об'єктів у метричному просторі. Важливим напрямом розвитку пропонованого підходу є його розширення до більш реалістичних варіантів завдань.

Кількість можливих маршрутів для одного агента в сукупності з необхідністю проаналізувати ці маршрути для багатьох агентів, унеможливлює рішення базового завдання навіть для невеликого графового простору і кількості агентів.

Напрями наукових досліджень, у яких можна шукати методи розв'язання базової задачі: по-перше, це теорія машинного навчання з підкріпленням як у загальному вигляді [7], так і її похідна підобласть – мультиагентне навчання з підкріпленням [8], по друге це теорія ігор, в якій досліджуються особливості взаємодії агентів з різними цілями, які не завжди збігаються[9].

Ми використовуємо у певному сенсі зворотний підхід. Взявши за основу гіпотезу про застосування алгоритму Дейкстри в умовах постановки завдання, уточнюватимемо, як та чи інша модифікація цього алгоритму може дозволити вирішити відповідний різновид завдання.

У цій роботі передбачається розуміння канонічного алгоритму Дейкстри, опис якого можна знайти у різних джерелах [10]. З погляду базового завдання досить знати, що алгоритм Дейкстри знаходить найкоротший маршрут між двома вузлами графа і має обчислювальну складність O (nodes * log nodes + edges * log edges) для розріджених графів. Тепер уявімо, що в цьому ж графовому просторі є ще кілька таких агентів, які вирішують таке ж завдання, але з іншими початковими та кінцевими пунктами. Якщо вирішити завдання кожного з них незалежно один від одного, може виявитися, що рішення (найкоротші маршрути) мають спільні дуги. На рисунку 1 показано найкоротші маршрути, знайдені

алгоритмом Дейкстри для двох агентів незалежно один від одного. Тут sa1 та sa2 – стартові точки агентів, а fa1 та fa2 – їх кінцеві точки. Дуга від вузла a до вузла b використовується в обох маршрутах.

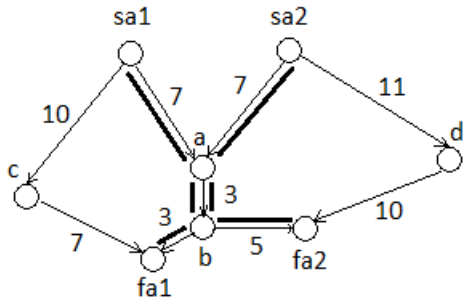



**Рис. 1.** Результати незалежного використання алгоритму для двох агентів.

Це не означає неминуче, що агент 1 і агент 2 виявляться під час своєї подорожі одночасно на цих дугах. Однак якщо не спробувати детально відстежувати процес переміщення агентів у часі, факт спільного використання дуг у кількох найкоротших маршрутах загалом підвищує ймовірність конфлікту у формі одночасного використання ділянки шляху (дуги).

Існують спроби такого відстеження. Зокрема, можна відзначити публікацію Д. Сільвера [11]. Однак такий підхід досить ресурсомісткий, оскільки означає підвищення розмірності розв'язуваного завдання.

Звідси випливає формулювання завдання як зниження ймовірності такого роду конфліктів кількома агентами.

Основна ідея досить проста та інтуїтивно сприймається як адекватна. Дамо можливість агенту 1 побудувати свій маршрут звичайним чином. Після цього додамо до ваги всіх використаних у маршруті дуг деяку величину збільшення. Після цього на графі, що змінився, своє завдання вирішує агент 2. Якщо є наступний агент, то знову відбувається збільшення ваг дуг і т.д.

Тепер після збільшення ваг дуг логіка алгоритму Дейкстри для агента 2 з меншою ймовірністю побудує маршрут, що використовує більш «важкі» дуги. Наприклад, задача, представлена на рисунку 2 із застосуванням збільшення ваги на 10 призведе до того, що агент 2 вибере маршрут, що не має загальних дуг з маршрутом агента 1.

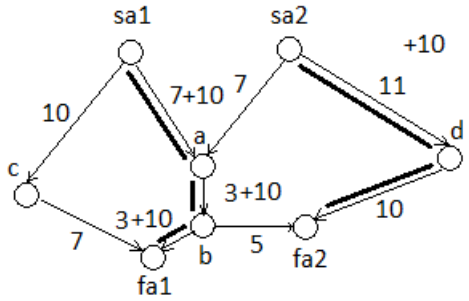



**Рис. 2.** Маршрути двох агентів із використанням збільшення ваги.

Цей приклад демонструє роль величини збільшення ваги дуги. Якби у наведеному прикладі збільшення було б менше, наприклад 5, маршрут агента 2 зберігся б таким же, як на рис. 1, тобто мав би місце конфлікт спільного використання дуги. Це означає, що одним із важливих аспектів подальшого дослідження є розробка механізму вибору ефективного значення величини збільшення.

Описаний вище прийом застосування алгоритму Дейкстри для знаходження найкоротшого шляху з одночасною мінімізацією конфліктів далі

називатимемо базовим підходом. Така назва обумовлена тим, що вже зараз проглядаються модифікації цього підходу для подолання деяких властивих йому обмежень.

Для експериментальної перевірки базового походу розроблено відповідне ПЗ.

План обчислювального експерименту має такий вигляд.

1. Генеруємо граф простору.
2. Розміщуємо кілька агентів у різних стартових вузлах та призначаємо їм цільові вузли.
3. Будуємо для кожного агента найкоротший маршрут без зміни дуг графів
4. Обчислюємо деякі узагальнені показники, наприклад, сумарну довжину маршрутів всіх агентів.
5. Будуємо для кожного агента найкоротший маршрут, виконуючи описану вище модифікацію дуг графів
6. Обчислюємо деякі узагальнені показники, наприклад, сумарну довжину маршрутів всіх агентів.
7. Порівнюємо параметри, обчислені на кроці 4 та 6.

Мета експерименту – визначити умови, у яких метод збільшення ваг дуг дає адекватні для предметної області результати.

Цей план насправді узагальнений і може мати модифікації, що мають різні прикладні інтерпретації.

Найбільш природним видається випадок, коли:

A) Граф досить поступово покриває евклідовий простір. Це може бути, наприклад, прямокутна сітка із досить невеликим розміром клітин.

B) Стартові точки агентів перебувають у відносній близькості один від одного.

C) Кінцеві точки агентів знаходяться у відносній близькості один від одного.

Дамо змістовну інтерпретацію цих умов.

Умова A) в екстремальному вигляді означає визначення простору можливих переміщень у вигляді дрібнозернистої сітки із квадратних клітин. Неформально це означає велику свободу вибору напряму руху, що дозволяє уникати більшості конфліктів. Однак на практиці таке середовище не становить великого інтересу, оскільки воно не містить перешкод і обмежень, що задаються штучно створеною і не завжди регулярною структурою графа.

Умови B) та C) означають близькість загального напрямку руху агентів, що призводить до збільшення потенційних конфліктів.

Для проведення описаних вище експериментів було розроблено відповідні програмні компоненти.

**Опис обчислювальних експериментів.** Уточнимо умови першого експерименту. Простір переміщень агентів задається зваженим орієнтованим графом, схематично представленим на рисунку 3. (фактично розмір графа експериментально був 10x10). Довжини (ваги) вертикальних та горизонтальних дуг дорівнюють 1, а довжини діагоналей $\sqrt{2}$, тобто моделюється стандартна евклідова сітка.

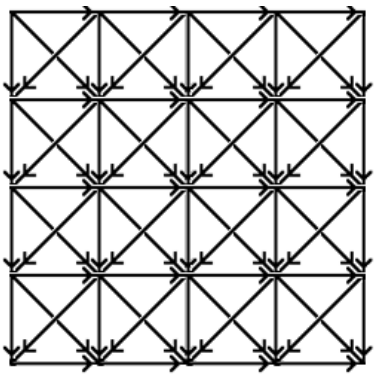


**Рис.3.** Граф, що використовується в експериментах (фактично 10x10).

У першій серії експериментів 10 агентів розміщувалися в лівому верхньому кутку графа, і всі вони прагнули потрапити у нижній правий кут.

Кожен експеримент у серії вирізнявся своїм значенням дельта ваги. У таблиці 1 показано, як із зміною величини збільшення ваги змінювалася сумарна відстань, пройдене агентами.

**Таблиця 1**

Сумарні довжини маршрутів за різних приріст ваги.

| Дельта ваги | Сумарна відстань |
|---|---|
| 0 | 127.279 |
| 0.2 | 144.880 |
| 0.4 | 152.994 |
| 0.6 | 159.923 |
| 0.8 | 165.923 |
| 1 | 171.923 |

Отже, спостерігаємо очікувану закономірність. При зростанні збільшення ваги сумарна відстань поступово зростає, оскільки агенти стають дедалі обережнішими, намагаючись уникати вже використані дуги.

Цікавим є кількість конфліктів, тобто спільно використовуваних агентами дуг графа. Однак цей показник буде мало репрезентативним, якщо ці конфлікти можуть відбуватися у різні моменти модельного часу.

Також програма, що моделює поведінку агентів, показує докладнішу інформацію про маршрути агентів таким чином:

Agent 0, best path 0:0->1:1->2:2->3:3->4:4->5:5->6:6->7:7->8:8->9:9.
Agent 1, best path 0:0->0:1->1:2->2:3->3:4->4:5->5:6->6:7->7:8->8:9->9:9.
Agent 2, best path 0:0->1:0->2:1->3:2->4:3->5:4->6:5->7:6->8:7->9:8->9:9.
Agent 3, best path 0:0->0:1->0:2->1:3->2:4->3:5->4:6->5:7->6:8->7:8->8:8->9:9.
Agent 4, best path 0:0->1:0->2:0->3:1->4:2->5:3->6:4->7:5->8:6->9:7->9:8->9:9.
Agent 5, best path 0:0->1:1->1:2->1:3->1:4->2:5->3:6->4:7->5:8->6:8->7:9->8:9->9:9.
Agent 6, best path 0:0->1:1->2:1->3:1->4:1->5:2->6:3->7:4->8:5->8:6->8:7->8:8->9:9.
Agent 7, best path 0:0->0:1->0:2->0:3->0:4->1:5->2:6->3:7->4:8->5:8->6:9->7:9->8:9->9:9.
Agent 8, best path 0:0->1:0->2:0->3:0->4:0->5:1->6:2->7:3->8:4->8:5->9:6->9:7->9:8->9:9.
Агент 9, best path 0:0->1:1->2:2->3:3->4:4->5:5->6:6->7:7->8:8->9:9.

Ця форма не зручна для аналізу, оскільки конфлікти не видно явно (тут вони для зручності виділені підкресленням), тому була виконана візуалізація маршрутів агентів. На рисунку 4 показано, як виглядають маршрути агентів при

різних збільшення ваги. Конфлікти показані за допомогою ліній, що повторюються, що з'єднують вузли.

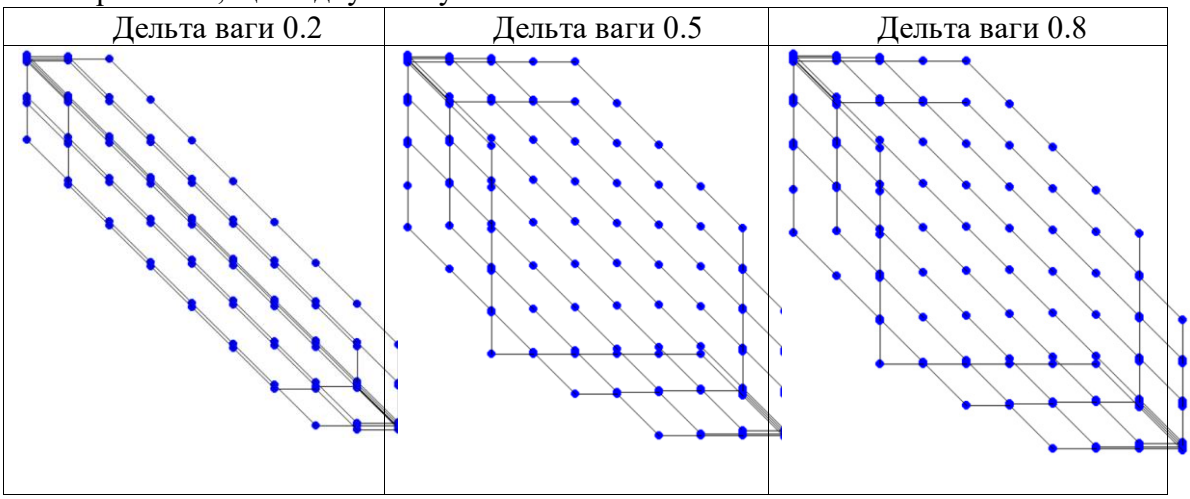
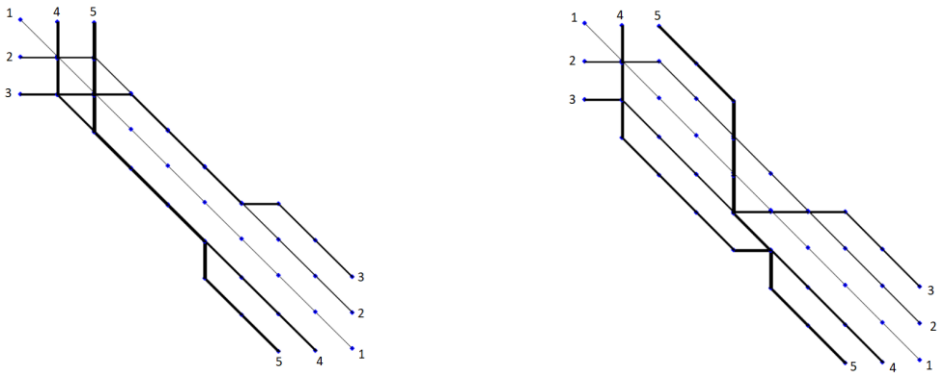



**Рис. 4.** Візуалізація кількох варіантів знаходження агентами найкоротших маршрутів при різних збільшеннях ваги.

У першій серії експериментів усі агенти мали однакові стартові та цільові вузли. Це зумовлює досить часті конфлікти у вигляді спільного використання дуг. При підвищенні величини збільшення ваги дуг агенти змушені ширше використовувати простір пошуку, що виглядає інтуїтивно природно.

Наступний експеримент, на наш погляд, демонструє ще цікавішу поведінку запропонованого механізму. Тепер ми розміщуємо кілька агентів у різних стартових вузлах неподалік один від одного (в околиці лівого верхнього кута), а цільові вузли задаємо різними, але в околиці правого нижнього кута. Більш того, конкретні вузли ми вказуємо так, щоб змусити перетинатися маршрути різних клієнтів. Результати деяких експериментів показано рисунку 5. Для зручності візуального відстеження маршрути різних агентів показані різною товщиною ліній. Тут бачимо, що без використання механізму збільшення ваги маршрути агентів перетинаються у великій кількості дуг. Застосування дельта ваги із значенням 0.8 дає якісно іншу картину. Маршрути агентів жодного разу не перетнулися! Агенти знайшли неочевидні маршрути, щоб уникнути конфліктів. При цьому сумарна пройдена відстань збільшилася незначно (59.255 проти 58.669)




Загальний шлях 58.66904755. Дельта ваги 0.0 Загальний шлях 59.25483399. Дельта ваги 0.8

**Рис. 5.** Візуалізація поведінки агентів, що розташовані в околиці один від одного і мають різні цільові вузли

**Порядок використання агентів у процесі застосування базового підходу.** Ще одна особливість базового підходу полягає в тому, що його результати залежить від порядку використання в ньому агентів. У деяких випадках цей порядок не можна змінити, оскільки він є наслідком особливостей предметної області. Наприклад, якщо порядок використання агентів пов'язаний із деяким поняттям пріоритету їхньої участі. Якщо ж послідовність агентів є зовнішнім чинником, нав'язаним алгоритму, можна застосувати деяку процедуру аналізу різних варіантів послідовностей участі агентів в алгоритмі.

Наступний експеримент відрізняється від попереднього тим, що відстані між вузлами (ваги дуг) тепер є випадковими, які в деякій мірі відрізняються від відстаней у правильній квадратній сітці евклідового простору. Це зроблено для того, щоб збільшити різноманітність варіантів конфігурацій маршрутів. Розмір збільшення ваги 0.2. Експеримент із 5 агентами допускає всього 5! варіантів послідовностей агентів у алгоритмі. Результати такого повного комбінаторного експерименту показано у таблиці 2.

**Таблиця 2**

Довжини маршрутів за різних послідовностей участі агентів.

| Група мар-шрутів | Реальна відстан | Ефективна відстан | Кіл-ть мар-шрутів | Група мар-шрутів | Реальна відстан | Ефективна відстан | Кіл-ть мар-шрутів |
|---|---|---|---|---|---|---|---|
| 1 | 56.6685 | 57.2685 | 4 | 13 | 56.8140 | 57.1140 | 2 |
| 2 | 56.6811 | 57.1811 | 12 | 14 | 56.8284 | 57.1284 | 4 |
| 3 | 56.6954 | 57.1954 | 29 | 15 | 56.8810 | 57.0810 | 1 |
| 4 | 56.7329 | 57.2329 | 10 | 16 | 56.8962 | 57.1962 | 2 |
| 5 | 56.7387 | 57.2387 | 4 | 17 | 56.9314 | 57.3314 | 4 |
| 6 | 56.7438 | 57.1438 | 8 | 18 | 56.9355 | 57.1355 | 4 |
| 7 | 56.7582 | 57.1582 | 14 | 19 | 56.9427 | 57.3427 | 3 |
| 8 | 56.7604 | 57.3604 | 1 | 20 | 56.9622 | 57.1622 | 1 |
| 9 | 56.7611 | 57.3611 | 1 | 21 | 56.9894 | 57.2894 | 3 |
| 10 | 56.7854 | 57.2854 | 1 | 22 | 57.0058 | 57.1058 | 3 |
| 11 | 56.8031 | 57.2031 | 4 | 23 | 57.0140 | 57.1140 | 2 |
| 12 | 56.8078 | 57.4078 | 12 | 24 | 57.1766 | 57.4766 | 2 |

Ефективна відстань, що фігурує в таблиці 2, означає довжину маршруту агента після того, як деякі дуги в графі були збільшені в ході виконання алгоритму. Хоча ця відстань не є частиною опису початкового стану світу, у деяких випадках вона може виявитися корисною характеристикою.

Перша група з 4-х послідовностей агентів демонструє найкращий показник реального шляху 56.6685 за рахунок збільшення довжин деяких дуг (ефективна відстань – 57.2685). Цікаво, що наступна група послідовностей дає дещо більшу реальну довжину 56.6811, але меншу ефективну (57.1811). Це означає, що у другій групі було менше конфліктів. Який із варіантів слід вважати найкращим, залежить від балансу між цінністю коротких маршрутів та мінімізацією конфліктів у конкретній предметній галузі.

Деякі результати для різних послідовностей візуально показані на рисунку 6.

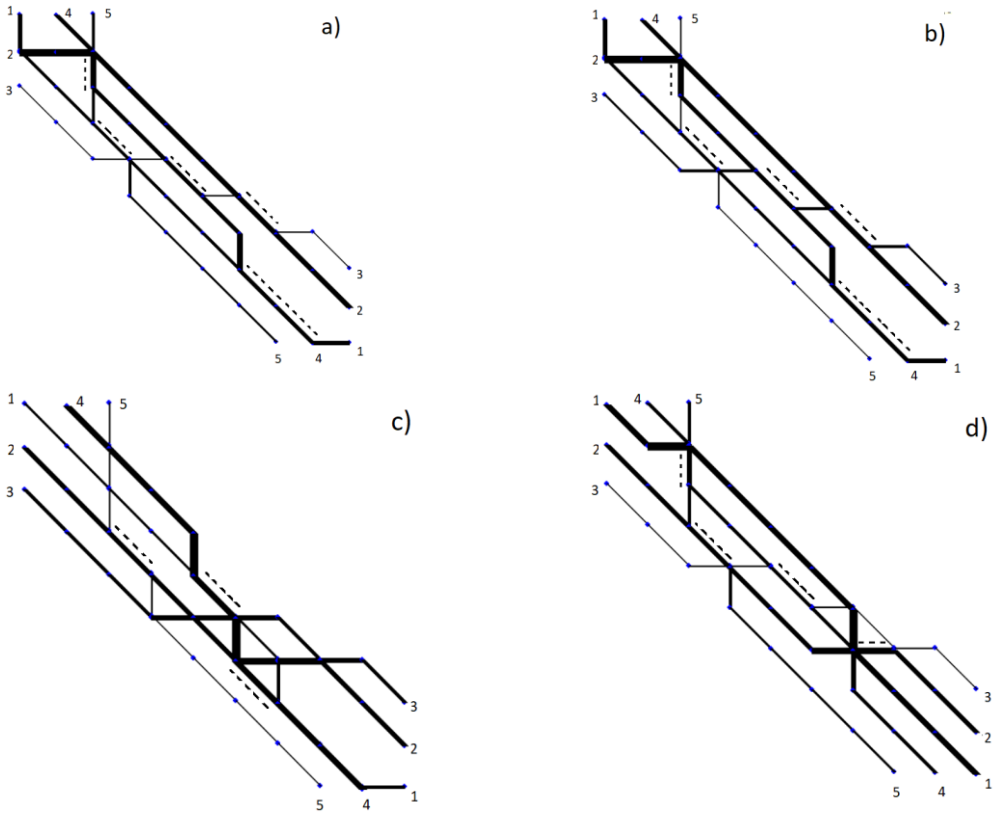



**Рис. 6.** Конфігурації маршрутів 5 агентів за різного порядку їхньої участі в алгоритмі (пунктиром показані конфлікти).

На рисунку 6 a) та рис. 6 b) показані результати роботи алгоритму, отримані для послідовностей агентів (3, 5, 1, 4, 2) (3, 5, 2, 4, 1). Це забезпечило найменшу загальну довжину маршрутів, однакову для двох послідовностей – 56.6684. Видно, що, незважаючи на різні послідовності агентів, їх маршрути в обох випадках однакові. Загальна кількість конфліктів (спільно використовуваних дуг) – 6 (показані пунктиром).

Рисунок 6 c) показує результат алгоритму для послідовностей агентів (2, 4, 3, 5, 1) (2, 4, 5, 3, 1) (2, 5, 3, 4, 1) (2, 5, 4, 3 1). У всіх цих випадках загальна відстань – 56.9314, що трохи гірше для найкращих послідовностей. Кількість конфліктів – 3. Це означає, що алгоритм зумів забезпечити меншу кількість конфліктів за незначного збільшення відстані. Який із варіантів вибирати залежить від того, що важливіше у конкретному випадку – мінімізація конфліктів чи відстані.

Рисунок 6 д) показує результат алгоритму для послідовностей агентів (4, 3, 0, 2, 1) та (4, 3, 1, 2, 0). Кількість конфліктів – 4. У цьому випадку загальна відстань - 57.1766. Ці послідовності програють за всіма показниками попереднім послідовностям і можуть бути відкинуті у процесі комбінаторного аналізу.

Таким чином, при невеликій кількості агентів можна провести комбінаторно вичерпний експеримент, знайшовши найкращі варіанти послідовностей агентів.

**Обмеження базового підходу.** Слід розуміти, що наведений вище підхід має суттєві обмеження. Це, проте, не означає його принципову прикладну обмеженість. Вже зараз проглядаються модифікації базового походу, які нівелюють деякі обмеження. У зв'язку з цим важливо сформулювати ці обмеження.

По-перше, простір прийняття рішень представлено у вигляді виваженого орієнтованого графа. Це означає, що найбільш очевидні застосування базового походу пов'язані з переміщення агентів в евклідовому просторі. Однак можлива узагальнена інтерпретація ваги дуги, що дозволяє застосовувати базовий похід для відповідних завдань без його модифікації.

Друге обмеження має на увазі, що поведінка агента в процесі проходження ним маршруту у графі здійснюється без урахування часового аспекту. Це припустимо, наприклад, коли час подолання маршрутів усіма агентами дуже малий. Умовно всі агенти долають маршрут миттєво. Ця особливість дійсно спостерігається в деяких випадках, навіть коли агенти діють у евклідовому просторі. Крім того, є предметні галузі, де відсутність тимчасового аспекту є природною. Наприклад, завдання розподілу обмеженої кількості ресурсів, необхідні виконання завдання перед її виконанням.

Третє обмеження – після того, як базовий підхід визначить для агента маршрут, цей маршрут не може бути змінений до того, як усі агенти не закінчать подорожі.

Останнє обмеження у тому, що конфліктом вважається спільне використання агентами дуг графа. У деяких випадках більш природною інтерпретацією конфлікту може бути одночасне знаходження у вузлах графа або деяке поєднання цих ситуацій.

**Висновки та перспективи.** Незважаючи на перелічені обмеження, базовий похід демонструє перспективну поведінку у процесі вирішення базового завдання MAPF. Результати описаних вище експериментів демонструють здатність знаходити неочевидні, але цілком адекватні маршрути для сукупності агентів.

Базовий підхід характерний наявністю основного гіперпараметра – величини збільшення ваги дуги. Зміна цього значення дозволяє знаходити під час обчислювального експерименту відповідний компроміс між довжиною маршруту та кількістю конфліктів.

Перспективи та напрямки розвитку базового підходу:

- модифікація поняття конфлікту, що дозволяє уникати перебування кількох агентів у вузлі;

- облік чинника часу, що дозволить перебування кількох агентів на дузі чи вузлі, якщо це відбувається у різні моменти часу;

- розробка автоматизованої процедури трансформації проблем з конкретних предметних областей у терміни базового завдання, тобто перетворення предметної області конкретної задачі на графовий простір з вагами дуг та вершин, що відповідають семантиці задачі.

**Список літератури**

1. Russell S., Norvig P. Artificial intelligence: a modern approach. Pearson, 2021. 1152 p.
2. Felner A., Goldenberg M., Sharon G., Stern R., Beja T., Sturtevant N., Schaeffer J., Holte R. Partial-Expansion A* with Selective Node Generation. *Twenty-Sixth AAAI Conference on Artificial Intelligence*. 2012. Vol.26. No.1
3. Sharon G., Stern R., Felner, Sturtevant N. (). Conflict-based search for optimal multi-agent pathfinding. *Twenty-Sixth AAAI Conference on Artificial Intelligence*. 2012.Vol.26. No.1
4. Chen C., Xu Q., Cai M., Wang J., Xu B., Li K. Conflict-free Cooperation Method for Connected and Automated Vehicles at Unsignalized Intersections: Graph-based Modeling and Optimality Analysis. *IEEE Transactions in Intelligent Transportation Systems*. 2022. Vol.14. No.8.

5. Hurter C., Dega A., Guibert N. Usage of more transparent and explainable conflict resolution algorithm: air traffic controller feedback. *34th Conference of the European Association for Aviation Psychology Pages*. 2022. P.270–278
6. Тарасов В.Б. От многоагентных систем к интеллектуальным организациям: философия, психология, информатика. М.: Эдиториал УРСС, 2002. 352 с.
7. Sutton R., Barto A. Reinforcement Learning: An Introduction. Massachussets: MIT Press, 2018. 548 p.
8. Busoniu L., Babuska R., De Schutter B., Multi-agent reinforcement learning: A survey. *Proceedings of the 9th International Conference on Control, Automation, Robotics and Vision (ICARCV 2006)*. Singapore, 2006. P.527–532.
9. Shoham Y., Leyton-Brown K. Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge University Press, 2008. 532 p.
10. Dijkstra E.W. A note on two problems in connexion with graphs. *Numer. Math. Springer Science+Business Media*. 1959. Vol.1, Iss.1. P.269–271.
11. Silver D. Cooperative Pathfinding. *Proceedings of the First Conference on Artificial Intelligence and Interactive Digital Entertainment*. 2005. P.117-122.

## AN APPROACH TO THE ELIMINATION OF CONFLICTS IN MULTI-AGENT SYSTEMS BASED ON DIJKSTRA'S ALGORITHM

V.G.Penko, O.V.Penko, V.V.Kogan

Odesa National University named after I. I. Mechnikov, 2 Dvoryanska St.,
Odesa, Ukraine,
e-mails: vpenko@onu.edu.ua, odael.odes@gmail.com, vladislav.kogan@gmail.com

The task of path finding by an agent is one of the basic tasks and is of great practical importance. As a rule, this is an optimization task that is solved in the space of the state graph in order to minimize the length of the path. Such a task becomes more difficult and becomes even more popular if it is solved on the basis of one graph space by several agents. In this case, another practically important optimization parameter appears, which is expressed by the number of conflicts between agents. Such an abstract formulation of the task has numerous variants of formalization and a large number of approaches to its solution. In this paper, the main element of the approach is the application of the classic Dijkstra algorithm, which is executed sequentially by each agent. When finding the shortest route, each agent modifies the graph space in such a way as to reduce the probability of conflicts with subsequent agents. This approach is original and a number of computational experiments were performed to verify it using software developed for this purpose. The experiments that were conducted demonstrate the adequate behavior of the main algorithm. The work analyzes the limitations inherent in the proposed approach within the framework of the abstract formulation of the task. The directions of further development of the basic approach have been determined.

**Keywords**: cooperative behavior of agents, Dijkstra's algorithm, multiagent systems, conflicts.

DOI 10.15276/imms.v13.no1-2.130
УДК 681.514:621.029

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp.130-136

# ВИЯВЛЕННЯ МУЛЬТИПЛІКАТИВНОГО ШУМУ В ЦИФРОВИХ ЗОБРАЖЕННЯХ В УМОВАХ ЗБЕРЕЖЕННЯ З ВТРАТАМИ

К.О.Петрук, В.В.Зоріло, О.Ю.Лебедєва

Національний університет «Одеська політехніка»
пр.Шевченка, 1, Одеса, 65044
email: vikazorilo@gmail.com

Методи виявлення порушень цілісності цифрової інформації (цифрових фото-, відео-, та аудіо-файлів) відіграють важливу роль для інформаційної та кібербезпеки. Серед порушень цілісності цифрових зображень окремою категорією можна виділити постобробку після можливої фальсифікації через клонування або фотомонтаж. Постобробка може виконуватись різними інструментами: розмиття, підвищення різкості, мультиплікативний шум тощо. Методи виявлення мультиплікативного шуму мало освітлені у відкритих джерелах. Існує метод, заснований на аналізі коефіцієнтів ДКП матриці цифрового зображення, який добре зарекомендував себе при збереженні оброблених файлів без втрат. Однак його не було перевірено при збереженні у форматі з втратами. Мета роботи – підвищення ефективності виявлення мультиплікативного шуму як обробки цифрового зображення. Обчислювальний експеримент проведено з використанням бази із 100 зображень, кожне з яких було піддано обробці мультиплікативним шумом з дисперсією 0,005 і збережено в jpeg у якості від 0 до 100 з кроком 5. Було знайдено різницю максимального та мінімального значення високих частот (амплітуду) двома різними способами і визначено два показники. Встановлено, що точність виявлення мультиплікативного шуму залежить від ступеня стиснення з втратами. Точність виявлення шуму сильно знижується, коли коефіцієнт стиснення встановлено нижче 70. Показник П1 підходить для виявлення мультиплікативного шуму у зображеннях з якістю від 98 до 100 і від 70 до 89, П2 – для зображень з якістю від 89 до 97. Проведені дослідження дозволили значно підвищити ефективність виявлення мультиплікативного шуму як порушення цілісності цифрового зображення.
**Ключові слова:** мультиплікативний шум, дискретне косинусне перетворення, порушення цілісності цифрового зображення.

**Вступ.** Мультиплікативний шум – це постобробка, яка може бути використана для приховання фальсифікації зображення або порушення стеганографічного повідомлення. Не завжди мультиплікативний шум може бути помітним візуально (рис. 1), особливо якщо зображення було збережено з втратами, наприклад, у форматі JPEG. Тому важливо знайти спосіб, який допоможе визначити наявність мультиплікативного шуму у зображенні.

Мета роботи – підвищення ефективності виявлення мультиплікативного шуму як обробки цифрового зображення.

Для досягнення поставленої мети потрібно виконати наступні задачі.

1. Дослідити сучасні методи виявлення пост обробки цифрового зображення при його фальсифікації.
2. Обрати метод виявлення мультиплікативного шуму для дослідження його ефективності в умовах збереження зображення після пост обробки у форматі з втратами.

3. Виконати модифікацію алгоритму метода, взятого за основу, та реалізувати його у програмному додатку.
4. Оцінити ефективність модифікованого алгоритму.

**Матеріали та методи.** Існує декілька способів визначення наявності постобробки у зображенні, серед яких – метод, заснований на аналізі сингулярних чисел матриці зображення (виявлення розмиття, штучного підвищення різкості) [1], і метод, заснований на аналізі коефіцієнтів дискретного косинусного перетворення (виявлення розмиття, мультиплікативного шуму в умовах збереження після обробки без втрат) [2].

Дискретне косинусне перетворення (ДКП) – це ортогональне перетворення, яке використовується, зокрема, в обробці сигналів та стисненні з втратами і засноване на перетворенні Фур'є [3]. На ньому базується найпопулярніший формат стиску з втратами – JPEG. Перевага даного формату в тому, що він дозволяє зменшити займаний файлом на диску простір для зберігання і його підтримують майже усі графічні програми і веб-браузери, тому його часто використовують в соціальних мережах [4].

Стиснення зображення відбувається за такими кроками:

1. Матриця зображення ділиться на блоки 8×8.
2. Для кожного блока зображення застосовується ДКП за формулою

$$DCT(i,j) = \frac{1}{\sqrt{2N}} C(i)\,C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} pixel(x,y) \cos\left[\frac{(2x+1)i\pi}{2N}\right] \cos\left[\frac{(2y+1)j\pi}{2N}\right]$$

За допомогою формули ДКП здійснюється перехід з просторової області у частотну. Тобто кожен блок зображення представляється у вигляді матриці 8×8, де зверху зліва знаходяться низькі частоти, які відповідають за фонові елементи зображення, а знизу справа – високі частоти, які відповідають за контури (рис. 2).

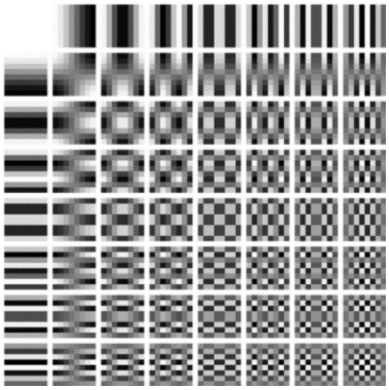


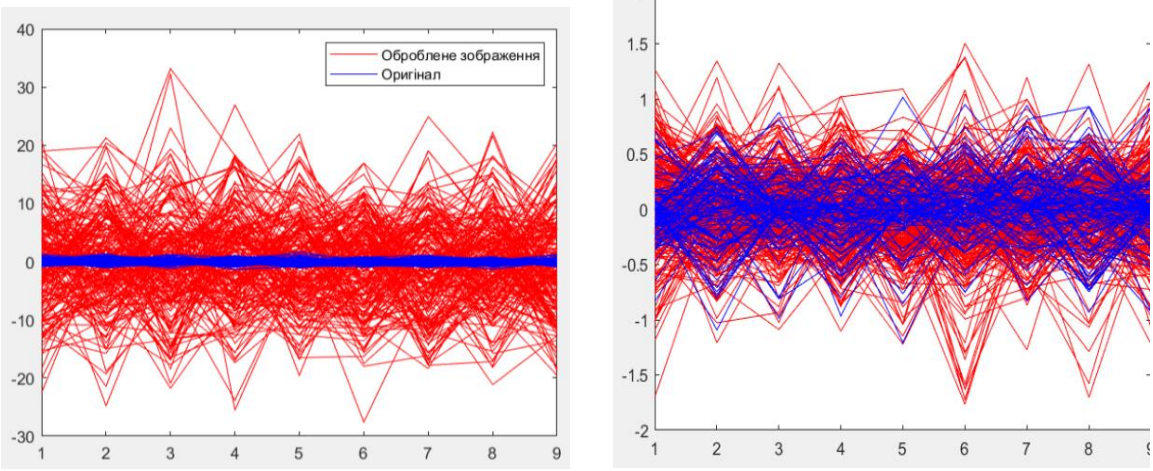
**Рис. 1**. Блок частотної області ДКП

3. Елементи блоків ділять на матрицю квантування та округлюють до найменшого цілого. У результаті високі частоти можуть обнулитися, тому на контурах можуть виникнути артефакти.
4. Отримана матриця записується у масив зігзагом зліва направо зверху вниз. Останні нулі не записуються, тому масив займає менше біт.

**Результати та обговорення.** Після додавання на зображення мультиплікативного шуму збільшується амплітуда коефіцієнтів високих частот ДКП (рис. 3). Метод виявлення мультиплікативного шуму на основі коефіцієнтів ДКП полягає у тому, що виявляється таке порогове значення, яке дозволяє відрізнити оброблене зображення від необробленого. На рисунку видно, що необроблені зображення

мають набагато меншу амплітуду, ніж оброблені. Але так як JPEG стиснення впливає на високі частоти, у разі збереження в гіршій якості амплітуда буде відрізнятися набагато менше (рис. 5), отже, відрізнити оброблене зображення від необробленого буде складніше.

Обчислювальний експеримент проведено з використанням бази із 100 зображень, кожне з яких було піддано обробці мультиплікативним шумом з дисперсією 0,005 і збережено в jpeg у якості від 0 до 100 з кроком 5. Було знайдено різницю максимального та мінімального значення високих частот (амплітуду) двома різними способами і визначено два показники.

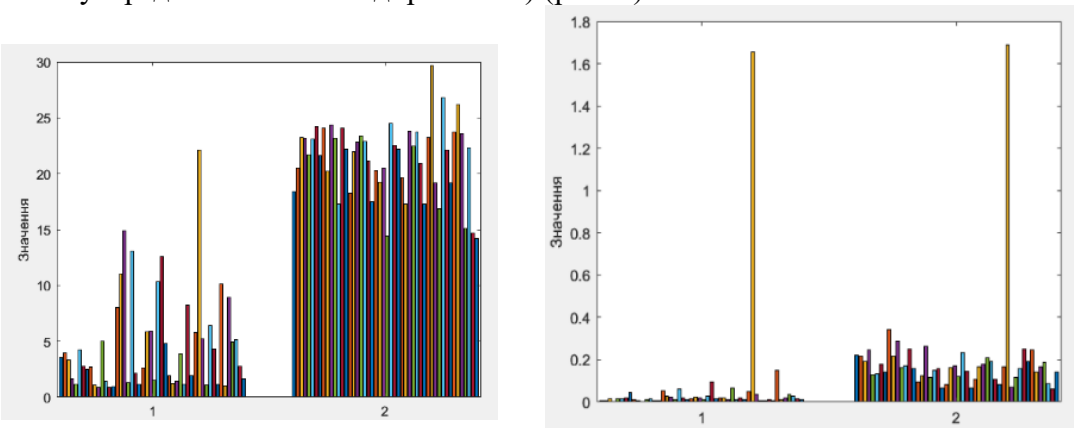



а. б.

**Рис. 2**. Порівняння коефіцієнтів високих частот ДКП обробленого і необробленого зображень, що збережені у форматі JPEG у якості 100 (а) та 70(б)

Показник 1 – різниця між найбільшим і найменшим значенням усереднених коефіцієнтів високих частот блоків матриці цифрового зображення (кількість усереднених значень дорівнює кількості блоків матриці) (рис. 6).

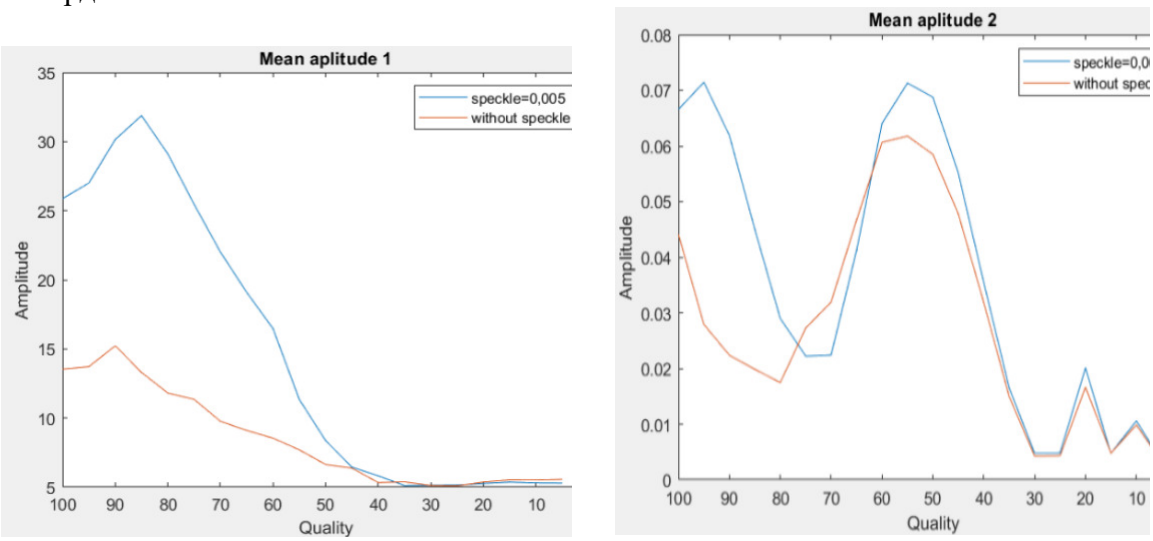
Показник 2 – різниця між найбільшим і найменшим усередненим по матриці значенням високочастотного коефіцієнта з відповідними індексами (кількість усереднених значень дорівнює 9) (рис. 6).




а. б.

**Рис. 3** – Порівняння зображень без мультиплікативного шуму (1) і з мультиплікативним шумом (2), де П1– а, П2– б.

Далі було проаналізовано П1 для зображень у різній якості. На графіку (рис. 4) відображено, як змінюються середні значення Показника 1 в залежності від якості збереження зображення. По осі абсцис відкладено значення якості, по осі ординат – П1.




а. б.

**Рис.4**. Середні значення П1(а) та П2 (б) для зображень у різній якості

Видно, що після деякого значення якості графіки наближуються один до одного. Те саме можемо спостерігати і для П2 (рис. 8).

Отже, видно, що після деякого значення якості стає неможливим відрізнити показники обробленого і необробленого зображень.

Експериментальним шляхом було визначено порогові коефіцієнти і їх точність (ACC) за формулою:

$$ACC=(TP+TN)/(TP+TN+FP+FN),$$

де TP – істинно позитивні результати;

TN – істинно негативні результати;

FP – помилки II роду, тобто хибно позитивні результати;

FN – помилки I роду, тобто хибно негативні результати.

Було отримано графік (рис. 9), на якому порівнюються значення точності П1 і П2. Видно, що точність П1 різко падає для зображень із якістю менше 70, а точність П2 падає для зображень із якістю менше 90. Тому подальші розрахунки були зроблені для зображень із якістю від 70 до 100.

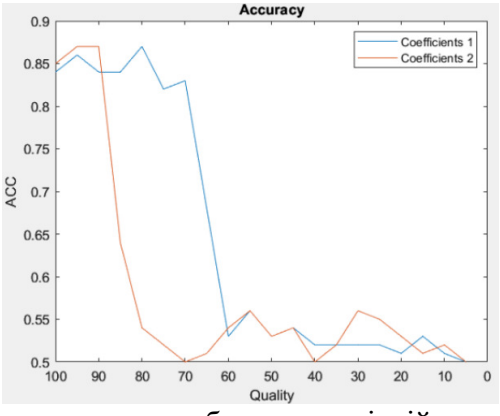



**Рис. 5**. ACC порогових значень для зображень у різній якості

Проаналізовано 100 зображень, для яких визначено пороговий коефіцієнт, який має найбільшу точність виявлення мультиплікативного шуму. Для значення мультиплікативного шуму 0,005 були отримані результати, що представлені в табл. 1.

**Таблиця 1**

Порогові значення показників при imnoise = 0.005

| Якість | 100 | 95 | 90 | 85 | 80 | 75 | 70 |
|---|---|---|---|---|---|---|---|
| П1 | 16.5524 | 17.698 | 19.4205 | 19.923 8 | 10.615 | 11.502 | 2.9000 |
| ACC1 | 0.7850 | 0.7950 | 0.7900 | 0.7950 | 0.7550 | 0.7700 | 0.6150 |
| Помилки 2 роду | 0.2150 | 0.2050 | 0.2100 | 0.2000 | 0.2450 | 0.2300 | 0.3850 |
| Помилки 1 роду | 0 | 0 | 0 | 0.0050 | 0 | 0 | 0 |
| П2 | 0.0180 | 0.0158 | 0.0133 | 0.0087 | 0.0092 | 0.0051 | 0.0000 |
| ACC2 | 0.7900 | 0.8100 | 0.8250 | 0.5950 | 0.5400 | 0.5100 | 0.5000 |
| Помилки 2 роду | 0.2150 | 0.1900 | 0.1750 | 0.4100 | 0.4600 | 0.4950 | 0.5000 |
| Помилки 1 роду | 0.0100 | 0.0050 | 0 | 0 | 0.0100 | 0 | 0 |

Можна побачити, що спочатку П2 з більшою точністю виявляє шум, але після значення якості 90 точність П1 збільшується, а П2 падає. Можна зробити висновок, що для різних значень якості краще підходить різний спосіб пошуку показників.

Коефіцієнт П1 підходить для виявлення мультиплікативного шуму у зображеннях з якістю від 98 до 100 і від 70 до 89, П2 – для зображень з якістю від 89 до 97. Отримані такі порогові коефіцієнти П1: 16.5524 для якості 100, 17.6982 для 95, 19.4205 для 90, 19.4205 для якості 85, 10.6152 для якості 80, 11.5025 для якості 75, 2.9 для якості 70. Порогові коефіцієнти П2 дорівнюють 0.0180 для якості 100, 0.0158 для 95, 0.0133 для 90, 0.0087 для 85, 0.0092 для 80, 0.0051 для 75.

Найбільша точність досягається при П2 для якості 90-95, при цьому помилки 2 роду становлять 19% і 17.5% відповідно, а помилки 1 роду – 0.5% і 0% відповідно, а також з П1 для якості 100 і 85 помилки 2 роду становлять 21.5% і 20%, а помилки 1 роду – 0 і 0.5% відповідно (рис. 6).

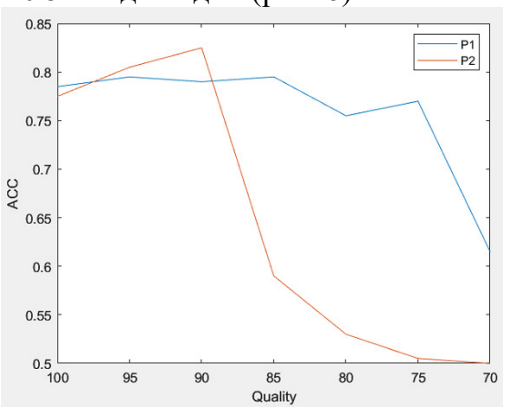



**Рис.6**. Порівняння точності порогових коефіцієнтів

**Висновки.** Досліджено вплив стиснення JPEG на виявлення мультиплікативного шуму в зображеннях методом, заснованим на аналізі високих частот ДКП: для зображень, що зберігаються з якістю від 70 до 100, були отримані порогові коефіцієнти, що дозволяють відділити оброблені зображення від необроблених. Порогові коефіцієнти були розраховані двома різними способами (П1 і П2) на основі бази із 100 зображень.

Встановлено, що точність виявлення мультиплікативного шуму залежить від ступеня стиснення з втратами. Точність виявлення шуму сильно знижується, коли коефіцієнт стиснення встановлено нижче 70.

Показник П1 підходить для виявлення мультиплікативного шуму у зображеннях з якістю від 98 до 100 і від 70 до 89, П2 – для зображень з якістю від 89 до 97. Отримані такі порогові коефіцієнти П1: 16.5524 для якості 100, 17.6982 для 95, 19.4205 для 90, 19.4205 для якості 85, 10.6152 для якості 80, 11.5025 для якості 75, 2.9 для якості 70. Порогові коефіцієнти П2 дорівнюють 0.0180 для якості 100, 0.0158 для 95, 0.0133 для 90, 0.0087 для 85, 0.0092 для 80, 0.0051 для 75.

Найбільша точність досягається з П2 для якості 90-95, при цьому помилки 2 роду становлять 19% і 17.5%, а помилки 1 роду – 0.5% і 0%, а також з П1 для якості 100 і 85, помилки 2 роду становлять 21.5% і 20%, а помилки 1 роду – 0 і 0.5% відповідно.

**Список літератури**

1. Розумяк Б.О., Зоріло В.В. Дослідження деяких розкладань і перетворень матриці цифрового зображення як основа метода виявлення мультиплікативного шуму. Одеса: НУОП, 2022.
2. Зоріло В.В., Корольова Є.О., Розумяк Б.О. Метод виявлення розмиття та мультиплікативного шуму як обробки цифрового зображення на основі аналізу коефіцієнтів ДКП. *Інформатика та математичні методи в моделюванні*. 2022. №3-4. С.23-31.
3. Discrete cosine transform. URL: https://en.wikipedia.org/wiki/Discrete_cosine_transform
4. Discrete cosine transform. URL: https://www.mathworks.com/help/signal/ref/dct.html
5. JPEG. URL: https://en.wikipedia.org/wiki/JPEG

# DETECTION OF MULTIPLICATIVE NOISE IN DIGITAL IMAGES UNDER LOSSY STORAGE CONDITIONS

K.O.Petruk, V.V.Zorilo, O.Yu.Lebedeva

National Odesa Polytechnic University
Shevchenko Ave., 1, Odesa, 65044, Ukraine
email: vikazarilo@gmail.com

Methods of detecting violations of the integrity of digital information (digital photo, video, and audio files) play an important role for information and cyber security. Among the violations of the integrity of digital images, post-processing after possible falsification through cloning or photomontage can be singled out as a separate category. Post-processing can be done with various tools: blurring, sharpening, multiplicative noise, etc. The methods of detecting multiplicative noise are poorly illuminated in open sources. There is a method based on the analysis of the DCP coefficients of the digital image matrix, which has proven itself well when saving processed files without loss. However, it has not been tested when saved in a lossy format. The purpose of the work is to increase the effectiveness of multiplicative noise detection as a digital image processing. The computational experiment was carried out using a database of 100 images, each of which was subjected to multiplicative noise processing with a variance of 0.005 and saved in jpeg as a quality from 0 to 100 with a step of 5. The difference between the maximum and minimum value of high frequencies (amplitude) was found in two different ways and two indicators are defined. It was established that the accuracy of multiplicative noise detection depends on the degree of lossy compression. The accuracy of noise detection is greatly reduced when the compression ratio is set below 70. The P1 indicator is suitable for detecting multiplicative noise in images with quality from 98 to 100 and from 70 to 89, P2 - for images with quality from 89 to 97. The conducted research allowed to significantly improve the effectiveness of detecting multiplicative noise as a violation of the integrity of a digital image.
**Keywords**: multiplicative noise, discrete cosine transformation, violation of digital image integrity.

DOI 10.15276/imms.v13.no1-2.137
УДК 681.514:621.029

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp. 137-143

# ВИЯВЛЕННЯ АУДІО-ПІДРОБОК ЗАСОБАМИ ШТУЧНОГО ІНТЕЛЕКТУ

М.А Стецовський., В.В.Зоріло, О.Ю.Лебедєва

Національний університет «Одеська політехніка»
пр.Шевченка, 1, Одеса, 65044
email: vikazorilo@gmail.com

Розвиток інформаційних технологій, зокрема, штучного інтелекту, призводить до широкого їх застосування у багатьох сферах нашого життя. Із стрімким розвитком штучного інтелекту зростає кількість випадків його застосування для генерації цифрових зображень, аудіо, відео тощо. Підробка цифрових аудіо файлів є небезпечною з точки зору використання зловмисниками для чинення інформаційно-психологічного впливу та маніпуляцій суспільством та окремими індивідами. Існують сучасні методи виявлення аудіо-підробок, виконаних засобами штучного інтелекту. Вони мають високу точність, при цьому не позбавлені недоліків. Основним недоліком є складна архітектура та висока ресурсоємність. Метою даною роботи є розробка нейронної мережі, яка дозволила б з задовільною точністю виявляти аудіо-підробку, та навчання якої не вимагало б значних обчислювальних ресурсів. Було обрано метод для модифікації, а саме метод виявлення аудіо підробок з використанням згорткової нейронної мережі. Було модифіковано метод виявлення аудіо підробок шляхом побудови моделі з новою архітектурою з меншою кількістю шарів, що дозволило значно скоротити часові витрати та потреби в значних обчислювальних ресурсах в порівнянні з аналогами. Отримані результати експериментів із застосуванням модифікованого методу показують задовільну ефективність і точність системи. Помилки 1 роду склали 24%, помилки другого роду – 9%. Розроблену модифікацію реалізовано у програмному додатку із зручним і простим інтерфейсом.
**Ключові слова:** штучний інтелект, нейронні мережі, аудіо-підробка, виявлення аудіо-фейку.

**Вступ.** З розвитком технологій поширення підробленого контенту викликає значне занепокоєння в різних сферах. У той час як візуальні підробки привертають значну увагу, аудіо підробки стали ще однією сферою, що викликає занепокоєння.

Аудіо підробки можуть завдати значної шкоди, включаючи поширення дезінформації, видавання себе за іншу особу та маніпуляції з аудіо доказами. Зі зростанням легкості створення переконливих аудіо підробок потреба в ефективних механізмах їх виявлення стає першочерговою.

Порівняно з візуальними підробками, кількість загальнодоступних наборів даних про аудіо підробки є відносно обмеженою. Такий дефіцит даних створює проблеми для навчання та оцінки моделей глибокого навчання, спеціально розроблених для виявлення аудіо підробок. Вирішення цієї проблеми вимагає спільних зусиль у створенні та поширенні різноманітних наборів даних про аудіо підробки.

Моделі глибокого навчання можуть вимагати значної обчислювальної потужності, що обмежує їхнє розгортання на пристроях з обмеженими ресурсами або платформах потокового мовлення в реальному часі. Оптимізація моделей для підвищення ефективності без втрати точності є ключовим напрямком досліджень.

Інтерпретованість моделей виявлення аудіо підробок є критично важливою проблемою. Розуміння та пояснення процесу прийняття рішень, що лежить в основі цих моделей, є важливим, особливо в юридичному або судовому контексті, де прозорість і

довіра мають першорядне значення. Розробка зрозумілих методів виявлення глибоких підробок є сферою активних досліджень.

Ефективними сучасними методами виявлення аудіо підробок є наступні.

DeepSpectrum – це модель глибокого навчання, спеціально розроблена для аналізу та класифікації спектроскопічних даних.

DeepSpectrum використовує можливості алгоритмів глибокого навчання для автоматичного вилучення значущих характеристик зі спектроскопічних даних і створення точних прогнозів або класифікацій. Навчаючись на великих масивах даних мічених спектрів, DeepSpectrum може вивчати складні закономірності та взаємозв'язки в даних, що дозволяє йому виконувати такі завдання, як ідентифікація матеріалів, контроль якості або виявлення аномалій [5].

Однією з ключових переваг DeepSpectrum є його здатність обробляти високорозмірні спектроскопічні дані, які часто містять численні спектральні смуги або канали. Модель може ефективно фіксувати і представляти складні спектральні особливості, що дозволяє проводити надійний аналіз і класифікацію.

Однак, як і будь-яка модель машинного навчання, DeepSpectrum також має певні обмеження. Для досягнення оптимальної продуктивності їй потрібна значна кількість маркованих навчальних даних, що може бути проблемою для областей з обмеженими або дефіцитними даними. Навчання моделі DeepSpectrum на наборі даних з 3000 вибірок потенційно може зайняти годину на епоху або навіть більше.

VGGish – це модель глибокого навчання, розроблена дослідницькою групою Google зі штучного інтелекту, спеціально призначена для завдань аналізу аудіо. Вона в першу чергу використовується для вилучення аудіо-вкладень або ознак зі спектрограм, які потім можуть бути використані для різних завдань, пов'язаних з аудіо, таких як класифікація аудіо, зіставлення схожості аудіо або виявлення аудіо-подій [6].

Однією з головних переваг VGGish є його здатність навчатися багатим і дискримінативним репрезентаціям з аудіо даних. Використовуючи свою глибоку архітектуру та велику кількість параметрів, що навчаються, VGGish може захоплювати як низькорівневі, так і високорівневі характеристики звуку, що робить його придатним для широкого спектру завдань аудіоаналізу.

Крім того, VGGish навчений на великому наборі аудіоданих, що дозволяє йому добре узагальнювати різні аудіодомени. Попередньо навчену модель VGGish, яка є загальнодоступною, можна використовувати як екстрактор ознак, де вихідні дані проміжних шарів моделі можуть бути використані як вбудовування аудіо для подальших завдань. Це дозволяє користувачам скористатися перевагами вивчених репрезентацій без необхідності тривалого навчання на власних наборах аудіоданих.

Ще однією перевагою VGGish є простота використання та інтеграції. Модель доступна як реалізація TensorFlow, що робить її сумісною з різними фреймворками глибокого навчання і дозволяє легко інтегрувати в існуючі робочі процеси.

Однак існує ризик надмірної адаптації, особливо при роботі з обмеженою кількістю мічених даних. Якщо навчальний набір даних відносно невеликий, модель може погано узагальнювати невидимі аудіо-зразки, що призведе до зниження продуктивності.

Навчання та використання моделі може вимагати значних обчислювальних ресурсів, включаючи пам'ять та обчислювальну потужність. Навчання моделі на низькопродуктивних або обмежених в ресурсах пристроях може бути складним, а висновок в реальному часі на таких пристроях також може бути складним в обчислювальному плані. Навчання моделі VGGish на наборі даних з 3000 вибірок потенційно може зайняти кілька годин на епоху або навіть більше.

Зазначені моделі дозволяють з високою точністю виявляти аудіо-підробки, однак це вимагає значних обчислювальних ресурсів, що ускладнює їх використання в побуті або в умовах неможливості доступу до інтернету або хмарних сервісів.

Метою даною роботи є розробка нейронної мережі, яка дозволила б з задовільною точністю виявляти аудіо-підробку, та навчання якої не вимагало б значних обчислювальних ресурсів.

**Матеріали та методи.** Як альтернативу зазначеним методам, основним недоліком яких є потреба у великих обчислювальних потужностях, в даній роботі запропоновано наступну модель нейронної мережі.

Модель, побудована за допомогою Sequential API, має шарувату структуру, яка поєднує шари Conv2D, MaxPooling2D, Flatten та Dense. Розглянемо кожен компонент більш детально:

Шари Conv2D: Використовуються два шари Conv2D з 16 фільтрами кожен. Ці шари виконують операції згортки над вхідною аудіо спектрограмою, яка представляє звук у візуальному форматі. Застосовуючи фільтри 3×3, модель може виокремлювати локальні особливості зі спектрограми, фіксуючи основні патерни, пов'язані з аудіо сигналом. Набір фільтрів, також відомих як ядра, у згортковому шарі згорткової нейронної мережі – це набір матриць, які використовуються для згортки з вхідними даними [1].

Кожне ядро виконує операцію згортки вхідних даних. Під час операції згортки ядро ковзає по вхідним даним (наприклад, по зображенню) і обчислює скалярний добуток між елементами ядра і відповідними вхідними елементами. Результатом цього обчислення є нова матриця, яка називається картою ознак або картою згортки.

Набір фільтрів у шарі згортки дозволяє виявляти різні локальні особливості вхідних даних. Кожен фільтр може спеціалізуватися на виявленні певних особливостей, таких як контури, текстури або форми. Кілька фільтрів можна використовувати для виявлення різних особливостей у різних частинах вхідних даних. Наприклад, деякі фільтри можуть виявляти вертикальні контури, інші – горизонтальні, а треті – особливості текстури.

Таким чином, набір фільтрів у згортковому шарі допомагає моделі виявляти і розпізнавати різні особливості у вхідних даних, формуючи карту особливостей, яка передається на наступний рівень мережі для подальшого аналізу і виконання завдань, таких як класифікація або виявлення об'єктів.

Функція активації: Функція активації Rectified Linear Unit (ReLU) використовується після кожного шару Conv2D. Функція ReLU (Rectified Linear Unit) є активаційною функцією, часто використовуваною в нейронних мережах. Вона приймає вхідне значення і повертає максимум між нулем і вхідним значенням. ReLU вносить нелінійність у модель, дозволяючи їй вивчати складні взаємозв'язки між виділеними ознаками. Ця нелінійна активація полегшує здатність моделі вловлювати складні деталі та дискримінаційні характеристики, присутні в аудіо даних. Математичне представлення ReLu функції наступне $f(x) = max(0, x)$.

Шари MaxPooling2D: Два шари MaxPooling2D слідують за шарами Conv2D. Ці шари виконують даунсемплінг, вибираючи максимальні значення у вікні об'єднання. Зменшуючи просторові розміри вхідних даних, MaxPooling2D допомагає зберегти найбільш важливу інформацію, зменшуючи при цьому обчислювальну складність [2].

Шар максимального пулінгу застосовується зазвичай після згорткових шарів у згорткових нейронних мережах. Його основна мета – зменшити просторові розміри карт ознак, ущільнюючи інформацію і витягуючи найважливіші ознаки.

Це допомагає знизити кількість параметрів у моделі, скоротити обчислювальну складність і зробити модель стійкішою до невеликих перекладацьких спотворень вхідних даних.

Зазвичай шари максимального пулінгу застосовуються послідовно із згортковими шарами для зменшення просторових розмірів карт ознак. Це дає змогу моделі зосередитися на найбільш значущих ознаках і покращує інваріантність до масштабування та переміщень об'єктів на зображенні.

Шар Flatten: Шар Flatten призначений для перетворення багатовимірних даних, отриманих на попередніх шарах, в одновимірний вектор. Це перетворення готує дані для наступних повністю з'єднаних шарів, дозволяючи моделі вивчати глобальні особливості звуку [3].

Після застосування згорткових шарів у згорткових нейронних мережах, вихідні дані можуть мати форму тривимірного тензора, наприклад, (batch_size, height, width, channels), де batch_size – розмір пакета (batch), height і width – розміри висоти і ширини, а channels – кількість каналів.

Операція Flatten() перетворює ці тривимірні дані в одновимірний вектор, об'єднуючи всі елементи в послідовність. Таким чином, кожен елемент тривимірного тензора стає окремим елементом одновимірного вектора.

Операція Flatten() використовується для переходу від згорткових шарів до повнозв'язаних шарів нейронної мережі. Після цієї операції дані можуть бути подані на шари з повнозв'язними нейронами, які очікують одновимірні входи. Це дає змогу нейронній мережі моделювати складні залежності між ознаками і зробити прогнози або класифікацію на основі цих даних.

Шар Dense:: Повнозв'язні шари (Dense Layers) у нейронних мережах відіграють роль об'єднання ознак, отриманих із попередніх шарів, і моделювання складніших залежностей у даних [4].

Вони встановлюють зв'язки між кожним нейроном у поточному шарі та кожним нейроном у попередньому шарі. Це означає, що кожен вихідний сигнал із попереднього шару впливає на кожен нейрон у повнозв'язному шарі.

Кожен нейрон у повнозв'язному шарі отримує зважену суму вхідних сигналів, де кожен вхідний сигнал множиться на свою відповідну вагу. Ваги визначаються під час навчання моделі і являють собою параметри, які модель намагається оптимізувати.

Після обчислення зваженої суми, нейрони повнозв'язного шару застосовують функцію активації до результату. Функція активації додає нелінійність у модель, даючи змогу моделі апроксимувати складні нелінійні залежності в даних.

Вихідний сигнал кожного нейрона в повнозв'язному шарі являє собою результат застосування функції активації до зваженої суми вхідних сигналів. Ці виходи можуть передаватися наступному шару в нейронній мережі або використовуватися для зробити прогнози, наприклад, у завданні класифікації.

Повнозв'язні шари дають змогу моделювати складніші залежності між ознаками та створювати більш гнучкі та виразні моделі. Їх часто використовують наприкінці нейронної мережі для ухвалення рішень або видачі остаточних виходів моделі.

До моделі включено два повнозв'язні шари. Перший щільний шар складається з 128 одиниць і використовує функцію активації ReLU. Цей шар виконує нелінійні перетворення вхідних даних, виділяючи ознаки вищого рівня. Останній щільний шар складається з одного елемента з сигмоїдною функцією активації, що дає ймовірнісний вихід для задач бінарної класифікації.

Вищезгадана архітектура моделі, представлена на рисунку 1, пропонує кілька помітних переваг для задач класифікації аудіо:

1) виділення локальних особливостей – шари Conv2D дуже ефективно виділяють локальні особливості в аудіо спектрограмі. Згортаючи фільтри над невеликими сприйнятливими полями, модель може вивчати значущі представлення локальних патернів, включаючи часові та спектральні характеристики. Ця здатність вловлювати дрібні деталі підвищує дискримінаційну здатність моделі;

2) ієрархічне представлення – шари Conv2D і MaxPooling2D, що накладаються один на одного, полегшують створення ієрархічного представлення звукових характеристик. Початкові шари захоплюють низькорівневі елементи, такі як краї, текстури та основні компоненти звуку. Коли інформація проходить через наступні шари, вивчаються більш складні та абстрактні представлення. Ця ієрархічна структура дозволяє моделі виокремлювати як низькорівневі, так і високорівневі ознаки з вхідної спектрограми;

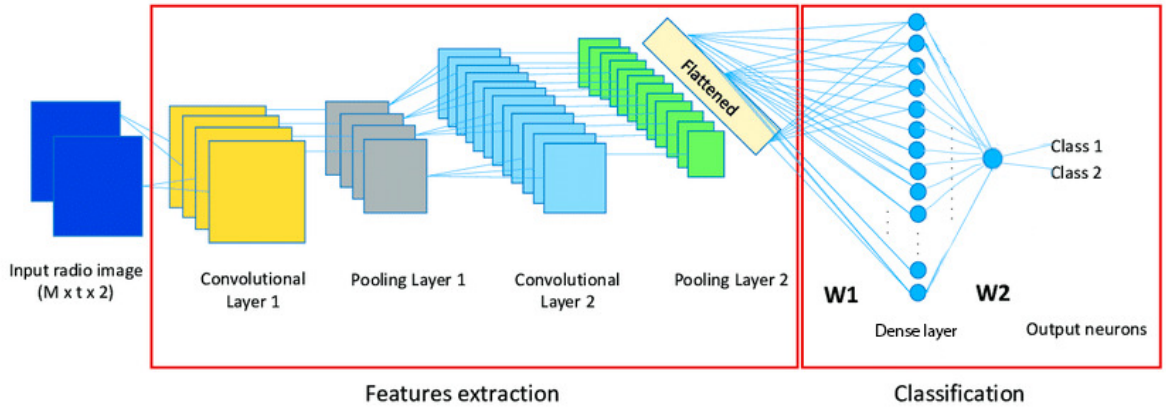
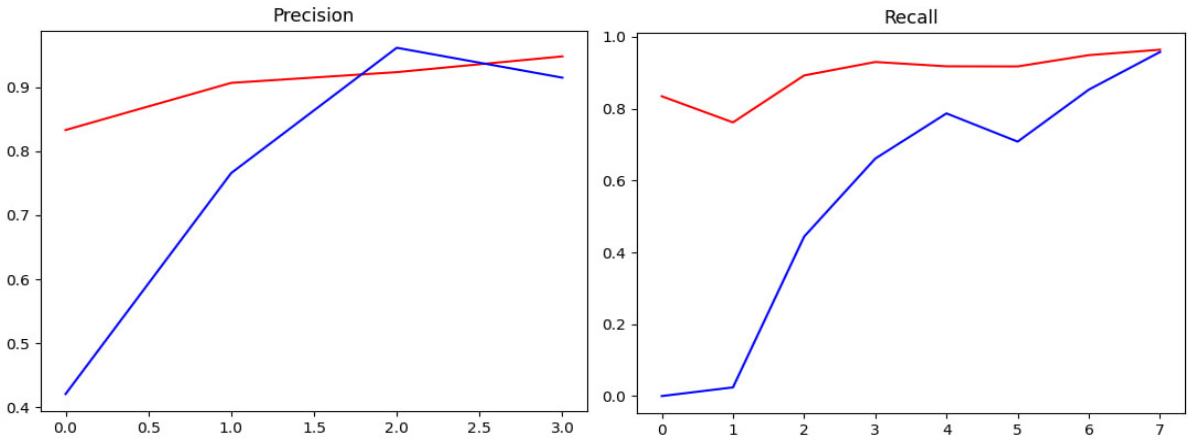



**Рис. 1.** Архітектура моделі

3) нелінійність і функції активації – включення функцій активації ReLU вносить в модель нелінійність. Ця нелінійність є життєво важливою для відображення складних взаємозв'язків, присутніх в аудіоданих, які часто демонструють нелінійні патерни і залежності. Фінальна сигмоїдна функція активації дозволяє моделі виробляти ймовірнісний вихід, що робить її придатною для задач бінарної класифікації;

4) ефективність параметрів: архітектура моделі забезпечує баланс між збором релевантної інформації та уникненням надмірного налаштування. Використовуючи помірну кількість фільтрів та операцій об'єднання, модель зменшує кількість параметрів порівняно з більш глибокими архітектурами. Така ефективність параметрів робить модель обчислювально ефективною, дозволяючи їй обробляти більші набори даних і зменшуючи ризик перенавчання, особливо при роботі з обмеженою навчальною вибіркою.

**Результати.** Тренування моделі на датасеті в 3000 екземплярів зайняло сумарний час в півгодини, було проведено 5 епох тренування. Виконано модифікацію методу виявлення аудіо підробок з використанням нейронної мережі. За попереднім результатом тренування впевненість склала близько 90%. Порівняно з аналогами це гірше відносно точності моделі, але краще в використанні обмеженої кількості ресурсів. На рис.2 червоним позначено результат, отриманий на навчальному датасеті, синім – на даних тестування.




а. б.

**Рис. 2**. Графіки впевненості (а) і повноти (б) моделі

Precision (впевненість) – це показник ефективності моделі в контексті бінарної класифікації, який вимірює частку вірно класифікованих позитивних результатів серед усіх екземплярів, які модель передбачила як позитивні. Він дає змогу оцінити, наскільки точно модель ідентифікує істинно позитивні екземпляри.

Recall (повнота) – це показник ефективності моделі, який вимірює здатність моделі правильно ідентифікувати позитивні екземпляри серед загальної кількості дійсних позитивних екземплярів у наборі даних. Він дає уявлення про те, наскільки модель здатна виявляти всі позитивні приклади або мінімізувати помилкові негативні результати.

Після проведення тестування зі 100 реальними та 100 підробленими екземплярами була визначена кількість помилок 1 та 2 роду (рис. 3).

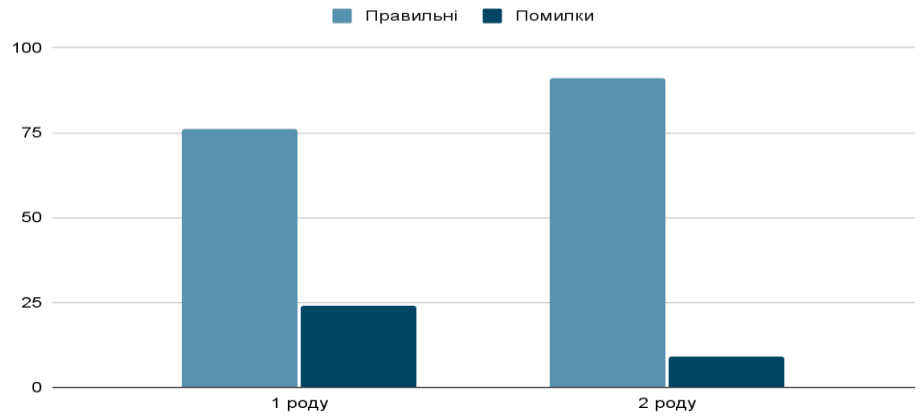



**Рис.3.** Помилки 1 та 2 роду

Помилки 1 роду склали 24%, 2 роду – 9% Загалом програма продемонструвала задовільну роботу та високу ефективність.

Помилки першого роду, або хибні позитивні спрацьовування, можуть іноді виникати в процесі аналізу даних або прийняття рішень. Вони можуть бути спричинені різними факторами, такими як статистична варіація або шум у даних. Однак, незважаючи на це, загальна кількість таких помилок залишається малою і не перевищує прийнятний рівень. Важливо зазначити, що програма все одно демонструє високу точність і надійність у виконанні своїх основних функцій.

Крім того, важливо усвідомлювати, що під час оцінювання ефективності програми необхідно враховувати й інші параметри, такі як показники точності, повноти та загальну здатність моделі передбачати правильні результати.

**Список літератури**

1. What is a convolutional layer?. Databricks. URL: https://www.databricks.com/glossary/convolutional-layer.
2. Papers with code – max pooling explained. The latest in Machine Learning | Papers With Code. URL: https://paperswithcode.com/method/max-pooling.
3. Layer – flatten. TensorSpace.js. URL: https://tensorspace.org/html/docs/layerFlatten.html.
4. Kaplan D. Dense layer: the building block to neural networks. URL: https://enjoymachinelearning.com/blog/dense-layer/.
5. GitHub – DeepSpectrum/DeepSpectrum. GitHub. URL: https://github.com/DeepSpectrum/DeepSpectrum.
6. VGGish. GitHub. URL: https://github.com/tensorflow/models/tree/master/research/audioset/vggish.

## DETECTION OF AUDIO FAKES BY MEANS OF ARTIFICIAL INTELLIGENCE

M.A. Stetsovskyi, V.V.Zorilo, O.Yu. Lebedeva

National Odesa Polytechnic University,
1, Shevchenko Ave, Odesa, Ukraine
vikazorilo@gmail.com

The development of information technologies, including artificial intelligence, is leading to their widespread use in many areas of our lives. With the rapid development of artificial intelligence, the number of cases of its application to generate digital images, audio, video, etc. is growing. Counterfeiting digital audio files is dangerous from the point of view of being used by criminals to exert information and psychological influence and manipulate society and individuals. There are modern methods for detecting audio fakes made by artificial intelligence. They are highly accurate, but not without drawbacks. The main drawback is the complex architecture and high resource intensity. The aim of this paper is to develop a neural network that would allow detecting audio fakes with satisfactory accuracy and whose training would not require significant computing resources. A method was chosen for modification, namely, the method of detecting audio fakes using a convolutional neural network. The method for detecting audio fakes was modified by building a model with a new architecture with fewer layers, which significantly reduced the time and computational resources required compared to analogues. The experimental results obtained using the modified method show satisfactory efficiency and accuracy of the system. Errors of the first kind amounted to 24%, errors of the second kind - 9%. The developed modification is implemented in a software application with a convenient and simple interface.

**Keywords:** artificial intelligence, neural networks, audio forgery, audio fake detection

DOI 10.15276/imms.v13.no1-2.144
УДК 004.56.5(043.2)

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp.144-161

# ОЦІНКА ВЛАСТИВОСТЕЙ ІНФОРМАЦІЙНИХ ВТОРГНЕНЬ

В.О.Хорошко, В.Д.Козюра, Ю.Є.Хохлачова, Н.С.Вишневська

Національний авіаційний університет,
1, просп. Любомира Гузара, Київ, 03058, Україна,
e-mail: professor_va@ukr.net

У статті запропонована оцінка властивостей інформаційних вторгнень, оскільки вони являють собою випадкові процеси і мають свої властивості. Це дозволяє з'ясувати рівень впливу їх на особистість чи групу людей, і розробити ефективну систему протидії. При цьому психологічний вплив або інформаційне вторгнення, що є складовою інформаційно-психологічного протиборства, - це вплив на людей та їх групи, що здійснюється з метою зміни ідеологічних та психологічних структур їх свідомості та підсвідомості, трансформації емоційних станів, стимулювання певних типів поведінки. Перебудова психіки під впливом інформаційно-психологічного впливу може бути різною, як у широті, так і по тимчасовій стійкості. Для того, щоб надати інформаційне вторгнення та психологічний вплив, необхідно спочатку спровокувати збої та перенесення впливів. Динамічна рівновага між ними порушується, людина починає переживати стан когнітивного дисонансу. Після цього можна спонукати до відновлення душевної рівноваги за рахунок зміни своїх колишніх звичних йому поглядів, переконань і відносин, табу і стереотипів поведінки. Тому дуже важливо оцінювати властивості інформаційних вторгнень чи психологічних впливів, що дозволяють з'ясувати рівень впливу їх на особистість чи групу людей, і розробити ефективну систему протидії. Показано, що для їх аналізу можна використовувати всі ті методи, які використовуються для оцінки та аналізу випадкових процесів. Також було проведено розгляд завдання виявлення стрибкоподібних сплесків, який показав, що за допомогою таких сигналів спостереження можуть бути диференційовані.
**Ключові слова:** кібербезпека, кіберпростір, інформаційне протиборство, інформаційно-психологічне протиборство, інформаційні вторгнення, математичні методи.

**Вступ.** Інформаційне протиборство – це поєднання теоретичних знань, практичних спеціальних методів та технологій. При цьому потенціал впливу інформаційно-психологічного протиборства величезний. Це обумовлено самою його суттю.

З часом неконтрольоване поширення та використання інформаційного та кіберпростору, разом з отриманням значних переваг від їх використання призвело і до виникнення нових принципів, пов'язаних з цими проблемами. Головною з них стало різке загострення міжнародної конкуренції за володіння інформаційними ресурсами та інформаційним ринком. При цьому задля забезпечення інформаційного протистояння та проведення окремих операцій під час локальних війн та збройних конфліктів сучасності.

У цьому досить активно почали використовувати можливості інформаційної магістралі, як Internet. Яскравим прикладом стали події в Ірані, Югославії, Лівії, Чечні, Грузії, Україні і тощо [1].

При цьому психологічний вплив або інформаційне вторгнення, що є складовою інформаційно-психологічного протиборства, це вплив на людей та їх групи, що здійснюється з метою зміни ідеологічних та психологічних структур їх свідомості та підсвідомості, трансформації емоційних станів, стимулювання певних типів поведінки [1].

Перебудова психіки під впливом інформаційно-психологічного впливу може бути різною, як у широті, так і по тимчасовій стійкості. За першим критерієм розрізняють парціальні зміни, тобто зміни якоїсь однієї психологічної якості, та загальні зміни психіки, тобто зміни низки психологічних аспектів індивіда. По другому зміни можуть бути короткочасними та тривалими.

Психологічний вплив (інформаційне вторгнення) має закономірності [3]:

- якщо вплив спрямований насамперед на споживчо-мотиваційну сферу людей, тоді його результати позначаються, насамперед, на спрямованості і силі спонукань людей;

- поєднання впливів на обидві названі сфери дозволяє впливати на вольову активність людей і таким чином керувати їхньою поведінкою;

- вплив на комунікативно-поведінкову сферу (специфіку взаємодій та спілкування) дозволяє створити соціально-психологічний комфорт/дискомфорт, змушує людей співпрацювати або конфліктувати з оточуючими;

- в результаті інформаційного вторгнення (психологічного впливу) на інтелектуально-пізнавальну сферу людини змінюються в потрібний бік її подання, характер сприйняття інформації, що надходить, і, в результаті, її «картина світу».

Відповідно до сказаного, для того, щоб надати інформаційне вторгнення та психологічний вплив, необхідно спочатку спровокувати збої та перенесення впливів. Динамічна рівновага між ними порушується, людина починає переживати стан когнітивного дисонансу. Після цього можна спонукати до відновлення душевної рівноваги за рахунок зміни своїх колишніх звичних йому поглядів, переконань і відносин, табу і стереотипів поведінки.

Тому дуже важливо оцінювати властивості інформаційних вторгнень чи психологічних впливів, що дозволяють з'ясувати рівень впливу їх на особистість чи групу людей, і навіть розробити ефективну систему протидії.

**Метою роботи** є оцінка властивостей інформаційних вторгнень. Оскільки інформаційні вторгнення являють собою випадкові процеси і мають їхні властивості, то для аналізу можна використовувати всі ті методи, які використовуються для оцінки та аналізу випадкових процесів.

**Основна частина.** Завдання виявлення зміни властивостей інформаційних вторгнень (ІВ) дуже важливе для протидії їх впливу на людину, групу людей і суспільство в цілому. Також ІВ являють собою випадкові процеси і мають їхні властивості, тому для їх аналізу можна використовувати всі ті методи, які використовуються для оцінки та аналізу випадкових процесів [4-7].

У роботах [14, 15] розглядається завдання виявлення стрибкоподібних сплесків, припущення, що вони рано чи пізно відбудуться. Проте, з допомогою таких сигналів спостереження можуть бути диференційовані. Ці обставини необхідно враховувати під час побудови алгоритмів виявлення. Якщо задано максимально допустимий час спостереження, то для виявлення ІВ можна використовувати непослідовні і послідовні алгоритми.

У першому випадку рішення про наявність чи відсутність ІВ приймається на основі всього доступного набору спостережень. Для прискорення виявлення ІВ можна використовувати послідовні методи, що ґрунтуються на випадковому числі спостережень [4, 5, 6]. Основна відмінність алгоритмів виявлення ІВ від класичних алгоритмів перевірки гіпотез полягає в тому, що рішення про відсутність вторгнення може бути прийнято лише на останньому доступному кроці спостереження, оскільки ніколи немає гарантії, що воно відбудеться після закінчення спостережень. Розглянуті у статті завдання є окремим випадком більш загального виявлення [8].

Припустимо, що з кроками з номерами $n = \overline{1, N}$, де $N$ – максимально можливе число кроків спостереження, доступні спостереженню, взагалі кажучи, векторні

випадкові величини $x_n \in X_n$. Передбачається, що можуть виникнути дві ситуації $\theta = 0$ та $\theta = 1$, причому подія $\{\theta = 1\}$ означає зміну імовірнісних властивостей послідовності $\{x_n, n \geq 1\}$, що відбувається у випадковий момент $\lambda_o \in [0, \infty)$ Подія $\{\lambda_o = \infty\}$ еквівалентна відсутності ІВ і має ненульову ймовірність, тобто $P(\lambda_0 = \infty) = P(\theta = 0) = \Pi_{00} \in (0,1)$. Вважатимемо заданими апріорні ймовірності наявності $\Pi_{01} = P(\theta = 1) = P(\lambda_0 \in [0, \infty])$ і відсутність $\Pi_{00}$ вторгнення $(\Pi_{00} = 1 - \Pi_{01})$ та умовний апріорний розподіл моменту появи $\Pi(\lambda) = P(\lambda_0) \leq \lambda[\theta - 1) = P_1(\lambda_0 \leq \lambda)$, $\lambda \geq 0$.

Нехай умовні щільності векторів $x^n, n \geq 1$, має вигляд:

$$p_0(x_1^n) = p(x_1^n \mid \theta = 0) = \prod_{i=1}^{n} p_{0i}(x_i) = p(x_1^n \mid \theta = 1, \lambda_0 > n\Delta); \quad (1)$$

$$p_1(x_1^n \mid \lambda) = p(x_1^n \mid \theta = 1, \lambda_0 > \lambda) = \prod_{i=1}^{j} p_{0i}(x_i) p_{\lambda\, j+1}(x_{j+1}) \prod_{i=j+2}^{n} P_{1i}(x_i), \quad (2)$$

$j\Delta \leq \lambda \leq (j+1)\Delta$, $j \leq n-1$, $n = \overline{1,n}$,

де $p_{\lambda n}(x_n)$ – щільність, що залежить від $\lambda$, причому

$$P_{\lambda n}(x_n) = \begin{cases} P_{0n}(x_n) \text{ при } \lambda = n\Delta, \\ P_{0n}(x_n) \text{ при } \lambda = (n-1)\Delta \end{cases} \quad (3)$$

Таким чином до моменту вторгнення спостереження $x_n$, $n = 1,2,3,...$ незалежні із щільностями $p_{0n}$, а після вторгнення у встановленому режимі – незалежні із щільностями $p_{1n}$. Залежність щільності $p_{\lambda n}$ обумовлена поступовістю ІВ, причому передбачається , що режим, що встановився, настає не пізніше ніж через час $\Delta$, що дорівнює інтервалу часу між відліками (тобто, за один крок).

Введемо функцію втрат наступним чином

$$g(\theta, \lambda, u_n, n) = \begin{cases} g_{01}(n), \theta = 0, u_n = 1, \\ g_{11}(n) + c \cdot (n - [\lambda]), \theta = 1, \lambda < n\Delta, u_n = 1, \\ \tilde{g}_{11}(n), \theta = 1, \lambda < n\Delta, u_n = 1, n = \overline{1,n}, \end{cases} \quad (4)$$

де $u_n = 1$ – рішення про наявність ІВ на $n$-м кроці; $c$ – вартість затримки у винесенні рішення про наявність вторгнення на один крок; $[\lambda] = n-1$ при $(n-1)\Delta \leq \lambda \leq n\Delta$.

Рішення $u_n = 0$ про відсутність ІВ на кроках $n = \overline{1, N-1}$ ототожнюється з рішенням $u_n$ щодо подовження спостережень, оскільки при подальшому спостереженні вторгнення може виникнути та бути виявленим. На $N$-му кроці спостереження припиняється з вірогідністю 1 і разом із втратами (4) виникають втрати зв'язані з прийняттям рішення $u_N = 0$:

$$g(\theta, \lambda, u_N, N) = \begin{cases} g_{00}(N), \theta = 0, u_N = 0, \\ g_{10}(N) + c \cdot (N - [\lambda]), \theta = 0, \lambda < N\Delta, u_N = 0, \\ \tilde{g}_{10}(N), \theta = 1, \lambda < N\Delta, u_n = 0. \end{cases} \quad (5)$$

В (4) та (5) значення $g_{i1}(n)$, $g_{i0}(N)$, $\tilde{g}_{1j}(n)$ не залежать від $\lambda$, причому $g_{11}(n) \le g_{01}(n)$, $g_{00}(N) \le \tilde{g}_{10}(N) < g_{10}(N)$. Їх залежність від номера кроку може бути обумовлена вартістю спостережень.

Далі необхідно дослідження властивостей послідовності $\{\Pi_n, n \ge 1\}$, де $\Pi_n = P(\theta = 1 \mid x_1^n)$ апостеріорна вірогідність події $\{\theta = 1\}$, та зв'язаною з $\Pi_n$ рівністю

$$\Pi_n = \vartheta\Lambda_n / (1 + \vartheta\Lambda_n), n \ge 0 (\vartheta = \Pi_{01} / \Pi_{00}) \tag{6}$$

Статистики $\Lambda_n = \overline{p}_1(x_1^n \int p_0(x_1^n))$ усередненого щодо розподілу $\lambda_0$ об'єкта прогнозу (УОП).

Тут $\overline{p}_1(x_1^n) = \int_0^\infty p_1(x_1^n \mid \lambda) d\Pi(\lambda)$.

Оскільки (1)-(3), отримуємо

$$\Lambda_n = A_n + \sum_{j=0}^{n-1} \beta_{j+1}(x_{j+1}) \prod_{i=j+2}^{n} \gamma_i(x_i), \tag{7}$$

де $\prod_k^n \gamma_j = 1,\ k > n_i,\ A_n = P(\lambda_0 \ge n\Delta \mid \theta = 1)$;

$$\beta_n(x_n) = \int_{(n-1)\Delta}^{n\Delta} \frac{p_{\lambda n}(x_n)}{p_{0n}(x_n)} d\Pi(\lambda); \tag{8}$$

$$\gamma_n(x_n) = p_{1n}(x_n) / p_{0n}(x_n). \tag{9}$$

З (7) отримуємо рекурентну формулу

$$\Lambda_{n+1} = A_{n+1} + \beta_{n+1}(x_{n+1}) + \gamma_{n+1}(x_{n+1})(\Lambda_n - A_n),\ n \ge 0,\ \Lambda_0 = 1, \tag{10}$$

з якої витікає, що

$$\Lambda_n \ge A_n \ge A_{n+1}, n \ge 1. \tag{11}$$

Тепер визначимо безумовну щільність

$\tilde{p}(x_1^{n+1}) = \Pi_{01}\, \tilde{p}_1(x_1^{n+1}) \Pi_{00} p_0(x_1^{n+1})$, $n \ge 0$, $n$ відповідну умовну щільність

$$p_{n+1}(x_{n+1} \mid x_1^n) = \tilde{p}\ (x_1^{n+1}) / \tilde{p}\ (x_1^n) =$$

$$= \frac{\vartheta\Lambda_{n+1} + 1}{\vartheta\Lambda_n + 1} p_{0n+1}(x_{n+1}) = \frac{1 - \Pi_n}{1 - \Pi_{n+1}} p_{0n+1}(x_{n+1}). \tag{12}$$

З (6), (10) та (12) слідує що

$$p_{n+1}(x_{n+1} \mid x_1^n) = p_{n+1}(x_{n+1} \mid \Lambda_n) = p_{n+1}(x_{n+1} \mid \Pi_n). \tag{13}$$

З (10) та (13) витікає, що системи $(\Pi_n, F_n^x \overline{P})$, $(\Lambda_n, F_n^x \overline{P})$, $n \ge 0$, де $F_n^x = \sigma(x_1^n) - \sigma$ - алгебра, яка породжена $x_1^n$, є марковськими неоднорідними випадковими функціями (див. лему 17 в [6]).

Також розглянемо статистичну $\tilde{\Pi}_n = P(0 \le \lambda_0 < n\Delta \mid x_1^n)$, що являє собою апостеріорну вірогідність наявності вторгнень до моменту $n\Delta$. Очевидно що

$$\tilde{\Pi}_n = \Pi_n P(\lambda_0 < n\Delta \mid 0 = 1, x_1^n). \tag{14}$$

Далі, оскільки подія $\{0 \le \lambda_0 < n\Delta\}$ еквівалентна події $\left\{\bigcup_1^n \mid \lambda_0 \in [(i-1)\Delta, i\Delta]\right\}$, причому

$\{\lambda_0 \in [(i-1)\Delta, i\Delta]\} \bigcap \{\lambda_0 \in [(i-1)\Delta, j\Delta]\} = \emptyset$, то

$$P(\lambda_0 < n\Delta \mid \theta = 1, x_1^n) = \sum_{i=1}^{n} P\{(i-1)\Delta \le \lambda_0 < i\Delta \mid \theta = 1\}. \quad (15)$$

Використовуючи формулу для умовної ймовірності та(1)-(3) отримуємо:

$$P\{(i-1)\Delta \le \lambda_0 < i\Delta \mid x_i^n, \theta = 1\} = \begin{cases} \beta_i(x_i)\prod_{i+1}^{n}\gamma_S(x_S)/\Lambda_n, i = \overline{1,n}, \\ \alpha_i / \Lambda_n, i \ge n+1, \end{cases} \quad (16)$$

де $\alpha_i = P\{(i+1)\Delta \le \lambda_0 < i\Delta \mid \theta = 1\}$.

З (15), (16) та (7) виходить, що

$$P\{\lambda_0 < n\Delta \mid \theta = 1, x_1^n\} = (\Lambda_n - A_n)/\Lambda_n, \; n > 0 \quad (17)$$

Комбінуємо (15), (16), та (6):

$$\tilde{\Pi}_n = \vartheta(\Lambda_n - A_n)/(1 - \vartheta\Lambda_n), n \ge 0; \quad (18)$$

$$\tilde{\Pi}_n = \Pi_n - \vartheta A_n/(1-\Pi_n), n \ge 0, \quad (19)$$

Використовуючи (18), (10) отримуємо рекурентне співвідношення для статистики $\tilde{\Pi}_n$:

$$\tilde{\Pi}_{n+1} = \frac{\tilde{\Pi}_n\gamma_{n+1}(x_{n+1}) + \vartheta\beta_{n+1}(x_{n+1})(1-\tilde{\Pi}_n)/(1+\vartheta A_n)}{\tilde{\Pi}_n\gamma_{n+1}(x_{n+1}) + \{1+\vartheta[A_{n+1}+\beta_{n+1}(x_{n+1})]\}(1-\tilde{\Pi}_n)/(1+\vartheta A_n)}, n \ge 0, \tilde{\Pi}_n = 0, \quad (20)$$

звідси випливає, що система $(\tilde{\Pi}_n, F_n^x, \overline{P}), n \ge 0,$ також є марківською випадковою функцією, (рівність (13) справедлива для $\tilde{\Pi}_n$).

У разі стрибкоподібного ІВ, коли $p_{\lambda n}(x_n) = p_{1n}(x_n)$ для всіх $(n-1)\Delta \le \lambda < n\Delta$, або дискретного розподілення $\lambda_0$

$$(\Pi(\lambda) = \sum_{i=0}^{k(\lambda)} \alpha_i I(\lambda - i\Delta), k(\lambda) = \min\{i : \lambda \ge i\Delta\},$$

$$I(\lambda) = 1 \text{ при } \lambda \ge 0, \text{ I}(\lambda)=0 \text{ при } \lambda < 0), \quad (21)$$

$$\beta_n(x_n) = \gamma_n(x_n)\alpha_n, \; n \ge 1.$$

Підставляючи (21) в (10) та (20) отримуємо

$$\Lambda_{n+1} = A_{n+1} + \gamma_{n+1}(x_{n+1})(\Lambda_n - A_{n+1}), \; n \ge 0, \; \Lambda_0 = 1, \quad (22)$$

$$\tilde{\Pi}_{n+1} = \frac{\gamma_{n+1}(x_{n+1})\left[\tilde{\Pi}_{n+1} + (1-\tilde{\Pi}_n)B_{n+1}\right]}{\left[\tilde{\Pi}_n + (1-\tilde{\Pi}_n)B_{n+1}\right]\gamma_{n+1}(x_{n+1}) + (1-B_{n+1})(1-\tilde{\Pi}_k)}, n \ge 0, \tilde{\Pi}_0 = 0, \quad (23)$$

де $B_{n+1} = \vartheta\alpha_{\partial n+1}/(1+\vartheta A_n)$.

В окремому випадку коли спостереження розподілені $p_{0n}(x_n) = p_0(x_n)$, $p_{1n}(x_n) = p_1(x_n), x \ge 1$ апріорна ймовірність наявності ІВ $\Pi_{01} = 1$ і апріорне розподілення ймовірностей $\lambda_0$ є геометричним:

$$P(\lambda_0 = \lambda) = \begin{cases} \alpha_0, \lambda = 0, \\ (1-\alpha_0)\alpha(1-\alpha)^{i-1}, \lambda = i\Delta, i \ge 1, \end{cases} \quad (24)$$

$$P\{\lambda_0 \in (i\Delta, (i+1)\Delta)\} = 0, \; i \ge 0.$$

Система $(\tilde{\Pi}_n, F_n^x, \overline{P}), n \ge 0$ виявляється однорідною, так як $\gamma_{n+1}(x_{n+1}) = \gamma(x_{n+1}), B_{n+1} = \alpha_{n+1}/A_n = \alpha = const$.

Однак якщо $\Pi_{01} < 1$, то як неважко побачити, система $(\tilde{\Pi}_n, F_n^x, \overline{P}), n \geq 0$, виявляється неоднорідною. Для того щоб вона була однорідною, необхідно виконання умови $B_n = B = const$, $n \geq 0$ тобто щоб $\vartheta A_n(1-B) - \vartheta A_{n+1} - B = 0$, $n \geq 0$, звідки слідує, що

$$A_n = (A_0 + 1/\vartheta)(1-B)^n - 1/\vartheta, \ n \geq 0. \tag{25}$$

Розподілення, що відповідає (25), не є імовірнісним, оскільки

$$\lim_{n\to\infty} A_n = \begin{cases} -\vartheta^{-1}, \ B \in (0,1], \\ \infty, \ B \leq 0, \\ -\infty, \ B > 1. \end{cases}$$

Таким чином, не існує ймовірного апріорного розподілення $\lambda_0$, при якому система $(\tilde{\Pi}_n, F_n^x, \overline{P}), n \geq 0$, являє собою однорідну марківську однорідну випадкову функцію, якщо ІВ з'являється з ймовірністю, менша за 1.

Системи $(\Lambda_n, F_n^x, \overline{P}), (\Pi_n, F_n^x, \overline{P}), n \geq 0$, також є неоднорідними марківськими функціями при будь яких апріорних розподіленнях $\lambda_0$. Для $(\Lambda_n, F_n^x, \overline{P})$ це безпосередньо випливає з (10), (22). При $\Pi_n$ втрачає смисл $(\Pi_n \equiv 1, n \geq 0)$, а $\tilde{\Pi}_n$ зв'язна з УОП $\Lambda_n = \int_0^\infty \left[ p(x_1^n \mid \lambda_0 = \lambda) \Big/ (x_1^n \mid \lambda_0 \geq n\Delta) \right] d\Pi(\lambda)$ рівністю (17).

Надалі зручно користуватися статистикою $L_n$ яка зв'язана з УОП $\Lambda_n$ рівністю

$$L_n = \Lambda_n - A_n. \tag{26}$$

З (10), (22) витікає, що для $L_n$ мають місце рекурентні співвідношення

$$L_{n+1} = \beta_{n+1}(x_{n+1}) + \gamma_{n+1}(x_{n+1})L_n, \ n \geq 0, \ L_0 = 0; \tag{27}$$

$$L_{n+1} = \gamma_{n+1}(x_{n+1})(\alpha_{n+1} + L_n), \ n \geq 0, \ L_0 = 0; \tag{28}$$

у випадку поступового та стрибкоподібного ІВ відповідно.

З (28) слідує, що $(L_n, F_n^x, \overline{P})$, $n \geq 0$, являється однорідною марківською функцією у випадку однорідних спостережень, стрибкоподібних ІВ з рівномірного розподілу $\lambda_0$ на інтервалі $(0,T]$ $(\alpha_n = \Delta/T, T > N\Delta)$.

Позначимо $\tilde{\Pi}_n = P(\lambda_0 < i\Delta \mid x_1^n), i \geq 1$, апостеріорну вірогідність наявності вторгнень до миті $i\Delta$; $M(\bullet \mid x_1^n)$ - умовне математичне очікування відносно розподілення з щільністю $p_{n+1}(x_{n+1} \mid x_1^n)$.

Лема. Статистика $\tilde{\Pi}_{in}$ є мартингалом

$$M(\tilde{\Pi}_{in+1} \mid x_1^n) = \tilde{\Pi}_{in}, \ n \geq 0; \tag{29}$$

$$M\tilde{\Pi}_{in} < \infty, \ n \geq 0. \tag{30}$$

Доведення. Справедливість (30) очевидна. Далі, зрозуміло, що

$$\tilde{\Pi}_{in|} = \sum_{j=1}^{i} \Pi_{jn}, \ n \geq 1, \tag{31}$$

де $\Pi_{jn} = P\{(j-1)\Delta \leq \lambda_0 < j\Delta \mid x_1^n\}, n \geq 1$.

Використовуючи (16), (6) і той факт, що $\Pi_{in} = P\{\lambda_0 \in [(i-1)\Delta, i\Delta) \mid x_1^n, \theta = 1\}, \Pi_n, i \geq 1$, отримуємо

$$\Pi_{in}=\begin{cases}\vartheta\beta_i(x_i)\prod_{i+1}^{n}\gamma_s(x_s)/(1+\vartheta\Lambda_n), i\le n,\\ \vartheta\alpha_i/(1+\vartheta\Lambda_n), i>n.\end{cases} \tag{32}$$

з урахуванням (32) та (12) знаходимо

$M(\Pi_{in+1}\mid x_1^n)=\Pi_{in}$, $n\ge 0, i\ge 1$, що разом з (31) доводить лему.

Лема дозволяє встановлювати достатність послідовності $\{L_n, n\ge 1\}$, (або $\{\tilde{\Pi}_n, n\ge 1\}$) в послідовності виявлення ІВ при функції втрат (4), (5). Дійсно, як випливає із загальних результатів [7] функція найменшого апостеріорного ризику (НАР) у розглянутій задачі задовольняє співвідношення

$R_n^N(x_1^n)=\min\{R_{n1}(x_1^n), R_{n0}^N(x_1^n)\}, n=\overline{1,N}$, де $R_n^N(x_1^n)-AP$, зв'язаний з рішенням $U_n=0$ тобто. з продовженням спостережень на $n$ кроці), що задовольняє рекурентному співвідношенню з [9], причому $R_{N0}^N=R_{N1}$, $R_{N1}, R_{N0}-AP$, зв'язані з прийняттям рішень $u_n=1$, $u_n=0$.

Використовуючи (4)-(6), (18), (19), після перетворень отримуємо

$$R_{n1}(x_1^n)=\Gamma_{n1}(L_n)+C\sum_{i=0}^{n-1}(n-1)\Pi_{i+1n};$$

$$R_{N0}(x_1^N)=\Gamma_{N1}(L_N)+C\sum_{i=0}^{N-1}(N-1)\Pi_{i+1N}, \text{ де}$$

$$\Gamma_{n1}(L_n)=(1+\vartheta\Lambda_n)^{-1}\{\vartheta L_n g_{11}(n)+\vartheta A_n\tilde{g}_{11}(n)+g_{01}(n)\}, \tag{33}$$

$$\Gamma_{N0}(L_N)=(1+\vartheta\Lambda_n)^{-1}\{\vartheta L_N g_{10}(N)+\vartheta A_N\tilde{g}_{10}(N)+g_{00}(N)\}. \tag{34}$$

Далі неважко показати, що

$$\sum_{i=0}^{n-1}(n-1)\Pi_{i+1n}=\sum_{i=0}^{n-1}\tilde{\Pi}_{i+1n}.$$

Тому

$$R_{n1}(x_1^n)=\Gamma_{n1}(L_n)+C\sum_{i=0}^{n-1}\tilde{\Pi}_{i+1n}. \tag{35}$$

$$R_{N0}(x_1^N)=\Gamma_{N0}(L_N)+C\sum_{i=0}^{N-1}\tilde{\Pi}_{i+1N}. \tag{36}$$

З (33)-(36) випливає, що на $N$-м кроці оптимальне правило має вигляд

$$=\begin{cases}1, L_N\ge L_N^0,\\ 0, L_N<L_N^0,\end{cases} \tag{37}$$

де $L_N^0=\dfrac{g_{01}(N)-g_{00}(N)+\vartheta A_N\left[\tilde{g}_{11}(N)-\tilde{g}_{10}(N)\right]}{\vartheta\left[g_{10}(N)-g_{11}(N)\right]}$. (38)

Для знаходження структури оптимального послідовного правила яке користується методом зворотної індукції та рекурентним ставленням із [9]. Враховуючи (13), (26), транзитивність статистики $L_N$ (з урахуванням (27)), рівності (35), (36) та лему на $(N-1)$-му кроці маємо

$$R_{n-10}^N(x_1^{N-1})=\int_{X_N}\min_{j=0,1}\Gamma_{Nj}\left[L_N(x_N, L_{N-1})\right]pN(x_N\mid L_{N-1})dx_N+C\sum_{i=0}^{N-1}\tilde{\Pi}_{i+1\,N-1}, \tag{39}$$

причому

$$\tilde{\Pi}_{n\,n-1}=\tilde{\Pi}_{n-1}+\tilde{\Pi}_{n\,n-1}=V(L_{n-1}+L_n)/(1+V\Lambda_{n-1}). \tag{40}$$

З (39) та (40) випливає що

$$R^N_{N-10}(x_1^{N-1}) = \Gamma^N_{\le N-10}(L_{N-1}) + C\sum_{i=0}^{N-2} \tilde{\Pi}_{i+1\ N-1}\ , \tag{41}$$

де

$$\Gamma^N_{N-10}(L_{N-1}) = \int\limits_{X_n} \min_{j=0,1} \Gamma_{Nj}\left[L_N(x_N, L_{N-1}) \mid p_N(x_N \mid L_{N-1}) \times dx_N + c\vartheta(L_{N-1} + \alpha_N)/(1+\vartheta\Lambda_{N-1})\right.$$

$$\Lambda_{N-1} = L_{N-1} + A_{N-1}.$$

Для $n = N-2, N-3$ з урахуванням співвідношень (29), (40), (41), (35) можна показати, що на довільному $n$ - му кроці має місце рівність

$$R^N_{N-10}(x_1^n) = \Gamma^N_{n0}(L_n) + C\sum_{i=0}^{n-1} \tilde{\Pi}_{i+1n}, n = \overline{1,N}\ , \tag{42}$$

де

$$\Gamma^N_{n0}(L_n) = \int\limits_{x_{n+1}} \min\left\{\Gamma_{n+11}\left[L_{n+1}(x_{n+1}, L_n)\right], \Gamma^N_{n+10}\left[L_{n+1}(x_{n+1}, L_n)\right]\right\} p_{n+1}(x_{n+1} \mid L_n) dx_{n+1} +$$

$$+ c\vartheta(L_n + \alpha_{n+1})/(1+\overline{\vartheta}\Lambda_n), n = \overline{1, N-1} \tag{43}$$

Таким чином, з (20), (35), (42) випливає, що оптимальне правило ІВ має вигляд

$\tau^0_N(L) = \min\left\{1 \le n \le N : L_n \notin \vartheta^N_{nn}\right\}$, де $\vartheta^N_{nn} = \left\{L_n : \Gamma^N_{n0}(L_n) < \Gamma_{n1}(L_n)\right\}, n = \overline{1,N}$ є достатньою (відповідно до (37)). Перехід в (33), (43) та статистики $\tilde{\Pi}_n$ за допомогою рівності (18) та подальше дослідження показує, що $\Gamma^N_{n0}(\tilde{\Pi}_n)$ є вигнутою і, отже (через обмеженість), безперервною функцією $\tilde{\Pi}_n$ на інтервалі [0,1]. Функція $\Gamma^N_{n0}(\tilde{\Pi}_n)$ є лінійною:

$$\Gamma_{n1}(\tilde{\Pi}_n) = \tilde{\Pi}_n\, g_{11}(n) + (1-\tilde{\Pi}_n)\left\{\tilde{g}_{11}(n) + \left[g_{01}(n) - \tilde{g}_{11}(n)\right] : (1+\vartheta A_n)\right\}.$$

Типова залежність $\Gamma_{n1}, \Gamma^N_{n0}$ від $\tilde{\Pi}_n$ наведено на рис.1. З малюнка випливає, що $\vartheta^N_{nn} = \left\{L_n < L^0_n\right\}$, де корінь рівняння

$$\Gamma^N_{n1}(y) = \Gamma^N_{n0}(y), n = \overline{1, N-1}, \tag{44}$$

а значить

$$\tau^0_N(L) = \min\left\{N, \min\left[n \ge 1 : L_n \ge L^0_n\right]\right\}, \tag{45}$$

($L^0_n$, довільно при $n \ge N+1$)

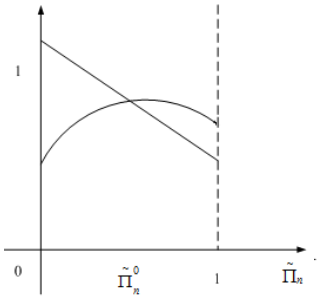



**Рис.1.** Типові залежності функції $\Gamma_{n1}$ та $\Gamma^N_{n0}$ від статистики $\tilde{\Pi}_n$

Послідовне правило виявлення ІВ відповідне (45) має вигляд

$$u_{n0}^{0}(L_n)=\begin{cases}1, L_n \geq L_n^0,\\ 0, L_n < L_n^0, n=\overline{1,N},\end{cases} \tag{46}$$

де при $n \leq N-1$ $L_n^0$ визначається з (44), а $L_N^0$ - співвідношенням (38).

З метою подальшої деталізації вважаємо, що для $L_N$ має місце рекурентне співвідношення (28). Строго кажучи, це справедливо при стрибкоподібному вторгненні [4].

Використовуючи (33), (34), (43), (46), (12), після перетворень на $(N-1)$-м кроці отримуємо

$$\Gamma_{N-10}^{N}(L_{N-1})=(1+\vartheta\Lambda_{N-1})^{-1}\left\{\vartheta(L_{N-1}+\alpha_N)\times\left[\sum_{j=0}^{1}g_{1j}(N)P_{1j}^{(1)}(L_{N-1})+c\right]\right.,$$
$$\left.+\left[\sum_{j=0}^{1}\vartheta A_N g_{1j}(N)+g_{0j}(N)\right]P_{0j}^{(1)}(L_{N-1})\right\}$$

де $P_{ij}^{(1)}(L_{\partial N-1})=\int\limits_{X_N^j}P_{iN}(x_N)dx_{Ni}$ ;

$X_N^0=\left\{x_N : L_N(x_{N_1}L_{N-1})<L_N^0\right\}$;

$X_N^1=\left\{x_N : L_N(x_N L_{N-1})\geq L_N^0\right\}$.

Розглядаючи $(N-2)$, $(N-3)$-й та інші кроки і застосовуючи (43), (46), по індукції можна показати, що на довільному $n$-му кроці

$$\begin{aligned}&\Gamma_{no}^{N}(L_n)=(1+\vartheta\Lambda_n)^{-1}\{\sum_{\upsilon=1}^{N-n}(\vartheta L_n[g_{11}(n+v)P_{11}^{(\upsilon)}(L_n)+\\&+cP_{10}^{(\upsilon-1)}(L_n)]+[\vartheta\mathrm{A}_n+\upsilon\tilde{g}_{11}(n+v)+g_{01}(n+v)]P_{01}^{(\upsilon)}(L_n))+\\&+\vartheta L_n g_{10}(N)P_{10}^{(N-n)}(L_n)+[\vartheta\mathrm{A}_N\tilde{g}_{10}(N)+g_{00}(N)\times\\&\times P_{01}^{(v)}(L_n)+\vartheta\sum_{S=1}^{N-n}\left[\alpha_{n+S}\sum_{v=S}g_n(n+v)P_1^{(S-1,v)}(L_n)\right]+\\&+\vartheta c\sum_{S=1}^{N-n-1}\alpha_{n+S}\left[P_{00}^{(S-1)}(L_n)+\sum_{v=S}^{N-n-1}P_0^{(S-1,v)}(L_n)\right]+\\&+\vartheta c\alpha_N P_{00}^{(N-n-1)}(L_n)\},n=\overline{1,N-1}\end{aligned}, \tag{47}$$

де $(v)$

$$P_{ij}(L_n)=\int P_{ij}^{(v-1)}(L_{n+1}(x_{n+1},L_n))_{P_{in+1}}(x_{n+1})dx_{n+1}, v\geq 2; \tag{48}$$

$$P_{ij}(L_n)=\int\limits_{x_{n+1}^j}p_{in+1}(x_{n+1})dx_{n+1}, i,j=0,1; \tag{49}$$

$$P_j^{(S,v)}(L_n)=\int\limits_{x_{n+1}^0}P_j^{(S-1,v-1)}(L_{n+1}(x_{n+1},L_n))P_{0n+1}(x_{n+1})dx_{n+1}, S\geq 1, v\geq 2, S<v; \tag{50}$$

$$P_j^{(0,v)}(L_{n+1})=P_{1j}^{(v)}(L_n), v\geq 1; P_{10}^{(0)}=P_{00}^{(0)}=1; \tag{51}$$

$$X_{n+1}^{(0)}=\left\{x_{n+1}:\gamma_{n+1}(x_{n+1})<\left(L_{n+1}^0\middle/ L_n+\alpha_{n+1}\right)\right\}; \tag{52}$$

$$X_{n+1}^{'}=\left\{x_{n+1}:\gamma_{n+1}(x_{n+1})<\left(L_{n+1}^0\middle/ L_n+\alpha_{n+1}\right)\right\};$$

(залежність $P_j^{(S,v)}$ від $n,N$ тут і далі не виписуємо);

функції $P_j^{(S,v)}(L)(50)$ в області $[0,L_n^0]$ являє собою умовні ймовірності прийняття $i$-го рішення на $(n+v)$-му кроці за умови спостереження $L_n = L$ та появи ІВ на інтервалі $\left[(n+s)\Delta,(n+s+1)\Delta\right]$;

$$P_j^{(S,v)}(L) = P\left\{u_{n+v} = \frac{j}{L_n} = L,\ \lambda_0 \in [(n+s)\Delta,(n+s=1)\Delta)\right\} L \in [0,L_n^0).$$

Поріг $L_n^0 = L_n^0(N)$ визначається з рівняння (44) і при заданні видів густин $P_{0n}, P_{in}$ в принципі може бути знайдено з використанням (47)-(52), (33). У разі щільності $P_{1n}, P_{0n}, n = \overline{1,N}$, що сильно розрізняються, можна зробити висновки без котеретизації $P_{1n}, P_{0n}$. Введемо параметр $q_n$, що характеризує ступінь помітності $P_{0n}, P_{1n}$, такий, що з його збільшенні $P_{0n}, P_{1n}$, дедалі більше різняться і за $q_n \to \infty$ [4]

$P_{ij}^{(v)}(L) \to 0, v \ge 1, i \ne j$;

$P_{ij}^{(s,v)}(L) \to 0, v \ge 2, s < v-1, i, j = 0,1$;

$P_0^{(v-1;v)}(L) \to 0$;

$P_{ij}^{(v)}(L) \to 0, v \ge 2$;

$P_1^{(v-1;v)}(L) \to 1, v \ge 2$;

$P_{ii}^{(1)}(L) \to 1, (L < L_n^0)$. (53)

Вважаючи відповідні функції (53) рівними своїм граничним значенням і використовуючи (47), (33), (44), неважко показати, що

$$\tilde{L}_n^0(\infty) = \frac{g_{01}(n)^{\vartheta^{-1}} + An\tilde{g}_{11}(n) - \sum_{S=2}^{N-n}\alpha_{n+S}g_{11}(n+S) - \alpha_{n+1}g_{11}(n)}{g_{11}(n+1) - g_{11}(n) + c}, \quad (54)$$

$$\tilde{L}_{n-1}^0(\infty) = \frac{[g_{01}(N-1) - g_{00}(N)]\vartheta^{-1} + A_{N-1}\tilde{g}_{11}(N-1) - A_N\tilde{g}_{10}(N) - \alpha_N g_{11}(N-1)}{g_{11}(N) - g_{11}(N-1) + c}, \quad (55)$$

де $\tilde{L}_n^0(\infty) = \lim_{q_n \to \infty} \tilde{L}_n^0(q_n)$; $\tilde{L}_n^0(q_n) = \tilde{L}_n^0(q_n) + \alpha_{n+1}$.

У граничному випадку $q_n \to \infty$ оптимальні пороги менші, ніж значення, одержувані за допомогою (54) і (55).

Тепер розглянемо випадок $\Pi_{01} = 1$ та неусічену послідовну процедуру. Втрат $g_n(n)$ в (4) відсутні. Припустимо

$$\tilde{g}_{11} = \varphi + c_1 n;\ \tilde{g}_{11}(n) = c_1 n, \quad (56)$$

де $c_1$ - вартість одного кроку спостереження; $\varphi_0$ - Втрати пов'язані з помилковою тривогою. У цьому випадку виявляється зручніше користуватися достатньою статистикою $\tilde{\Pi}_1$, що дорівнює апостеріорній ймовірності ІВ до моменту $n\Delta$ (див. (23)).

Враховуючи (18), (54), (56) при $\Pi_{01} = 1$, $N \to \infty$, отримуємо, що оптимальне граничне значення порога в несіченій процедурі у просторі $\tilde{\Pi}_n$

$$\tilde{\Pi}_n^0(\infty)=\frac{\varphi_0-c_1\sum\limits_{s=1}^{\infty}sB_{n+s}-cB_{n+1}}{\varphi_0+c_1(1-\sum\limits_{s=1}^{\infty}sB_{n+s})+(1-B_{n+1})}, \tag{57}$$

де $B_{n+s}=\alpha_{n+s}\big/A_n$

зокрема, при геометричному розподілі $\lambda_0$ (24)

$$B_{n+s}=\alpha(1-\alpha)^{S-1};\ \sum_{s=1}^{\infty}sB_{n+s}=\alpha^{-1} \tag{58}$$

і з (57), (58) випливає, що $\tilde{\Pi}_n^0$ не залежить від $n$:

$$\tilde{\Pi}_n^0(\infty)=\tilde{\Pi}^0(\infty)=\frac{\varphi_0-c_1\big/\alpha-c\alpha}{\varphi_0-c_1\dfrac{(1-\alpha)}{\big/\alpha+c(1-\alpha)}},n\geq 1. \tag{59}$$

У разі однаково розподілених спостережень при втратах виду (56) та $N\to\infty$ від номера $n$ не залежить не тільки граничне значення порога $\tilde{\Pi}^0(\infty)$, а й додаткове $\tilde{\Pi}^0(q)$, що є наслідком однорідності послідовності $(\tilde{\Pi}_n,F_n^x,P),n\geq 0$.

Однак при $\Pi_{01}<1$ оптимальний поріг залежить від номера кроку при будь-якому апріорному розподілі $\lambda_0$, оскільки послідовність $(\tilde{\Pi}_n,F_n^x,P),n\geq 0$ неоднорідна.

Якщо залежність вартість затримки у прийнятті рішення про наявність ІВ від $n-\lambda$ нелінійна, то статистика $L_n$ (і відповідно $\tilde{\Pi}_n,\Lambda_n$) не є достатньою в задачі послідовного виявлення. При знаходженні достатніх статистик у разі можна використовувати методику [6,9,10]. У задачі непослідовного виявлення оптимальне правило має вигляд (37), (38) при довільній залежності вартості затримки у прийнятті рішення про наявність ІВ від $n$ [4].

Досі розглядалося байєсівське завдання виявлення ІВ. Зокрема, було встановлено, що якщо вторгнення рано чи пізно відбудеться (тобто $\Pi_{01}=1$), спостереження незалежні та однорідні, апріорний розподіл моменту ІВ геометричний (див. (24)) функція тепер має вигляд (4), (56), то в силу того, що апріорна ймовірність ІВ $\left\{\tilde{\Pi}_n,\tilde{F}_n^x,P\right\},n\geq 0$, є однорідним Марківським процесом, оптимальне не усічене послідовне правило виявлення має вигляд

$$u_n^0(\tilde{\Pi}_n)=\begin{cases}1,\tilde{\Pi}_n\geq\Pi^0,\\ 0,\ \tilde{\Pi}_n\geq\Pi^0,\tilde{n}\geq 1.\end{cases} \tag{60}$$

Тут $\tilde{\Pi}_n\geq\tilde{\Pi}^0(q)$ постійний поріг, залежний від параметра (), що характеризує ступінь помітності щільностей $p_1(x),p_0(x)$ і величин $\varphi_0,c_1,c$, що є відповідно втрат при першій помилковій тривозі, вартістю одиниці спостережень і вартістю затримки у виявленні вторгнення на один крок. При сильно розрізняються щільності (більших за $q$) близьким до оптимального буде поріг (59).

Задаємо допустиму ймовірність помилкової тривоги $\varphi_0 \in (0,1)$ і позначимо через $\Delta(\alpha)$ клас правил виявлення $u(x)$, котрим ймовірність помилкової тривоги вбирається у $\alpha$, тобто $P(\tau(u) < \lambda_0)$, де $\tau(u)$ - момент вторгнення, що відповідає правилу $u(x)$; $P$ - міра, що відповідає $(x_n \lambda_0), n \geq 1$. Тоді, зводячи умовно експериментальну задачу до байссовской, можна показати [6], що правило (60) при виборі порога $\tilde{\Pi}_\alpha^0$ таким чином, що $P(\tau_0 < \lambda_0) = \alpha$ мінімізує в класі $\Delta(\alpha)$ середню затримку $M(\tau - \lambda_0)$ у виявленні вторгнення:

$$M(\tau - \lambda_0)^+ = \inf_{u(x) \in \Delta(\alpha)} M(\tau(u) - \lambda_0)^+ .$$

Тут $M$ – символ усереднення у міру $P; (y-z)^+ = \max(y-z, 0); \tau_0 = \inf\left\{ n : \tilde{\Pi}_n \geq \tilde{\Pi}_\alpha^0 \right\}$ - момент вторгнення, що відповідає правилу (60). Оскільки $M(\tau - \lambda_0)^+ = P(\tau \geq \lambda_0) M(\tau - \lambda_0 \mid \tau \geq \lambda_0)$, то правила (60) мінімізує також клас $\Delta(\alpha)$ математичного очікування $M(\tau - \lambda_0 \mid \tau \geq \lambda_0)$. Оптимальний поріг $\tilde{\Pi}_\alpha^0 = \tilde{\Pi}_\alpha^0(q)$ залежить від ступеня помітності щільностей $p_1(x), p_0(x)$, та його точне визначення є складним завданням. Однак, зауважуючи, що $\alpha = P(\tau_0 < \lambda) = M(1 - \tilde{\Pi}_{\tau_0}), M\tilde{\Pi}_{\tau_0}(q)$ отримуємо $\tilde{\Pi}_\alpha^0(q) \leq 1 - \alpha$.

Якщо в якості порога використовувати верхню межу $1-\alpha$ ймовірність помилкової тривоги у такій процедурі $\alpha(u) \leq \alpha$, оскільки $\alpha(u) = M(1 - \tilde{\Pi}_{\tau(u)})$, $M\tilde{\Pi}_{\tau(u)} \geq 1 - \alpha$, $\tau(u) = \inf\left\{ n : \tilde{\Pi}_n \geq 1 - \alpha \right\}$. Значення $\alpha(u)$ буде близько до $\alpha$ тільки при малих перескоках порога величиною $\tilde{\Pi}_{\tau(u)}$, що може бути виконано при "близьких" щільностях (малих $q$) і малих $\alpha$.

Правило (60) оптимально як і байєсівське, і у умовно екстремальному завданні лише за геометричному апріорному розподілі $\lambda_0$ (як і (24)). Якщо це розподіл інше то статистика $\tilde{\Pi}_n, n \geq 0$, виявляється неоднорідної функцією і оптимальний поріг залежить від часу. При цьому невідомо, чи є правило (60) близьким до оптимального хоча б асимптоматично при великому з середнього часу виявленні ВВ у малих $\alpha$. У той же час лема, що для критичних додатків правило (60) з постійним порогом може становити інтерес тільки в тому випадку, якщо воно близьке до оптимального при слабких реченнях характеру апріорного розподілення. Проте вдається побудувати модифікацію правила (60), яка виявляється оптимальною без введення апріорного розподілу моменту $\lambda_0$ [11].

Нехай $\lambda \in \{1, 2, 3, ..., \infty\}$ – невідомий момент збою, про властивості якого робиться жодних припущень. Через $M_\partial$ позначимо математичне очікування, що відповідає розподілу спостережень при фіксованому $\lambda$, через $M_\infty$ – те ж очікування при $\lambda = \infty$, через $\Delta(T)$ – клас правил $M_\infty \tau(u) \geq T$, де $T$ – задана компонента $(T < \infty)$, що характеризує обмеження на середній час до хибної тривоги. Бажано, щоб середній час до хибної тривоги було якомога більше, а середній час затримки у виявленні збою, що вимірюється величиною

$\tilde{\tau}_\lambda(u) = M_\lambda(\tau(u) - \lambda \mid \tau(u) \ge \lambda$, – якнайменше для всіх $1 \le \lambda < \infty$. Однак такого поступово кращого правила не існує. Тому мінімальне правило, що мінімізує величину $\sup \tilde{\tau}(u) 1 \le \lambda < \infty$ до класу $\Delta(T)$. Побудова такого правила заснованого на тій ідеї, що якщо правило є узагальненим байєсівським та еквівалентним, то воно мінімальне [12] (тобто правило є байєсівським щодо деякого невласного розподілу і виходить шляхом граничного переходу від послідовності байєсівських правил і $\tilde{\tau}_\lambda$ йому залежить від $\lambda$.

Використовуючи (15), (16), (21), неважко показати, що правило (60) еквівалентно порівнянню порогом статистики $\varphi_{n,d} = \sum_{k=1}^{n} \prod_{i=k}^{n} \left[ \gamma(x_i) \Big/ (1-\alpha) \right]$, яка є нормованою на величину $\lambda(1-\alpha)^{n-k}$ статистикою $L_n$. У межі при $\alpha \to 0$ статистика

$$\varphi_{n,\alpha} \to \varphi_n = \sum_{k=1}^{n} \prod_{i=k} \gamma(x_i).$$

Розглянемо правило

$$u_n^A(\varphi_n) = \begin{cases} 1, \varphi_n \ge A, \\ 0, \varphi_n < A, n \ge 1, \end{cases} \qquad (61)$$

де А деякий поріг, а статистика $\lambda_n$ задовольняє рекурентному співвідношенню

$$\varphi_{n+1} = \gamma(x_{n+1})(1 + \varphi_n), \ \varphi = 0, \qquad (62)$$

і є аналогом об'єкта прогнозу , усередненого на інтервалі $\overline{1,n}$ по невласному рівномірному розподілу, що приписує всім точкам числової прямої ваги, рівні 1 $(\gamma(x) = p_1(x) \Big/ p_0(x))$. Позначимо через $\tau_A = \inf\{n \ge 1 : \varphi_n \ge A(T)\}$ можливе виявлення вторгнення, що відповідає правилу (61), в якому поріг $A = A(T)$ вибирається таким чином, що $M_{00}\tau_A = T$. З (62) слід, що $M_{00}(\varphi_{n+1} \mid x_1^n) = 1 + \varphi_n$, отже $\varphi_n - n$ є маргіналом з нульовим середнім і $M_\infty(\varphi_{\tau A} - \tau_A) = 0$. Якщо не зважати на перескок порога статистикою $\varphi_{\tau A}$, то $A(T) \approx T$, хоча ця оцінка може бути грубою. Принаймні вважаючи $A(T) \approx T$, маємо $M_\infty \tau_t \ge T$, тобто $u^T(\varphi) \in \Delta(T)$. Правило (61), проте, є мінімаксним, оскільки

$$\tau_\lambda(u^T(\varphi)) = M(\tau_{A(T)} - \lambda \mid \tau_{A(T)} \ge \lambda = \int_0^{A(T)} M_\lambda(\tau_{A(T)} - \lambda \mid \tau_{A(T)} \ge \lambda, \varphi_{\lambda-1} = \varphi) \times P(d\varphi \mid \tau_{A(T)} \ge \lambda)$$

залежить від $\lambda$. Це з тим, що $\bar{\tau}_\lambda(\varphi_{\lambda-1}) = M_\lambda(\tau_{A(T)} - \lambda \mid \tau_{A(T)} \ge \lambda, \varphi_{\lambda-1})$ залежить від $\lambda$. Дійсно, при $\lambda = 1$ знаходимося зовсім в інших умовах, ніж при $\lambda \ge 2$, оскільки $\varphi_0 \equiv 0$, а $n \ge 1$, є випадковими величинами з розподілами залежать від $n$ (процес $\{\varphi_n\}$ однорідний, але нестаціонарний). Можна, однак, "підтримувати" статистику $\varphi_n$ рандомізацією в початковий момент, щоб виключити цей ефект. Більш точно, генеруватимемо величину $z$ відповідно до розподілу $\psi(y) = P(z \le y) = P_\infty(\tilde{\varphi}_n \le y \mid \tau_A^0 \ge n+1)$, де $P_\infty$ - міра, що відповідає спостереженням

при $\lambda = \infty$ (тобто за відсутності збоїв); $\left\{\tilde{\varphi}_n\right\}$- послідовність, що задовольняє рекурентному співвідношенню

$$\tilde{\varphi}_{n+1} = \gamma(x_{n+1})(1+\tilde{\varphi}_n), \tilde{\varphi}_0 = z; \tag{63}$$

$\tau_A^0 = \inf(n \geq 1 : \varphi_n \geq A)$ момент ІВ, що відповідає правилу виявлення $u(\tilde{\varphi})$ виду

$$u(\tilde{\varphi}) = \begin{cases} 1, \tilde{\varphi}_n \geq A, \\ 0, \tilde{\varphi}_n < A, n \geq 1 \end{cases}. \tag{64}$$

Таким чином, послідовність $\left\{\tilde{\varphi}_n\right\}$ відрізняється від $\left\{\varphi_n\right\}$тільки тим, що виходить не з нуля, а з випадкової точки $\tilde{\varphi}_n = z \in [0, A)$. Розподіл величини $z$ підбирається так, щоб послідовність $\tilde{\varphi}_n, n = 1,2,3,...$, на множинах $\{\tau_A^0 \geq n+1\}$ була однорідною. У цьому значення $\tilde{\tau}_\lambda^0(y) = M_\lambda(\tau_A^0 - \lambda \mid \tau_A^0 \geq \lambda, \tilde{\varphi}_{\lambda-1} = y)$ не залежить від $\lambda$, оскільки будь-якого $\lambda \geq 1$ при филесированому $\varphi_{\lambda-1}$ положення однаково. Отже, середній час запізнення у виявленні збою $\tilde{\tau}_\lambda^0 = \int_0^A \tau_\lambda^0(y) d\psi(y)$ також не залежить від $\lambda$ і правило (64) еквівалентно. Тому є природним претендентом на мінімаксне правило. Однак вдається довести лише асимптотичну мінімаксність при великих значеннях $T(II)$.

Теорема. Нехай розподіл величини $\gamma(x)$ безперервно і

$\int \max\{0, \gamma(x)\} p_1(x) dx < \infty$.

Тоді для кожного $T < \infty$ існують значення $A = A(T) < \infty$ і ймовірний захід $\psi_A$, такі, що $M_\infty \tau_A^0 = T$ і для будь-якого правила $u(x) \in \Delta(T)$ при $T \to \infty$

$$\sup_{1 \leq \lambda < \infty} M_\lambda(\tau(u) - \lambda \mid \tau(u) \geq \lambda) \geq \sup_{1 \leq \lambda < \infty} M_\lambda(\tau_A^0 - \lambda \mid \tau_A^0 \geq \lambda) + o(1)$$

З цієї теореми випливає майже мінімаксність правила (64), оскільки $\tilde{\tau}_\lambda^0 \equiv \tilde{\tau}_1^0(T) \to \infty$, решта члена $o(I) \to 0$ при $T \to \infty$.

Певні заходи $\psi_A$ є складним завданням. Зрозуміло, проте, що з $T \to \infty$ міра $\psi_A(y)$ збігається з граничним (стаціонарним) розподілом статистики $\varphi_n$ при $n \to \infty$. Для бернуллііївської послідовності та послідовності спостережень з показовим розподілом міри $\psi_A$ визначено у [11]. Оскільки $\tilde{\varphi}_n - n$ є мартингалом з математичним очікуванням $M_\infty \tilde{\varphi}_0$ (відповідно до (63)), то $M_\infty(\tilde{\varphi}_{\tau_A^0} - \tau_A^0) = M_\infty \tilde{\varphi}_0$ і поріг $A(T)$ визначається з рівняння $M_\infty \tilde{\varphi}_{\tau_A^0} = M_\infty \tilde{\varphi}_0 + T$.

Якщо як поріг взяти $\overline{A}(T) = T + M_\infty \tilde{\varphi}_0$, то $M_\infty \tau_A^0 \geq T$. Проте, по-перше, у своїй зростає $\tilde{\tau}_A^0$ і, по-друге, знайти $M_\infty \tilde{\varphi}_0 = \int_0^A y d\psi_A(y)$ можна лише після визначення міри $\psi_A$. Ці обставини породжують проблеми практичного використання побудованого мінімаксного правила (64). У той же час ясно, що принаймні при

великих значеннях $\lambda$ затримка $\bar{\tau}_\lambda$ у виявленні збою правилом (62) не повинна суттєво відрізнятись від $\bar{\tau}_1^0$ у випадку $T \to \infty$, оскільки зі збільшенням часу спостереження процес $\{\varphi_n\}$ можна вважати стаціонарним.

У разі експоненційного однопараметричного сімейства, коли вхідна (62) величина

$$\gamma(x) = \exp\{\theta x - b(\theta)\} = \gamma_\theta(x),$$

де $b(\theta)$ – опукла вниз функція, що має дві похідні $(b(\theta) = b'(0) = 0)$, асимптотичні властивості правила (61) при $T \to \infty$ досліджені в [11].

Введемо такі позначення: $M_{t,\lambda}$ – математичне очікування збою $\lambda$ та параметру $\theta = t$; $I_0 = M_{\theta,1}\left[\ln \gamma_\theta(x_1)\right] = \theta b'(\theta) - b(\theta)$ – інформаційна кількість Кульбану;

$N_b(\theta) = \inf\left\{n : \sum_{i=1}^{n} \ln \gamma_\theta(x_i) \ge B\right\}$, $\beta_\theta = \lim_{B\to\infty} M_{\theta,1} exp\left\{-\left[\sum_{n=1}^{N_B(\theta)} \ln \gamma_\theta(x_n) - B\right]\right\}$ – величина, що певним чином характеризує середній перескок порога; $E_{t,\theta} = \sup_{1\le\lambda<\infty} M_{t,\lambda}(\tau_A(\theta) - \lambda + 1 \mid \tau_A(\theta) \ge \lambda)$ – максимальне за моментом значення середньої затримки в її виявленні з правилами (61) вбудованого значення $\theta$, коли в дійсності параметр після збору рівня $t$. По очевидним причинам $E_{t,\theta} = M_{t,1}\tau_A$, і, відповідно, для оцінки $E_{t,\theta}$ потрібно оцінити останнє очікування. З [11] випливає, що якщо поріг $A$ в (61) вибрати за формулою $A = \beta_\theta T$ і $I_\theta < \infty$, $0 < tb'(\theta) - b(t) < \infty$ для $\theta \in \Theta$, то при $T \to \infty$ і $\theta \in \Theta(\Theta \subset R^1)$

$$M_\infty \tau_A(\theta) = T(1 + o(1)); \tag{65}$$

$$E_{t,\theta} = \frac{1}{tb'(\theta) - b(t)}\left[\ln T + C_{t,\theta} + o(1)\right], \tag{66}$$

де $C_{t,\theta}$ – деяка константа.

Значення $E_{t,\theta}$ зростає зі збільшенням різниці між $t$ і $\theta$. Тому, коли значення параметра $\theta$ щільності після збою заздалегідь невідоме, характеристики правила (61) можуть виявитися незадовольними. Істотно найкращі характеристики при цьому має правило

$$\tau = \inf\left\{n \ge 1 : \bar{\varphi}_n \ge T\beta_F\right\},$$

де $\bar{\varphi}_n = \int_\theta \varphi_n(\theta) dF(\theta)$ - усереднена за деякою мірою $F(\theta)$ статистика (62) $\beta_F = \int_\theta \beta_{\theta^{-1}} dF(\theta)$. Для цього правила за умови, що $F\{\theta : I_\theta < \infty\} = 1$ і $F'(\theta)$ позитивна і безперервна, справедливі наступні асимптотичні рівності [11]

$$M_\infty \tau = T(1 + o(1));$$

$$\bar{E}_t = I_t^{-1}\left[\ln T + \frac{1}{2}\ln b_2 T + C_{t,F} + o(1)\right], \qquad T \to \infty, \qquad \text{де}$$

$E_t = \sup_{\lambda\ge 1} M_{t,\lambda}(\tau - \lambda + 1 \mid \tau \ge \lambda) = M_{t,1}\tau$.

Таким чином, головний член асимптотичного розкладання ризику збігається з головним членом розкладання ризику для випадку, коли $\theta = t$ (точно відомо). Тому правило з усереднення асимптоматично оптимально для задачі з невідомим

значенням параметра щільності після збою. Наявність подвійного логарифму в розкладанні $E_t$ є платою за апріорну невизначеність.

Аналітичний результат може бути отриманий для модифікації правила, що з правила максимальної правдоподібності.

Правило виявлення (61) дискретного часу було запропоновано та досліджувалося методом Монте-Карло в [13]. Вперше (у разі безперервного часу) це правило розглянуто у [6]. В [14] досліджено послідовне правило виявлення Пейджа, засноване на порівнянні з порогом принципу максимальної правдоподібності. Це правило асимптотично оптимально при $T\to\infty$ (у сенсі першого порядку) у класі $\Delta(T)$ за ризиком $\sup ess \sup\limits_{\lambda\ge 1} M_\lambda\left[(\tau-\lambda+1)\,|\,x_1^{\lambda-1}\right]$, який для нього дорівнює $I^{-1}\ln T(1+o(1))$, де $I=M_1\left[\ln\gamma(x_1)\right]$ – інформаційна кількість Кульбака. У [15] встановлена строга мінімаксність цього правила у класі $\Delta(T)$ для всіх $T$.

Узагальнюємо отримані результати на випадок, коли виникнення збою даних $x_n$, $n>\lambda$, що спостерігаються, розподілені відповідно до одного з декількох розподілів $p_i(x_n)$, $i=\overline{1,M}$, $M\ge 2$ і необхідно визначити з яким. Бажано знайти таке правило, яке мінімізує у класі $\Delta(T)$ максимальні значення $\sup\limits_{\lambda\ge 1}\bar\tau_{i\lambda}=\sup\limits_{\lambda\ge 1}M_{i\lambda}(\tau-\lambda\,|\,\tau>\lambda)$ для всіх $i=\overline{1,M}$.

Однак знайти таке правило і навіть довести його існування за $M>1$ і кінцевого $T$ не вдається.

Тому зосередимо увагу на випадок $T\to\infty$. Розглянемо правило

$$u_n^*(\varphi_n)=\begin{cases} j, \varphi_{nj}\ge A_i(T), \\ O, \varphi_{nj}<A_i(T)=\overline{1,M}, n\ge 1, \end{cases} \tag{67}$$

де $\varphi_n=(\varphi_1,\varphi_2,\ldots\varphi_{nM})$; $\varphi_{ni}=\sum\limits_{k=1}^{n}\prod\limits_{S=k}^{n}\gamma_i(x_s)$; $\gamma_i(x)=p_i(x)\big/p_0(x)$. Компоненти статистики $\varphi_n$ задовольняють рекурентним співвідношенням з (62).

$$\varphi_{n+1i}=\gamma_i(x_{n+1})(1+\varphi_n),\ \varphi_{0i}=0,\ i=\overline{1,M}. \tag{68}$$

Момент ІВ $\tau^*=\tau(u^*)$, відповідає правилу (67)

$$\tau^*=\min_{i\in 1,M}\tau_i,\ \tau_i=\inf\left\{n:\varphi_{ni}\ge A_i(T)\right\}. \tag{69}$$

Зрозуміло що $(M_\infty\tau^*)^{-1}\le\sum\limits_{i=1}^{M}(M_\infty\tau_i)^{-1}$. Тому, якщо пороги $A_i(T)$ обрани таким чином, що $M_\infty\tau_i\ge MT$, то $(c_i\ge 1)$

$$M_\infty\tau^*\ge\frac{1}{\sum\limits_{i=1}^{M}(M_\infty\tau_i)^{-1}}=\frac{TM}{\sum\limits_{i=1}^{M}c_i^{-1}}\ge\frac{T}{\min\limits_i c_i^{-1}}\ge T, \tag{70}$$

причому бажано, щоб $M_\infty\tau_i\ge MT$.

З (68) слід, що статистики $\varphi_{ni}^{-n}$, $i=\overline{1,M}$,є $P_\infty$ – мартингалами з нульовими середніми, тому $M_\infty\tau_i=M_\infty\varphi_{\tau_i i}$. Так як $M_\infty\varphi_{\tau_i i}\ge A_i(T)$, то вибір $A_i(T)=MT$ забезпечує нерівності $M_\infty\varphi_{\tau_i i}\ge MT$ і відповідно до (70) належність правила (67) класу $\Delta(T)$. Далі зважаючи на (69) $M_{i\lambda}(\tau^*-\lambda\,|\,\tau^*>\lambda)=\bar\tau_{i\lambda}^*\le M_{i\lambda}(\tau_i-\lambda\,|\,\tau_i>\lambda)$.

Оскільки аналогічно (66) при $A_i(T) = MT$ і $T \to \infty$ $M_{i\lambda}(\tau_i - \lambda \mid \tau_i > \lambda) = I_i^{-1} \ln(MT)(1+o(1)) = I_i^{-1} \ln T(1+o(1))$, де $I_i = \int_{-\infty}^{\infty} p_i(x) \ln \gamma_i(x) dx$, то $\bar{\tau}^*_{i\lambda} \le I_i^{-1} \ln T(1+o(1))$, $T \to \infty$. (71)

Однак очевидно, що середня затримка у виявленні збою для багатоальтернативного правила не може бути меншою за середню затримку у разі, коли є лише одна (і-а) гіпотеза. Оскільки мінімальна затримка за наявності однієї і-ї гіпотези не менше величини $I_i^{-1} \ln T(1+o(1))$, іноді $T \to \infty$, то для всіх багатоальтернативних правил класу $\Delta(T)$ справедлива наступна оцінка.

$$\sup_{\lambda \ge 1} M_{i\lambda}(\tau - \lambda \mid \tau > \lambda) \ge I_i^{-1} \ln T(1+o(1)). \quad (72)$$

З (71) та (72) випливає, що

$$\bar{\tau}^*_{i\lambda} = I_i^{-1} \ln T(1+o(1)),\ T \to \infty,\ i = \overline{1,M}. \quad (73)$$

Таким чином, багатоальтернативне правило (67) при $A_i(T) = MT$ асимптотично еквівалентно і для середньої затримки у виявленні збою при і-й гіпотезі справедливо рівності (73), що збігається з аналогічною рівністю за наявності лише однієї гіпотези. Тому воно є асимптотично мінімаксним правилом за першим порядком:

$$\lim_{T \to \infty} \frac{\sup_{\lambda \ge 1} \bar{\tau}^*_{i\lambda}(T)}{\inf_{u \in \Delta(T)} \sup_{\lambda \ge 1} M_{i\lambda}(\tau - \lambda \mid \tau > \lambda)} = 1.$$

**Висновки.** Таким чином, процес оцінювання властивості інформаційних вторгнень чи психологічних впливів, що дозволяють з'ясувати рівень впливу їх у особистість чи групу людей, і навіть розробити ефективну систему протидії є дуже важливим. Враховуючи те, що основна відмінність алгоритмів виявлення інформаційних впливів від класичних алгоритмів перевірки гіпотез полягає в тому, що рішення про відсутність вторгнення може бути прийнято лише на останньому доступному кроці спостереження, оскільки ніколи немає гарантій, що воно відбудеться після закінчення спостережень, розглянуті у статті завдання є окремим випадком більш загального виявлення.

**Список літератури**


1. Зелинский С.А. Информационно-психологические воздействия на массовое сознание. СПб: Скифия, 2008. 403с.
2. Ольшанский Д.В. Психология масс. СПб: Питер, 2002. 403с.
3. Пирцхалава Л.Г., Хорошко В.А., Хохлачова Ю.Е., Шелест М.Е. Информационное противоборство в современных условиях. К: ЦП «Компринт», 2019. 226.с.
4. Вальд А. Последовательный анализ. М: Физматгид, 1999. 330 с.
5. Кокс Д., Льюис П., Статистический анализ последовательных событий. Мир, 1999. 318 с.
6. Ширяев А.Н. Статистический последовательный анализ. Оптимальные правила остановки. М: Наука, 2009. 277 с.
7. Сейдж Э.П., Мелз Дж.Л. Теория оценивания и ее применения в связи и управлений. М: Связь, 2006. 498 с.
8. Бакут П.А., Жулина Ю.В., Иванчук Н.А. Обнаружение движущихся объектов. М: Сов.радио, 2000. 298 с.
9. Опірський І.Р. Оптимізація послідовних процесів прийняття рішень при умовно екстремальній постановці задачі. *Інформаційна безпека*. 2014. Т.16. №.4. С. 120-127.

10. Репин В.Г. Тартановский Г.П. Статистический синтез при априорной неопределенности и адаптация информационных систем. М.; Советское радио, 1977.
11. Pollar P.M. Average Run lengths of an optimal Method of Detecting in Distribution. *Ann.Statist*. 2007. V.15. №2. P. 749-779.
12. Закс Ш. Теория статистических выводов. М: Мир, 1995. 779 с.
13. Pollak M., Siegmund D.A. Diffusion Process and its Applications to detecting a change in the Drift of Brownian Motion. *Biometrika*. 2005. V.72. №2. P. 267-280.
14. Lorden G. Procedures for Reacting to a Change in Distribution. *Ann. Math. Statist*. 2001. V. 42. №6. P. 1897-1908.
15. Moustakides G.N. Optimal Stopping Times for Detecting Changes in Distributions. *Ann. Statist*. 2006. V. 14. №4. P. 1379-1389.

## ASSESSMENT OF PROPERTIES OF INFORMATION INTRUSIONS

V.O.Khoroshko, V.D.Kozyura, Yu.E.Khokhlachova, N.S.Vishnevska

National Aviation University,

1, Lubomyra Huzara ave., Kyiv, 03058, Ukraine,
e-mail: professor_va@ukr.net

The article offers an assessment of the properties of information intrusions, as they are random processes and have their own properties. This makes it possible to find out the level of their influence on an individual or a group of people, and to develop an effective countermeasure system. At the same time, psychological influence or information intrusion, which is a component of informational and psychological confrontation, is an influence on people and their groups, which is carried out with the aim of changing the ideological and psychological structures of their consciousness and subconsciousness, transforming emotional states, and stimulating certain types of behavior. The restructuring of the psyche under the influence of informational and psychological influence can be different, both in breadth and in terms of temporal stability. In order to provide informational intrusion and psychological impact, disruptions and transferences must first be induced. The dynamic balance between them is disturbed, a person begins to experience a state of cognitive dissonance. After that, you can encourage him to restore mental balance by changing his former views, beliefs and attitudes, taboos and stereotypes of behavior. Therefore, it is very important to evaluate the properties of informational intrusions or psychological influences, which allow to find out the level of their influence on an individual or a group of people, and to develop an effective countermeasure system. It is shown that all the methods used for the evaluation and analysis of random processes can be used for their analysis. The task of detecting spiking bursts was also examined, which showed that with the help of such signals, observations can be differentiated.

**Keywords:** cyber security, cyberspace, information warfare, information and psychological warfare, information intrusions, mathematical methods.

DOI 10.15276/imms.v13.no1-2.162
УДК 519.6:004.93

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp.162-172

# КОМБІНОВАНІ АЛГОРИТМИ ВИЗНАЧЕННЯ ПОЧАТКОВОГО РІШЕННЯ ЗАДАЧ ДИСКРЕТНОЇ ОПТИМІЗАЦІЇ

Б. І. Юхименко[1], Н.П. Волкова[2], Ю.Ю. Козіна[3]

Національний університет «Одеська політехніка»,
Одеса, 65044, пр. Шевченко, 1,
e-mails: biruteyu@gmail.com[1], volkova.n.p@op.edu.ua[2], yuliyakc21@gmail.com[3]

Проблему вирішення завдань дискретної оптимізації повністю не вирішено. Безліч публікацій, наукових розробок, алгоритмів та програмних продуктів не дає можливості перевести математичний апарат розв'язання задач дискретної оптимізації до класу P складності обчислень. Усі аналітичні та комбінаторні алгоритми вирішення задач лінійної та нелінійної оптимізації є NP складними. У зв'язку з цим будь-які розробки, щодо підвищення ефективності роботи алгоритмів, залишаються затребуваними та актуальними. У цій роботі запропоновано використовувати детерміновані та ймовірнісні прийоми формування пріоритетної черги компонент вектора рішень з метою присвоєння їм позитивних значень. Послідовне формування варіанта рішень можна використовувати як отримання наближеного рішення чи рекордного значення цільової функції, що у точних алгоритмах як вихідне рішення, підлягає поліпшенню. У роботі наведено способи формування пріоритетної черги конкретизації компонентів вектора рішень. Основу детермінованих методів становить ідея жадібного алгоритму. Місце розташування у черзі визначається величиною відповідної компоненти вектора вартості. Облік величини нев'язки у системі обмежень підвищує пріоритетність компоненти. За таким засобом модифікується другий детермінований спосіб. Імовірнісна оцінка пріоритетів ґрунтується на ідеях алгоритмів мурашиної колонії та імітації відпалу. Розмір ймовірності визначає значимість компоненти – претендента на позитивне значення. Наведено числовий приклад невеликої розмірності задачі про ранець, що демонструє отримання наближеного розв’язання.

**Ключові слова:** дискретна оптимізація, рекорд, пріоритетна черга, метод гілок та меж.

**Вступ.** Дискретна оптимізація використовується у різних галузях діяльності. Завдання планування, транспортування вантажів, логістики та багато інших приводяться до моделей цілочисельного, частково цілочисельного, лінійного та нелінійного математичного програмування, до моделей з булевими змінними. Методи реалізації цих моделей часто не відповідають сучасним вимогам. Вирішення багатьох прикладних завдань, що мають велику розмірність, не призводить до оптимального результату. Вони відстають в оперативності і у деяких випадках не дають очікуваного рішення. З математичних позицій методи дискретної оптимізації належать до класу NP складності. Тому, завдання їх модифікації, залучення комбінаторних прийомів, що збільшують швидкість збіжності, є актуальним. Мотивація у сучасній дискретній оптимізації – як перейти з класу NP складності до класу P.

Останнім часом, для вирішення багатьох комбінаторних завдань, задач цілочисельного лінійного програмування використовуються прийоми, що імітують поведінку не інтелектуальних істот. Популярними стали алгоритми мурашиної

колонії [1]. Вони ефективно використовуються для вирішення задачі про комівояжера, розподілу виробничих замовлень, розміщення продуктивних сил та ін. Характерні моменти поведінки мурах як масовість, інформативність, прямий та зворотний зв'язок, дозволяє розробляти наближені ймовірні алгоритми. Інтерпретація та формалізація поведінки мурах є основою створення алгоритмічних прийомів, що залежать від низки параметрів. Число мурах-агентів виконують однакову роботу – паралельне рішення одного і того ж завдання, рівень інформативності – кількість феромонів, переданих особинам колонії як повідомлень, зміна ситуації – випаровування феромонів – визначає величину ймовірності прийнятого рішення. Для кожного конкретного випадку, конкретної задачі та її структурних особливостей є індивідуальний вибір значень параметрів. Це не є детермінований процес. Неможливо теоретично довести ні кінцівку, ні точність одержуваного рішення. Це недолік таких алгоритмів, які імітують поведінку живих істот чи процесів які є у природі.

Поліпшення ефективності роботи перебірних – комбінаторних алгоритмів може вестись різними шляхами. Оскільки методика розв'язання оптимізаційних завдань передбачає отримання вихідного рішення, питання їх визначення доречні й у дискретній оптимізації. Чим точніше вихідне рішення, тим швидше отримуємо оптимальне рішення під час використання точних алгоритмів. Крім того, вони можуть використовуватись як деякі наближені рішення. При вирішенні практичних завдань, що мають великі розмірності та не відрізняються точністю вихідної інформації, оптимальне рішення не може бути потрібне. Крім того, значення цільової функції вихідного рішення використовується як рекордне під час роботи комбінаторних алгоритмів.

**Мета роботи.** Метою роботи є пошук комбінованих алгоритмів визначення початкового розв’язання задач дискретної оптимізації для підвищення оперативності отримання рішення. У цій роботі пропонується використовувати деякі детерміновані та ймовірнісні прийоми як засоби отримання пріоритетної черги конкретизації значень компонент вектора рішень. Отриманий варіант розв’язання оптимізаційної задачі може використовуватися як наближене рішення. Значення цільової функції розглядається як рекордне значення при відсіюванні неперспективних підмножин варіантів на базі методу гілок та меж чи інших комбінаторних алгоритмів. Пропонуються засоби визначення початкових варіантів рішень без великих обчислювальних складнощів отримання рішення, яке може бути вихідним для точних алгоритмів, а також наближеним, якщо особлива точність не потрібна.

**Основна частина.** Основним комбінаторним методом вирішення завдань дискретної оптимізації є метод гілок та кордонів. Бездоказова його збіжність за рахунок кінцівки безлічі варіантів, гнучкість і відкритість дозволяє вводити блоки, що модифікують та що прискорюють процес отримання оптимального рішення. Наявність процедури оцінювання множини варіантів, розбиття множини на підмножини, а також ознаки оптимальності, дає можливість повний перебір варіантів призвести до часткового перебору, за рахунок відсіювання неперспективних підмножин. Насамперед, це алгоритмізація різних способів отримання оцінок (кордонів) множини (підмножин) варіантів розв'язання задачі. Найчастіше використовується ідея розширення – звуження їхньої області визначеності [2].

Якщо $Z^{'}$ оптимальне значення цільової функції, що отримане на розширеній множині варіантів $G'$, то воно може використовуватися як оцінка вихідної множини $G$. Позначимо її через $\xi(G)$.

При розв'язанні задач цілочисельного лінійного програмування (ЦЛП) розширення безлічі варіантів проводиться шляхом приведення завдання до відповідної задачі лінійного програмування (ЛП), тобто. відкидаються вимога цілісності компонентів вектора рішень.

У роботі [3] наведено два способи розширення безлічі варіантів для вирішення задачі про багатомірний ранець методом гілок та меж. Дано експериментальну оцінку їх ефективності.

Дійсно, якщо визначається максимальне значення цільової функції $Z$, де

$$Z = \max \sum_{j=1}^{n} c_j x_j \tag{1}$$

на множині варіантів $G$, що задається обмеженнями

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i, \quad i = \overline{1,m} \tag{2}$$

$$x_j \in \{0,1\}, \quad j = \overline{1,n} \tag{3}$$

то замінив вимогу (3) на (3`) отримаємо множину варіантів $G'$, що задається обмеженнями (2)-(3'), де

$$x_j \in [0,1], \quad j = \overline{1,n} \tag{3′}$$

Маємо задачу лінійного програмування

$$Z' = \max \sum_{j=1}^{n} c_j x_j$$

при обмеженнях

$$\sum_{j=1}^{n} a_{ij} x_j \le b_i, \quad i = \overline{1,m},$$

$$x_j \in [0,1], \quad j = \overline{1,n}.$$

Завжди $G \subseteq G'$ і отже, $Z' \ge Z$. Оптимальне значення $Z'$ є оцінкою множини цілочисельних варіантів тобто $Z' = \xi(G)$.

Більш спрощений спосіб отримання оцінок полягає у вирішенні одновимірних $m$ не цілочисельних завдань про ранець, а саме

$$Z_i' = \max \sum_{j=1}^{n} c_j x_j$$

при обмеженнях $\sum_{j=1}^{n} a_{ij} x_j \le b_i, \; x_j \in [0,1], \quad j = \overline{1,n}$, складових по черзі підмножини $G'$, $i = \overline{1,m}$. Оцінка безлічі цілих варіантів рішення визначається як:

$$\xi(G) = \min Z_i', \; i = \overline{1,m}.$$

Очевидно, що для всіх $i = \overline{1,m}$ справедливе співвідношення $G \subseteq G'$.

Якщо складність обчислень визначається кількістю виконуваних операцій, другий спосіб значно менш складний. Отримання оптимальних не цілочисельних рішень одномірних завдань про ранець методом Данцига [4] простий, не містить

жодних ітераційних процесів. Цього сказати про перший спосіб не можна. Визначення оцінки $\xi(G)$ вимагає використання симплекс-методу, що відрізняється своєю обчислювальною складністю. Якщо складність обчислень визначається числом ітерацій, виконаних до отримання оптимального рішення, краще використовувати перший спосіб. Значення цільової функції $Z'$ на розширеному множині менше відхилиться від $Z$ на вихідному. Кількість вершин дерева рішень, що переглядаються, значно зменшиться. Процедура розбиття безлічі варіантів на підмножини є досліджуваною і модифікованою частиною методу. Відомі різні підходи поділу безлічі варіантів на підмножини. Від цього залежить пошук перспективного, що містить оптимальний варіант підмножини однієї з перших робіт за методом гілок і кордонів є алгоритм Ленд і Дойг [5]. Пропонується безліч варіантів ділити на два підмножини. Параметром поділу є нецілочисельна компонента $x_k$ в оптимальному варіанті розв'язання відповідної задачі ЛП. Мотивація така – виключити з розгляду варіанти, у яких зустрічатимуться нецілочисельні значення компоненти $x_k$ навколо її оптимального значення. Безліч варіантів ділиться на два підмножини шляхом доповнення до основних обмежень таке $x_k \le [x_k]$ чи таке $x_k \ge [x_k + 1]$ де [a] – ціла частина від a . Такий спосіб розбиття, як і весь алгоритм, не мав успіху. Предком методу гілок та меж вважається алгоритм Літла та ін. [6] для вирішення задачі про комівояжер. Параметром поділу є пара міст, яку доцільно включити до циклу об'їзду. Пари оцінюються величиною «втрат» у значення цільової функції, якщо не будуть включені до циклу. Перевага надається парі з найбільшою кількістю втрат. Існує безліч модифікацій методу для вирішення задачі про комівояжер: точні [7], наближені [8], імовірнісні – наближені [1], з симетричною матрицею відстаней [4], обчислювальні алгоритми для вирішення задач великої розмірності [9] та багато інших. Ідея послідовної побудови рішення [10] внесло деяке впорядкування в процедуру розбиття безлічі варіантів на підмножини. На кожному етапі розбиття конкретизуються можливі значення компоненти вектора рішень. Множина, що розбивається, ділиться на частини, кожна з яких містить варіанти, що відрізняються за значенням змінної, що визначається. Проблема полягає у визначенні пріоритетної черги компонент, згідно з якою вони будуть конкретизовані. Від цього залежить ступінь наближення до оптимального одержуваного рішення [11]. Відомі також алгоритми як жадібний [12], генетичний [13], мурашиної колонії [14], багато комбінованих (наприклад [15]), що пропонують різні способи формування пріоритетної черги. Багато хто з них використовується для отримання наближеного рішення, рекордного значення цільової функції або гарного початкового варіанта рішення, що підлягає подальшому покращенню. У методі гілок та меж рекордне значення використовується як засіб відсіювання варіантів.

Наближені алгоритми комбінаторної оптимізації є двоетапними. На першому етапі якимось засобом визначається варіант рішення, але в другому – його поліпшення. Поліпшення – це зміна значень параметрів, спеціальні прийоми, застосування методик іншого способу, або, з так званої ідеї двоїстості. У задачах з булевими змінними деяким компонентам, що мають значення «1», присвоюється значення «0» і навпаки. Що ж до першого етапу, то допустиме рішення виходить або детермінованим способом, або з використанням елементів випадкового пошуку. Сама процедура отримання варіанта рішення розглядається як присвоєння позитивних значень компонентам вектора рішень згідно з їхньою пріоритетною чергою, яка визначається з урахуванням структурних особливостей завдання, або

пропорційно величині ймовірності, що обчислюється в залежності від ряду параметрів.

У роботі запропонована така схема класифікації наближених алгоритмів комбінаторної оптимізації (рис. 1).

Нижче наводиться короткий опис підходів формування пріоритетної черги компонент вектора рішень для задач у постановці (1)-(2)-(3).

Позначення:

– вектор рішень $X = \{x_j\}, \quad j = \overline{1,n}$;

– множина індексів конкретизованих компонент $I_x = \{j / x_j = 1\}$;

– множина індексів компонент вектора рішень, яким можна присвоїти значення «1»: $V_j = \left\{ \bar{j} \in I_x / (b_i - \sum_{j \in I_x} a_{ij}) \geq 0 \right\}, \ \forall i$;

– пріоритетна черга індексів компонент вектора рішень $S_j = \{j_1, j_2, ..., j_k\}$.

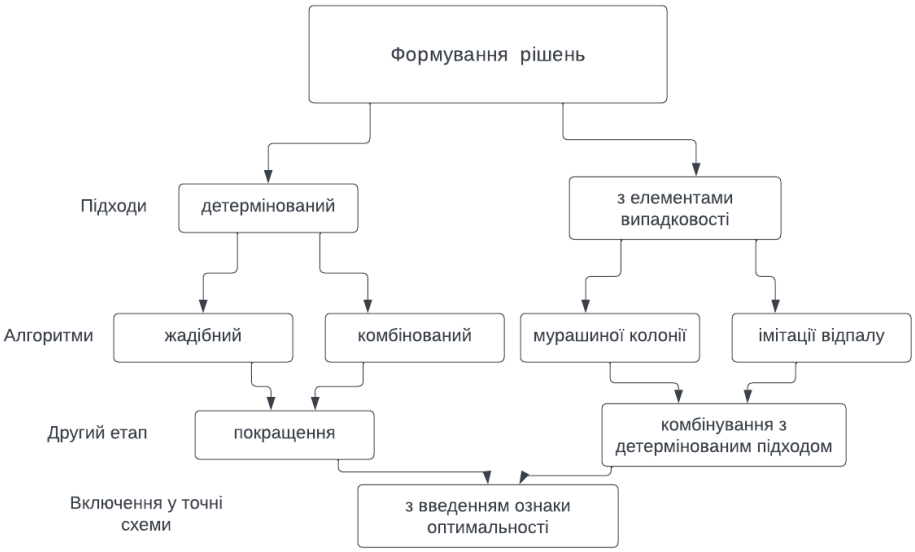



**Рис.1.** Класифікація наближених алгоритмів комбінаторної оптимізації

При детермінованому підході пріоритетна черга формується лише з урахуванням числових даних, які входять у математичну модель. До них належить

$$C = \{c_j\}_{j=\overline{1,n}}; \ A = a_{ij}, \ i = \overline{1,m}; \ j = \overline{1,n};$$

та

$$B = \{b_i\} \, i = \overline{1,m}.$$

Ідея жадібного алгоритму передбачає облік лише компонент вектора цін як величин, які вносять «внесок» значення цільової функції.

Якщо $c_{j_1} \geq c_{j_2} ,..., \geq c_{j_k}$, то пріоритетна черга $S_j = \{j_1, j_2, ..., j_k\}$.

Комбінований спосіб поєднує два моменти: ідею жадібного алгоритму та врахування величини нев'язки у системі обмежень при конкретизації чергової

компоненти $x_j$ вектора рішень. Визначається «вагова» оцінка $q_j$ для $x_j / j \in V_j$ таким чином отримаємо:

$$q_j = c_j \sum_{i=1}^{m} (b_i - a_{ij}) .$$

Ранжування в порядку зростання величин $q_j$ визначає пріоритетну чергу $S_j$ .

При недетермінованому підході пріоритетність компонентів визначається величиною ймовірності. Чим більша ця величина, тим більший шанс вибору відповідної компоненти. Сам вибір, зазвичай, здійснюється шляхом генерації випадкової величини, рівномірно розподіленої в інтервалі (0,1). Інтервал ділиться на частини пропорційно до величин ймовірностей. Попадання значення випадкової величини у відповідну частину інтервалу визначає номер компоненти. Відкритим залишається питання про обчислення величин ймовірностей, що надають пріоритетність компонентам. Один із способів це імітація поведінки живих неінтелектуальних істот чи природних явищ. Формалізація такого недетермінованого природного процесу призводить до набуття числових значень. У роботі [1] описано, як поведінка особин мурашиної колонії подає ідею створення алгоритмів розв'язання задач комбінаторної оптимізації. Багаточисельність жителів колонії, кожен із яких виконує самостійно свою роботу, досягаючи загальної мети, сприймається як елемент масивності. У імовірнісних алгоритмах це представляється як набір статистичних даних про отримувані рішення. Вирішується те саме завдання багатьма агентами (ніби мурахами), число яких представляється параметром у роботі алгоритму.

Інший дуже важливий момент це наявність передачі між особами колонії. При пересуванні відкладається слід через звані феромони. Чим більше їх конкретизація, тим більше мурах піде цим шляхом. Формально – кількість керує процесом та забезпечує прийняття рішень. Природно, кількість феромонів може збільшуватися, але з часом і зменшаться - випаровування. Результат рішення визначається їх кількістю. У мурашиних алгоритмах це є двома параметрами $\alpha$ та $\beta$. Випаровування феромонів залежить від відстані між точками пересування. Чим вона більша, тим значніше зменшується їх кількість. Говорять, зменшується інформативність про стан справ. Цей компонент виявляється у алгоритмах через структурні особливості розв'язуваної задачі. З метою отримання ймовірнісної оцінки компоненти вектора рішень необхідно дати формальну оцінку накопичення інформативності про неї (феромонів) з урахуванням зворотного інформаційного зв'язку (випаровування феромонів) через структурні особливості завдання, що розв'язується.

Позначимо:

$\alpha$ – рівень інформативності;

$\beta$ – рівень зворотнього інформаційного зв’язку;

$a_j$ – інформація про *j*-у компоненту вектора рішень (наявність віртуальних феромонів) $j = \overline{1,n}$;

*r* – кількість паралельно розв'язуваних завдань (мурах колонії).

Тоді величина ймовірності визначається за формулою

$$P_j^l = \frac{a_j^{\alpha} Y(c)^{\beta}}{\sum_{j \in V_j^l} a_j^{\alpha} Y(c)^{\beta}} \quad j \in V_j^l ; l = \overline{1,r} \tag{4}$$

де

$$Y(c)=\{c_j - \min c_j, якщо\, c_j > \min c_j\}.$$

Величина ймовірності визначається для тих компонентів, які з структурних особливостей, можуть набувати значення «1».

Отримаємо $r$ варіантів рішення $X^l=\{x_j^l\}$ $j=\overline{1,n};\ l=\overline{1,r}$, де кожна $x_j$ – а компонента приймає значення «1» з ймовірністю, що визначається за формулою (4).

Початкове значення $a_j$ є однаковим для всіх $j=\overline{1,n}$ та всіх $l=\overline{1,r}$. Надалі при виконанні етапу поліпшення наближеного рішення значення будуть змінюватися з урахуванням накопиченого досвіду отримання безлічі варіантів рішення.

Врахування структурних особливостей задачі, що виражається через величину $Y(c)$, може здійснюватися індивідуально (див. напр. [16]). Використовуючи мотивацію генетичного алгоритму [12] «отримати добре потомство», визначаються вагові оцінки $\gamma_j$ для кожної компоненти $x_j$ вектора рішень, де

$$\gamma_j = \sum_{l=1}^{r} x_j^l / n,\ j \in V_j^l.$$

Покращення наближеного рішення розглядається як ітераційний процес. Якщо $t$ номер ітерації, то при переході від ітерації $t$ до ітерації $t+1$ змінюється значення $a_j$ за формулою (5)

$$a_j(t+1)=(1-\rho)a_j(t)+\gamma_j \qquad (5)$$

де $\rho$ – регулюючий параметр.

Другий детермінований підхід формування вектора рішень, заснований на ідеї імітації відпалу [17]. Це формалізація фізичного процесу, пов'язаного із зміною внутрішньої енергії тіла, яке розігрівається до високої температури і далі починає повільно остигати. Внутрішня його енергія переходить із одного стану до іншого. Потрібно знайти точку з мінімальною внутрішньою енергією. Дослідження того, що відбувається, дало можливість створити алгоритм. Оптимізаційна задача розглядається як фізична система, а допустиме рішення та значення цільової функції як стан тіла та його внутрішня енергія. У такій аналогії процес відпалу представляє процес знаходження рішення з мінімальним значенням цільової функції.

Досить простий у обчислювальному сенсі алгоритм імітації відпалу використовувався для розв'язання задачі про комівояжера, найпростішого завдання розміщення та інших прикладних завдань [18]. Крім того, доведена його асимптотична збіжність [17]. З математичних позицій алгоритм належить до класу алгоритмів локальної оптимізації. Оптимальне рішення (локальний оптимум) шукається на околиці локального оптимуму. При вирішенні оптимізаційних завдань з булевими змінними поняття «околиця» сприймається як сукупність рішень, у яких значення однієї з компонентів замінюється протилежним («1» на «0» чи навпаки).

Алгоритм імітації відпалу безпосередньо не встановлює пріоритетну чергу конкретизації компонентів вектора рішень. В околиці деякого допустимого рішення випадковим чином на основі ймовірнісної оцінки здійснюється пошук локального оптимуму. Величина ймовірності залежить від температур $t$ та напряму, у якому ведеться пошук. Хай визначена початкова температура $t_0$. Використовуя будь-який спосіб формування рішення, отримано варіант $X$ та значення цільової функції $Z(X)$. На околиці $X$ за допомогою деякого оператора $\Psi$ визначено варіант $X'$ та $Z(X')$.

Обчислюється відхилення $\Delta = Z(X') - Z(X)$, якщо $\Delta > 0$, то рахується що стан $X'$ краще, ніж $X$ з ймовірністю $p = e^{-\Delta/t}$.

Перехід у новий стан також має випадковий характер. Генерується випадкова величина $\xi$ рівномірно розподілена в інтервалі (0,1). Якщо $\xi > p$, то стан, що описує $X$ зберігається, в іншому випадку переходимо в стан $X'$ тобто $X = X'$ і $Z(X) = Z(X')$; зменшується температура на деякий $\Delta t (t = t - \Delta t)$ та повторюється ітераційний процес пошуку нового локального оптимуму. Процес керується або досягненням певної температури $t$, або досягненням достатнього значення цільової функції.

Приклад. Розв'язується багатовимірне завдання про ранець з використанням ряду модифікуючих блоків методу гілок і кордонів, кожен з яких певною мірою покращує ефективність його роботи. Числові дані завдання запозичили з [4]. Маємо задачу у постановці:

$$Z = \max(6x_1 + 5x_2 + 2x_3 + 3x_4 + 4x_5 + 5x_6 + 4x_7 + 8x_8 + 7x_9 + 3x_{10})$$

при обмеженнях

$$\begin{cases} 2x_1 + 2x_2 + 2x_3 + x_4 + 2x_5 + 2x_6 + 2x_7 + 3x_8 + 3x_9 + x_{10} \le 10 \\ x_1 + 3x_2 + 3x_3 + 2x_4 + 2x_5 + 2x_6 + x_7 + 4x_8 + 2x_9 + 2x_{10} \le 7 \\ 4x_1 + x_2 + 2x_3 + 2x_4 + 2x_5 + 3x_6 + 3x_7 + 3x_8 + 2x_9 + x_{10} \le 8 \end{cases}$$

$$x_j \in \{0,1\}, \quad j = \overline{1,10}.$$

Оцінка множини варіантів $\xi(G)$ розраховується використовую процедуру його розширення. Було вирішено три не цілочисельних одновимірних задач про ранець методом Данцига [4]. Величини $\lambda_j^1 = \frac{c_j}{a_{ij}}, j = \overline{1,n}$ ранжуються у порядку їх не спадання, на основі чого визначаються послідовності присвоєння значень «1» відповідним компонентам відповідного вектора рішень. Маємо:

**Таблиця 1.**

Результати розрахунків оцінок множин варіантів

| $i$ | Послідовність | Варіант рішення | $Z_i^*$ | $\xi(G)$=24 |
|---|---|---|---|---|
| 1 | 1 4 10 8 2 6 9 5 7 3 | 1 1 0 1 0 ½ 0 1 0 1 | 27,5 | |
| 2 | 1 7 9 6 8 5 2 4 10 3 | 1 0 0 0 0 1 1 ¼ 1 0 | 24,0 | |
| 3 | 2 9 10 8 5 6 1 4 7 3 | 0 1 0 0 ½ 0 0 1 1 1 | 25,0 | |

Початкове рекордне значення цільової функції R визначалося згідно з пріоритетною чергою компонент вектора рішень. Розглядалися два способи: перший спосіб – пріоритетність компоненти визначалася величиною відповідної компоненти вектора вартості; другий – з урахуванням величини нев'язки у системі обмежень.

**Таблиця 2.**

Результати розрахунків пріоритетності компоненти

| Спосіб | Пріоритетна послідовність | Варіант рішення | R |
|---|---|---|---|
| 1 | 8 9 1 2 6 5 7 4 10 3 | 0 0 0 0 0 0 1 1 1 0 | 19 |
| 2 | 9 1 2 6 8 5 7 10 4 3 | 1 1 0 0 0 0 0 0 1 0 | 18 |

Оптимальне значення цільової функції $Z^*$ знаходиться в інтервалі $19 \le Z^* \le 24$.

Процес отримання оптимального рішення та значення $Z^*$ наведено на рисунку 2. Дерево рішень будувалося шляхом послідовної конкретизації змінних вектора рішень згідно з пріоритетною чергою S, отриманою способом 1.Оцінки підмножин визначались алгоритмом Данцига. Підмножини представлені $j$ чи $\bar{j}$ означає яка компонента та яким значенням конкретизується. Продовження гілки дерева відсікається якщо оцінка підмножини не більша за рекорд R або відповідне значення неприпустимо.

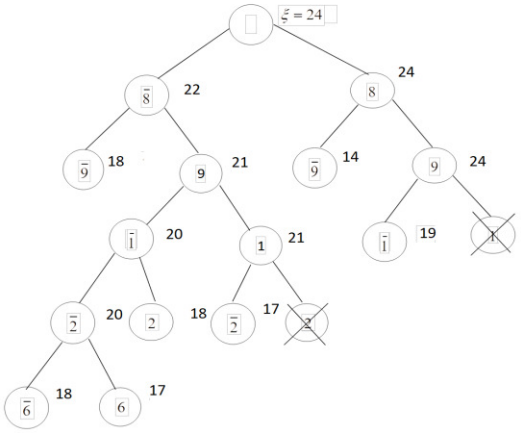



**Рис.2**. Дерево рішень

Рішення цього прикладу алгоритмом мурашиної колонії здійснювалося за таких умов. Початкове значення $\alpha_0 = 0{,}1$. Далі, після виконання кожної ітерації $\alpha$ визначалося згідно з наведеною формулою (5). Параметр $\beta = 0{,}1$ зберігався незмінним. Значення $\gamma_j \left(j = \overline{1,n}\right)$ визначалися на кожній ітерації. Функція $Y(c)$ (облік структурних особливостей задачі) обчислювалося як наведено за формулою 4, де $V_j^i$ – множина індексів компонент вектора рішень претендентів на значення «1». Процедуру рішення здійснювало 6 мурах (агентів). Результат наступний: виконано 20 ітерацій, оптимальне рішення отримано двічі. На 11 ітерації п'ятою мурахою і на 17-ї ітерації третьою мурахою.

**Висновки.** Ефективність роботи наближених алгоритмів оцінюється ступенем наближеності одержуваного рішення до оптимального. Важливу роль також грає оперативність отримання прийнятного рішення. Пропонуються способи визначення початкових варіантів рішень, що дозволяють без великих обчислювальних складнощів отримати рішення, яке може бути вихідним для точних алгоритмів, а також наближеним, якщо особлива точність не потрібна.

Наприклад, у способі гілок та меж важливу роль відіграє наявність початкового рекордного значення цільової функції R. Кількість висячих вершин дерева рішень при роботі безпосередньо залежить від величини R і його близькості до оптимального значення цільової функції. Чим ближче до оптимального значення R, тим більше неперспективних підмножин буде виключено з розгляду. Розглянуті

способи отримання рішення дає добре, інколи і оптимальне рішення. Тому будь-який алгоритм методу гілок та меж може бути модифікований способом отримання початкового варіанта рішення, значення цільової функції якого буде рекордом R. Такий позитивний момент надає суворості ознаки оптимальності методу. Крім того, спрощується формування R-близьких рішень.

Запропоновані способи визначення початкових варіантів рішень дозволили зменшити час на отримання рішення в 1,2 рази у порівнянні з класичними алгоритмами розв'язання задач дискретної оптимізації.

**Список літератури**


1. Штовба С.Д. Муравьиные алгоритмы. *Exponenta Pro. Математика в приложениях*. 2003. № 4. С.70-75. URL: http://surl.li/hbolu
2. Таха Х.А Введение в исследование операций. М.: Вильямс, 2007. 910 с.
3. Ткаленко О.Ю. К вопросу оценки сложности алгоритмов метода ветвей и границ. *Project, Program, Portfolio p3 Management*: Матеріали другої Міжнародної науково-практичної конференції. 2017. Т.1. С.88-92. URL: http://dspace.opu.ua/jspui/ handle/123456789/6992
4. Сигал И.Х., Иванова А.П. Введение в прикладное дискретное программирование. М.: Физматлит, 2007. 304 с.
5. Land A.H., Doig A.G. An automatic method of solving discrete programming problems. *Econometrica*. 1960. V. 28, № 3. P. 497-520. URL: https://doi.org/10.2307/1910129.
6. Литл Дж, Муртик К., Суини Д., Керел К. Алгоритм для решения задачи коммивояжера. *Экономика и математические методы*. 1965. Т.1. вып.1. С. 94-107. URL: http://surl.li/hboyh
7. Lai X., Hao J.K., Glover F., Lii Z. A two-phase tabu – evolutionary algorithm for the 0-1 multidimensional knapsack problem. *Information Sciences* 2018. V.436. P. 282-301. URL: https://doi.org/10.1016/j.ins.2018.01.026
8. Erlebach T., Kellerer H., Pferschy U. Approximating multiobjective knapsack problems. *Management Science*. 2002. 48 (12). P.1603-1612. https://doi.org/10.1287/mnsc.48.12.1603.445
9. Хачатуров В. Р., Веселовский В.Е., Злотов А.В. Комбинаторные методы и алгоритмы решения задач дискретной оптимизации большой размерности. М.: Наука , 2000. 360 с.
10. Михалевич В.С. Последовательные алгоритмы оптимизации и их применение. *Кибернетика*. 1965. №1. С. 45—55; №2. С. 85—88.
11. Юхименко Б.И., Козина Ю.Ю. Сравнительная характеристика алгоритмов метода ветвей и границ для решения задач целочисленного линейного программирования. Труды *Одесского политехнического университета*. 2005. Вып.2. С.199-204. URL: http://surl.li/hbphv
12. Дюбин Г.И., Корбут А.А. Жадные алгоритмы для задачи о ранце: поведение в среднем. *Сибирский журнал индустриальной математики*. 1999. №2 (24). С.68-98. URL: http://surl.li/hbpjl
13. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы. – М.: Физматлит. 2006. 250с.
14. Штовба С.Д. Муравьиные алгоритмы: теория и приложения. *Программирование*. 2005. №4. С.3-18. URL: http://surl.li/hbpkj
15. Юхименко Б.И., Волкова Н.П. Приближенные алгоритмы решения задач о многомерном ранце. *Дослідження в математиці і механіці*. 2017. Т.22 .

вып.2(30). С.104-115. URL: http://rmm-journal.onu.edu.ua/article/view/135745/pdf_39
16. Юхименко Б.И., Ткаленко О.Ю. Алгоритм муравьиной колонии для многомерной задачи о ранце. *Реєстрація, зберігання і обробка даних*. 2019. Т.21, №2. С.3-11. URL: http://drsp.ipri.kiev.ua/article/view/180014/184142
17. Lundy M., Mees A. Convergence of an annealing algorithm. *Math. Programing.* 1996. V.34. pp.111-124. URL: https://link.springer.com/article/10.1007/BF01582166
18. Леванова Т.В. Алгоритм муравьиной колонии и имитации отжига для задач о p-медиане. *Автоматика и телемеханика*. 2004. №3. С.80-88. URL: http://surl.li/hbpne

## COMBINED ALGORITHMS FOR DETERMINING THE INITIAL SOLUTION OF DISCRETE OPTIMIZATION PROBLEMS

B. I. Yukhimenko[1], N.H. Volkova[2], Yu.Yu. Kozina[3]

National Odesa Polytechnic University,
Shevchenko Ave., 1, Odesa, 65044, Ukraine
e-mails: biruteyu@gmail.com[1], volkova.n.p@op.edu.ua[2], yuliyakc21@gmail.com[3]

The problem of solving the task of discrete optimization is not completely solved. Lack of publications, scientific developments, algorithms and software products do not give the ability to transfer the mathematical apparatus of solving discrete optimization problems to the class P of computational complexity. All analytical and combinatorial algorithms for solving problems of linear and non-linear optimization are NP completeness. There are developments to improve the efficiency of robotic algorithms, they are filled with requests and actual ones. In this paper, it is proposed to use deterministic and probabilistic methods for forming a priority queue of decision vector components in order to assign positive values to them. After the formation of the variant of the solution, it is possible to vectorize as if the approached solution is taken away from the record value of the goal function, which in exact algorithms is like a solution, which results in a polyp. In this paper has a method for forming a priority line for concretizing the components of the solution vector. The basis of deterministic methods is the idea of a greedy algorithm. The location in the queue is determined by the value of the corresponding component of the cost vector. The appearance of the value of non-visibility in the system of restrictions increases the priority of the component. Behind such a way, another method of determination is modified. Probability assessment probability of priorities is based on the ideas of algorithms in an ant colony and simulated annealing. The scope of probability indicates the significance of the component – a contender for a positive value. A numerical example of a small variability of the task about a knapsack has been introduced, which demonstrates the imitation of a nearby solution.
**Keywords**: Discrete optimization, record, priority queue, branch and bound method.

DOI 10.15276/imms.v13.no1-2.173
УДК 004.056.5

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp.173-179

# ПІДВИЩЕННЯ СТІЙКОСТІ СТЕГАНОГРАФІЧНОГО МЕТОДУ З КОДОВИМ УПРАВЛІННЯМ ВБУДОВУВАННЯМ ДОДАТКОВОЇ ІНФОРМАЦІЇ ПРИ РОБОТІ З ЦИФРОВИМ ВІДЕО

О.О.Яворський, А.В.Соколов

Національний університет «Одеська політехніка»
Україна, Одеса, 65044, пр-т Шевченка, 1.
email: radiosquid@gmail.com

Розвиток інформаційних технологій, що призвів до зростання долі мультимедійного контенту у світовому трафіку, збільшує роль стеганографічної компоненти у сучасних системах захисту інформації. При цьому, задля забезпечення стеганографічного захисту інформації в режимі реального часу при роботі з цифровим відео, стеганографічний метод, окрім відповідності основним критеріям ефективності, має забезпечувати низький рівень обчислювальної складності, що можливо при виконанні стеганоперетворення у просторовій області контейнера. Одним з таких сучасних методів є стеганографічний метод з кодовим управлінням вбудовуванням додаткової інформації. При цьому окремий інтерес становить задача підвищення стійкості зазначеного стеганографічного методу в умовах атак стисненням алгоритмами стиску цифрового відео, насамперед, розповсюдженого алгоритму MPEG-4. Метою даної роботи є підвищення стійкості стеганографічного методу з кодовим управлінням вбудовуванням додаткової інформації до атак стисненням при роботі з цифровим відео. У роботі проведені експериментальні дослідження стійкості стеганографічного методу з кодовим управлінням вбудовуванням додаткової інформації в умовах атаки стисненням проти вбудованого повідомлення алгоритмом стиску цифрового відео MPEG-4 при вбудовуванні додаткової інформації із застосуванням різних колірних моделей. Показано, що вбудовування додаткової інформації у Y-компоненту колірної моделі YCbCr дозволяє зменшити кількість помилок при вилученні додаткової інформації на 17%. Запропоновано вбудовування додаткової інформації із застосуванням стеганографічного методу з кодовим управлінням лише у динамічні блоки, що дозволило зменшити кількість помилок при вилученні додаткової інформації на 8%. Отримані результати можуть бути використані у практичних стеганографічних застосунках з метою підвищення стійкості стеганографічної компоненти систем захисту інформації.

**Ключові слова:** стеганографія, кодове управління вбудовуванням інформації, цифрове відео, перетворення Уолша-Адамара.

**Вступ і постановка задачі.** Стрімкий розвиток сучасних інформаційних систем, що йде шляхом повсюдного застосування пристроїв, що генерують, оброблюють та передають цифрове відео (ЦВ), призводить до значного зростання ролі стеганографії у застосовуваних системах захисту інформації. При цьому до сучасних стеганографічних методів висуваються значні вимоги, що визначають їх ефективність [1, 2]. Окрім стійкості до атак проти вбудованого повідомлення, значної пропускної спроможності, забезпечення надійності сприйняття, робота з ЦВ передбачає необхідність дотримання низької обчислювальної складності стеганографічних методів, що застосовуються.

При цьому, найчастіше, забезпечення стійкості стеганографічного методу до атак проти вбудованого повідомлення потребує роботи стеганографічного методу в одній з областей перетворення (найбільш застосовуваним є сингулярне розкладання матриць блоків контейнера), що характеризуються значними обчислювальними затратами для своєї роботи, і, відповідно, мало підходять для

обробки ЦВ, особливо, якщо вона здійснюється в режимі реального часу. Саме через це, відомі у літературі стеганографічні методи, що характеризуються стійкістю до атак проти вбудованого повідомлення не заявлені як такі, що можуть працювати з ЦВ [3…8]. Тоді як стеганографічні методи, що працюють з ЦВ, найчастіше засновані на методі LSB і не здатні забезпечити стійкість до атак проти вбудованого повідомлення [9…14].

Перспективним при роботі з ЦВ вбачається застосування стеганографічного методу з кодовим управлінням вбудовуванням додаткової інформації (ДІ) [15], який при роботі у просторовій області контейнера (а, отже, і низькій обчислювальній складності) здатний забезпечити вбудовування ДІ у ту чи іншу частотну складову контейнера. При цьому, при використанні низькочастотних або середньочастотних складових для вбудовування ДІ, метод [15] показує стійкість при роботі з цифровими зображенням до атак стисненням алгоритмом JPEG, що навіть перевищує стійкість відомих стеганографічних методів, які засновані на застосуванні простору перетворень.

Однак, незважаючи на значні результати отримані шляхом розробки стеганографічного методу з кодовим управлінням вбудовуванням ДІ, через особливості застосування ЦВ в якості контейнеру лишаються недослідженими актуальні питання підвищення стійкості стеганографічного методу до атак алгоритмами стиснення відео.

*Метою* даної роботи є підвищення стійкості стеганографічного методу з кодовим управлінням вбудовуванням ДІ до атак стисненням при роботі з ЦВ.

**Вибір колірної моделі для вбудовування ДІ у ЦВ.** Експериментальні дослідження стеганографічного методу з кодовим управлінням вбудовуванням ДІ при роботі з ЦВ в умовах атаки стисненням алгоритмом MPEG-4 проводилися на вибірці з 150 випадкових ЦВ розподільної здатності 1280x720, та тривалості 15 секунд кожне, що застосовувалися в якості контейнеру. У кожне з зазначених ЦВ вбудовувалася ДІ у відповідності до стеганографічного методу [15], після чого ЦВ піддавалося стисненню алгоритмом MPEG-4 з різними рівнями коефіцієнта якості *QF* із подальшим вилученням ДІ і оцінкою кількості помилок, що відбулися.

Зазначимо, що з практичної точки зору важливим є вибір колірної моделі у якій представлені кадри відео і, відповідно, колірної компоненти у які вбудовується ДІ. Можливим є застосування моделі RGB або YCbCr, при цьому вбудовування відбуватиметься у одну із зазначених колірних компонент.

В стандарті MPEG-4 квантування проводиться у моделі YCbCr. Компоненти Y, Cb та Cr представляють яскравість та колірну інформацію зображення, відповідно. Переведення в модель YCbCr дозволяє забезпечити менший рівень квантування елементів Y-компоненти, а також більший рівень квантування для елементів компонент Cb, Cr, що відповідає особливостям сприйняття візуальної інформації людиною.

Зазвичай, відношення квантування між компонентами може варіюватися в залежності від конкретних умов і налаштувань алгоритму MPEG-4.

На рис. 1 показано графік залежності кількості помилок при вилученні ДІ в умовах атаки стисненням алгоритмом MPEG-4 при вбудовуванні ДІ в компоненту R моделі RGB, а також компоненту Y моделі YCbCr. При цьому в обох випадках використовувалося бінарне кодове слово $T^{+}_{b,16,(5,1)}$, порядку $\mu = 16$, що впливає на трансформанту перетворення Уолша-Адамара (5,1).

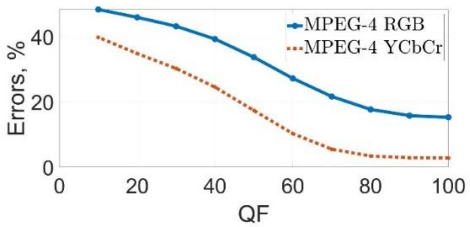



**Рис. 1.** Графік залежності кількості помилок від коефіцієнту стиснення *QF* при використанні колірних моделей RBG та YCbCr

Базуючись на розглянутих особливостях роботи алгоритму MPEG-4, який використовується для проведення атаки стисненням, а також на результатах проведеного обчислювального експерименту, можемо зробити висновок, що застосування простору YCbCr при вбудовуванні ДІ у ЦВ забезпечує значне підвищення стійкості стеганографічного методу до атак стисненням. Так, при коефіцієнті якості *QF*=60, вбудовування ДІ у компоненту Y дозволяє зменшити кількість помилок на 19.78%.

Відзначимо при цьому, що особливості взаємозв'язку збурень, що обумовлені вбудовуванням стеганоповідомлення та/або квантуванням трансформант ДКП під час стиснення блоків контейнера, при їх представленні у колірних моделях RBG і YCbCr потребують подальших теоретичних досліджень.

**Вбудовування ДІ у динамічні блоки ЦВ.** Алгоритм MPEG-4 під час своєї роботи, як і інші алгоритми, призначені для стиснення ЦВ, враховує зв'язки між поточним і попереднім кадром, що дозволяє значно збільшити ефективність стиснення через існування статичних блоків, що не змінюються, або мало змінюються від кадру до кадру. Такі блоки зберігаються лише у опорних кадрах, тоді як інші кадри посилаються на ці блоки, зберігачі у собі лише різницю між блоком у даному та опорному кадрі, яка, до речі, зазвичай піддається більшому рівню стиснення. Таким чином, ДІ, що була вбудована у статичні блоки, з більшою ймовірністю може бути втраченою при атаці стисненням алгоритмом MPEG-4, аніж ДІ, що вбудована у динамічні блоки. Як показали проведені експерименти цю особливість роботи алгоритмів стиснення ЦВ можна застосувати для підвищення стійкості стеганографічного методу з кодовим управлінням вбудовуванням ДІ до атак алгоритмами стиснення ЦВ.

Для цього при виконанні стеганоперетворення, вбудовування ДІ слід виконувати тільки у динамічні блоки, тоді як статичні блоки мають ігноруватися.

Задля введення визначення динамічного блоку введемо показник динамічності. Нехай задані поточний блок $X_{(l,m),k}$ з номером $(l,m)$ деякого кадру з номером $k>1$, а також відповідний йому блок попереднього кадру $X_{(l,m),k-1}$, обидва розміру $\mu\times\mu$.

Тоді показник динамічності для даного блоку визначається як

$$\delta_{(l,m),k}=\sum_{i=1}^{\mu}\sum_{j=1}^{\mu}\left|X_{(l,m),k}(i,j)-X_{(l,m),k-1}(i,j)\right|,\ k>1, \qquad (1)$$

при цьому для блоків кадру $k=1$ прийнято $\delta_{(l,m),1}\to\infty$.

**Визначення.** Динамічним назвемо блок, показник динамічності якого дорівнює або перевищує задане граничне значення $\varepsilon$.

Згідно до (1) очевидно, що максимальне значення показника динамічності складає $255\mu^2$, тоді як його мінімальне значення відповідає 0, що можливо тоді, коли жодний елемент блока не змінився відносно попереднього кадру.

Розглянемо приклад. Нехай задано конкретні блоки $X_{(l,m),k}$ і $X_{(l,m),k-1}$ розміру $8\times8$

$$X_{(l,m),k}=\begin{bmatrix}122&122&120&118&117&115&114&114\\122&122&120&118&117&115&114&114\\122&122&120&119&117&117&115&114\\123&123&120&119&118&117&115&115\\123&123&120&119&118&118&116&115\\122&122&120&119&118&117&117&117\\123&122&122&120&120&118&118&118\\123&123&122&121&120&119&118&118\end{bmatrix},\quad X_{(l,m),k-1}=\begin{bmatrix}123&123&122&119&118&116&114&114\\123&123&122&119&118&116&114&114\\123&123&121&119&118&117&115&115\\123&123&121&120&119&118&116&115\\123&123&122&120&119&118&117&117\\123&123&121&120&119&118&117&117\\124&124&123&122&121&119&118&118\\124&124&123&122&121&120&119&119\end{bmatrix}, \tag{2}$$

при цьому різниця між елементами блоків поточного і попереднього кадру для реальних відеорядів зазвичай є незначною, наприклад, для випадку блоків кадрів (2) отримуємо

$$X_{(l,m),k}-X_{(l,m),k-1}=\begin{bmatrix}-1&-1&-2&-1&-1&-1&0&0\\-1&-1&-2&-1&-1&-1&0&0\\-1&-1&-1&0&-1&0&0&-1\\0&0&-1&-1&-1&-1&-1&0\\0&0&-2&-1&-1&0&-1&-2\\-1&-1&-1&-1&-1&-1&0&0\\-1&-2&-1&-2&-1&-1&0&0\\-1&-1&-1&-1&-1&-1&-1&-1\end{bmatrix} \tag{3}$$

тобто для нашого випадку $\delta_{(l,m),k}=53$.

Розкид показників динамічності для блоків ЦВ дуже сильно залежить від особливостей конкретно обраного ЦВ, тим не менш, у контексті підвищення стійкості стеганографічного методу з кодовим управлінням вбудовуванням ДІ інтерес становить задача дослідження розподілу показників динамічності для заданої вибірки відео.

На рис. 2 показано гістограму розподілу значень показника динамічності блоків для вибірки з 150 випадкових ЦВ розподільної здатності 1280x720 тривалістю 15 секунд, при цьому загальна кількість досліджених блоків розміру $16\times16$ сягнула значення $9.7\cdot10^7$. Для стислості представлення інформації на рис. 2 показані значення кількості блоків, що опинилися у перших 100 інтервалах, кожний з яких має довжину у 10 одиниць.

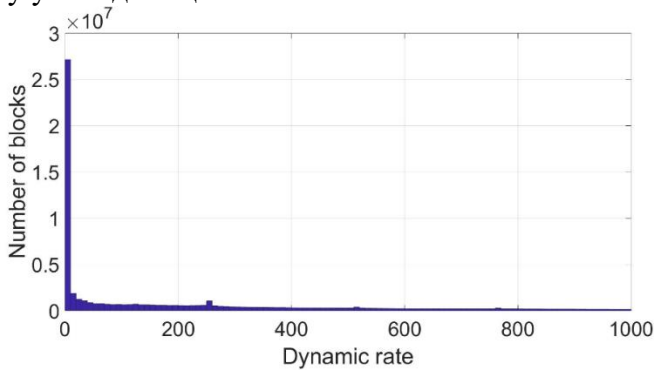



**Рис. 2.** Гістограма розподілу значень показника динамічності блоків

Аналіз даних рис. 2 дозволяє дійти висновку, що через природу відеоряду найбільш ймовірною є поява блоків, що характеризуються граничним значенням показника динамічності у межах $0\le\varepsilon<10$. При цьому, ймовірність появи блоків із граничним значенням показника динамічності $\varepsilon>50$ складає $\sim0.67$, $\varepsilon>500$ складає $\sim0.45$. Наведені значення будемо використовувати для проведення наступних досліджень.

Задля підтвердження ефективності вбудовування інформації лише у динамічні блоки, для яких перевищено заданий поріг показника динамічності $\varepsilon$, було проведено експерименти по вилученню ДІ на виборці з 150 випадкових ЦВ розподільної здатності 1280x720 довжиною 15 секунд кожне із застосуванням бінарного кодового слова $T^{+}_{b,16,(2,1)}$ порядку $\mu=16$, що впливає на трансформанту ДКП (2,1) в умовах атаки стисненням алгоритмом MPEG-4 з різними коефіцієнтами

якості QF. При цьому експерименти проводилися для двох граничних значень показника динамічності блоків $\varepsilon = 50$ і $\varepsilon = 500$, тоді як стеганоперетворення відбувалося із застосуванням колірної моделі YCbCr. Результати проведених досліджень представлені на рис. 3.

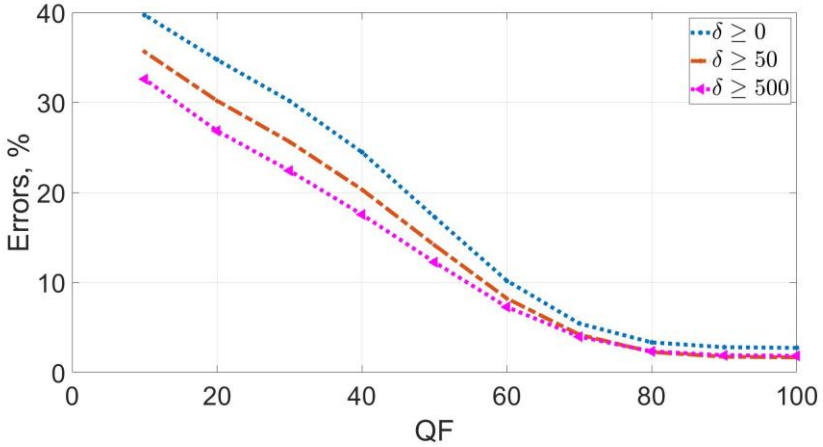



**Рис. 3.** Графік залежності кількості помилок від коефіцієнту стиснення при вбудовуванні ДІ у динамічні блоки

Аналіз даних, представлених на рис. 3, показує зменшення кількості помилок при вилученні ДІ з стеганоповідомлення під впливом атаки стисненням алгоритмом MPEG-4 при вбудовуванні ДІ у блоки, що характеризуються більшими значенням показника динамічності $\delta_{(l,m),k}$. Так при вбудовуванні ДІ лише у блоки, що характеризуються показником динамічності $\delta_{(l,m),k} \geq 50$, кількість помилок при вилученні ДІ зменшилася на значення до 4.55%, тоді як при вбудовуванні ДІ лише у блоки, що характеризуються показником динамічності $\delta_{(l,m),k} \geq 500$, кількість помилок при вилученні ДІ зменшилася на значення до 8%.

Таким чином, практично підтверджено, що застосування для вбудовування ДІ лише динамічних блоків, дозволяє підвищити стійкість стеганографічного методу з кодовим управлінням вбудовуванням ДІ при роботі з ЦВ.

**Висновки.** Відзначимо основні результати проведених досліджень:

1. Проведено дослідження стійкості стеганографічного методу з кодовим управлінням вбудовуванням ДІ при роботі з ЦВ в умовах атаки проти вбудованого повідомлення стисненням алгоритмом MPEG-4.

2. Показано, що задля підвищення стійкості до атак проти вбудованого повідомлення стисненням алгоритмом MPEG-4, вбудовування ДІ слід проводити у Y-компоненту колірної моделі YCbCr. При цьому, у порівнянні з застосуванням колірної моделі RGB, зменшення кількісті помилок сягає величину до 17%.

3. Підвищено стійкість до атак стисненням стеганографічного методу з кодовим управлінням вбудовуванням ДІ за рахунок її вбудовування лише у динамічні блоки цифрового контейнера. При цьому, кількість помилок зменшується на значення до 8% у порівнянні із класичним застосуванням стеганографічного методу з кодовим управлінням вбудовуванням ДІ.

**Список літератури**

1. Кобозева А.А., Хорошко В.А. Анализ информационной безопасности. К.: ГУИКТ, 2009. 251 с.
2. Грибунин В.Г., Оков И.Н., Туринцев И.В. Цифровая стеганография. М.: Солон-Пресс, 2002. 272 с.

3. Li Z., Zhang M., Liu J. Robust image steganography framework based on generative adversarial network. *Journal of Electronic Imaging*. 2021. Vol. 30, Issue 2. P. 023006.
4. Wang S., Zheng N., Xu M. A Compression Resistant Steganography Based on Differential Manchester Code. *Symmetry*. 2021. Vol. 13, No. 2. P. 345.
5. Zhu Z., Zheng N., Qiao T., Xu M. Robust Steganography by Modifying Sign of DCT Coefficients. *IEEE Access*, 2019. Vol. 7. P. 168613-168628.
6. Chanu Y. J., Singh Kh. M., Tuithung T. A Robust Steganographic Method based on Singular Value Decomposition. *Int. Journal of Information & Computation Technology*, 2014. Vol. 4, No. 7. P. 717-726.
7. Мельник М.А. Стеганоалгоритм, устойчивый к сжатию. *Інформаційна безпека*. 2012. №2(8). С. 99-106.
8. Chang C.C., Lin C.C., Hu Y.S. An SVD oriented watermark embedding scheme with high qualities for the restored images. *Int. Journal of innovative computing, information & control*. 2007. Vol. 3, No. 3. P. 609-620.
9. Yadav P., Mishra N., Sharma S. A secure video steganography with encryption based on LSB technique. *2013 IEEE international conference on computational intelligence and computing research.* New Delhi: IEEE. 2013. P. 1-5.
10. Younus Z. S., Younus G. T. Video steganography using knight tour algorithm and LSB method for encrypted data. *Journal of Intelligent Systems.* 2019. Vol. 29. No. 1. P. 1216-1225.
11. Gupta H., Chaturvedi S. Video steganography through LSB based hybrid approach. International Journal of Computer Science and Network Security (IJCSNS). 2014. Vol. 14. No. 3. P. 99.
12. Kunhoth J. Video steganography: recent advances and challenges. *Multimedia Tools and Applications.* 2023. P.1-43.
13. Abed S. An automated security approach of video steganography–based lsb using fpga implementation. *Journal of circuits, systems and computers*. 2019. Vol. 28. No. 05. P. 1950083.
14. Hacimurtazaoglu M., Tutuncu K. LSB-based pre-embedding video steganography with rotating & shifting poly-pattern block matrix. *PeerJ Computer Science.* 2022. Vol. 8. P. e843.
15. Kobozeva A.A., Sokolov A.V. Robust Steganographic Method with Code-Controlled Information Embedding. *Problemele energeticii regionale.* 2021. No. 4 (52). P. 115-130.

# INCREASING THE ROBUSTNESS OF THE STEGANOGRAPHIC METHOD WITH CODE CONTROL WHEN OPERATING WITH DIGITAL VIDEO

O.O.Yavorskyi, A.V.Sokolov

National Odesa Polytechnic University
1, Shevchenko Avenue, Odesa, 65044, Ukraine,
email: radiosquid@gmail.com

The development of information technologies, which has led to an increase in the amount of multimedia content in World traffic, makes the role of the steganographic component in modern information protection systems significant. At the same time, to ensure steganographic protection of information in real-time when operating with digital video, the steganographic method is expected to meet the main effectivity criteria, as well as provide a low level of computational complexity, which is possible when performing steganographic transformation in the spatial domain of the container. One such modern method is the steganographic method with code control of additional information embedding. The problem of increasing the robustness of the specified steganographic method in the conditions of compression attacks by compression algorithms of digital video, primarily, the widespread MPEG-4 algorithm, is of particular interest. The purpose of this paper is to increase the robustness against attacks by compression of the steganographic method with code control of additional information embedding when operating with digital video. In this paper, experimental research on the robustness of the steganographic method with code control of additional information embedding in the conditions of a compression attack against the embedded message by digital video compression algorithm MPEG-4 when embedding information using different colour models is performed. It is shown that embedding additional information in the Y-component of the YCbCr colour model allows reducing the number of errors when extracting additional information by 17%. It is also proposed to embed additional information using the steganographic method with code control only in dynamic blocks, which made it possible to reduce the number of errors when extracting additional information by 8%. The obtained results can be applied in practical steganographic applications to increase the robustness of the applied steganographic component of information protection systems.

**Keywords:** steganography, code control of information embedding, digital video, Walsh-Hadamard transform.

DOI 10.15276/imms.v13.no1-2.180
УДК 004.032

Informatics and Mathematical Methods in Simulation
Vol.13 (2023), No. 1-2, pp.180-189

# РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ОНЛАЙН ПРОКТОРИНГУ

А.А. Брескіна

Національний університет «Одеська Політехніка», просп. Шевченка , 1, Одеса, 65044, Україна; e-mail: anastasia.breskina@gmail.com

Стрімкий розвиток технологій машинного навчання, збільшення доступності пристроїв і широкий доступ до Інтернету значно сприяли зростанню дистанційного навчання. Поряд із системами дистанційного навчання з'явилися системи прокторингу, які мають на меті оцінювати роботу студентів, імітуючи роботу викладача. Однак, незважаючи на розвиток технологій обробки зображень і машинного навчання, сучасні системи прокторингу все ще мають обмежену функціональність: в деяких системах настільки погано були реалізовані методи та алгоритм комп'ютерного зору (хибні спрацьовування при роботі зі студентами різних національностей) та класифікації дій студентів (дуже жорсткі вимоги щодо поведінки студентів), що деякі програмні продукти навіть відмовились від застосування модулів, що використовують елементи штучного інтелекту. Також є проблемою, що сучасні системи переважно зосереджені на відстеженні виключно обличь та погляду студентів і не відстежують їхні пози, дії та емоційний стан. Однак саме оцінка дій та емоційного стану має вирішальне значення не лише для самого навчального процесу, але й для благополуччя студентів, оскільки під час дистанційного навчання вони проводять тривалий час за комп'ютерами або іншими пристроями, що дуже сильно виплаває як на їх фізичного здоров'я, так і на рівень стресу. Наразі контроль цих показників лежить виключно на викладачах або навіть на самих студентах, яким доводиться самостійно опрацьовувати матеріали тестів та самостійних робіт. Додатковою проблемою є якість обробки та зберігання персональних даних студентів, бо більшість систем потребують ідентифікації учня з використанням документів, які підтверджують особистість та зберігають відео роботи учнів па своїх серверах. На основі аналізу усіх цих проблем, що перешкоджають навчальному процесу та потенційно ставлять під загрозу здоров'я учнів у довгостроковій перспективі, у цій статті були представлені додаткові функціональні вимоги до сучасних систем автоматизованого онлайн прокторингу, зокрема, необхідність аналізу дій людини для оцінки фізичної активності та моніторингу гігієнічних практик під час використання комп'ютерів у навчальному процесі, а також вимоги щодо максимального захисту особистих даних учнів.

**Ключові слова:** дистанційне навчання, системи автоматизованого онлайн прокторингу, захист особистих даних, аналіз емоцій людей, аналіз дій людей.

**Вступ.** Технології дистанційного навчання – це ненова прикладна область, що почала свій розвиток з розвитком і поширеністю комп'ютерних технологій та Інтернету. Проблема оцінки поведінки студентів у процесі складання іспитів та виконання індивідуальних робіт існувала завжди. Для їх вирішення були запропоновані системи прокторингу. Системи онлайн прокторингу – це інформаційні системи, призначені для нагляду за процесом виконання тестових чи екзаменаційних завдань та моніторингу і оцінки доброчесності студентів. Ці системи імітують роль викладача, спостерігаючи та оцінюючи поведінку студентів. Спочатку це були системи синхронного прокторингу, де за учнями спостерігали живі люди (самі вчителі, або наймані працівники) [1]. Для автоматизацій процесу та зниження коштовних витрат, були представлені системи асинхронного прокторингу [2], де увесь процес здачі тестових завдань записувався та аналізувався вчителем після самого іспиту постфактум. Але розвинення методів і

моделей штучного інтелекту дали надію на автоматизацію цього процесу оцінки доброчесності студентів. Саме цім інформаційним системам, системам автоматизованого онлайн прокторінгу [2], і присвячена дана робота.

**Аналіз існуючих систем автоматизованого онлайн прокторингу.** Існує велика кількість систем автоматизованого онлайн прокторингу, що основані на штучному інтелекті. За особливістю реалізацій цих систем та гнучкістю їх використання учнями у різноманітних, вже існуючих системах дистанційно навчання, було запропоновано класифікувати їх на такі групи (Рис. 1):

- плагіни, що можуть с легкістю інтегруватися вже у існуючи системи, не вимагаючи від учня часу на встановлення додаткового програмного продукту. Ці програми мають доступ до обмеженого функціоналу в рамках основної платформи: у більшості працюють в браузерах та не мають доступу до всього робочого столу студента. Мають також доступ до мікрофону та відео;
- програмні продукти, що виконують виключно функції системи прокторингу. Працюють як «обгортки» на інші програмні додатки (браузери, тощо) та контролюють додатки та сайти, що учень може використовувати у процесі самостійної роботи над завданнями. Такі системи мають доступ до робочого столу, клавіатури, мікрофону та відео. Ці програми необхідно додатково встановлювати на комп'ютери учнів;
- програмні продукти, які не інтегруються у інші системи. Вчителю необхідно оформляти тестові завдання безпосередньо у цих програмах. Це викликає проблему синхронізації результатів роботи учнів з. іншими системами дистанційного навчання, які можуть використовуватися навчальним закладом, та потребує додаткових дій як від вчителя, так і від учня.

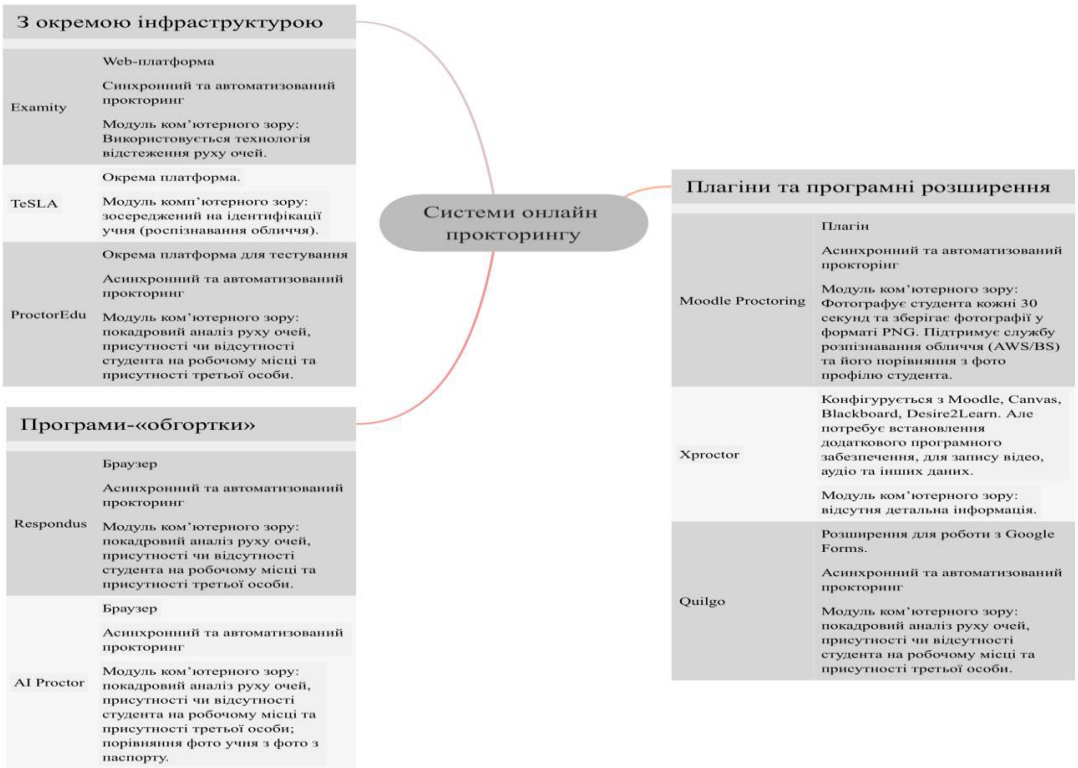



**Рис.1.** Приклад класифікації існуючих програмних продуктів з підтримкою автоматизованого онлайн прокторингу.

Три системи прокторингу на основі штучного інтелекту (ШІ), а саме Quilgo, AI Proctor та ProctorEdu, пройшли тестування для оцінки їхньої продуктивності. Системи продемонстрували хорошу роботу в аналізі активності на робочому столі

та в браузері у всіх тестових випадках. Вони також досягли задовільних результатів в аналізі аудіошуму, правильно ідентифікувавши приблизно 80 відсотків нестандартних ситуацій, хоча іноді давали кілька помилкових спрацьовувань. Однак модуль комп'ютерного зору викликав значні труднощі. Фактично, всі програмні продукти спрацювали вірно лише у 25-30% тестових випадків, а в решті випадків переважали хибнонегативні результати. Це означає, що в той час як системи продемонстрували майстерність в аналізі дій на робочому столі та використанні браузера, модуль комп'ютерного зору виявився проблематичним. За аналізами інших розробників та самих студентів інші системи страждали від неадекватної обробки зображень і алгоритмів штучного інтелекту, стикаючись з труднощами при роботі з учнями різного расового походження. Крім того, ці системи генерували численні хибні спрацьовування через надмірно суворі вимоги до поведінки студентів (вимоги були постійно дивитися у весь час у монітор, наприклад). Як наслідок, наприклад, розробники ProctorU вирішили прибрати ці проблемні ШІ-модулі [3]. У тих самих відгуках та аналізі існуючих систем прокторінгу [4] також було багато питань щодо захисту особистих даних студентів, бо для ідентифікації в більшості платформ необхідно демонструвати свої особисті документи, а повні записи процесу роботи учнів зберігалися без постобробки і додаткового знеособлення даних. Такий підхід для роботи з особистими даними є, наприклад, порушенням загального регламенту про захист даних (GDPR).

**Мета роботи.** Мета роботи – вдосконалення вимог до систем автоматичного онлайн прокторингу та розробка прототипу такої системи.

**Основна частина.** Як було визначено за результатами аналізу існуючих систем, в них були представлені такі фундаментальні проблеми:

- обробка персональних даних у системах прокторингу є непрозорою і не відповідає вимогам GDPR. Існує значне занепокоєння щодо неможливості легко видалити персональні дані та браку інформації про якість та безпечність зберігання даних;
- функціонування модулів розпізнавання образів у цих системах є непередбачуваним;
- системи висувають надто жорсткі вимоги до поведінки студентів, що призводить до додаткового стресу для студентів;
- повністю відсутній контроль за дотриманням гігієнічних вимог щодо використання комп'ютерів та моніторингу рівня стресу, що може мати пагубний вплив на здоров'я студентів.

Аналіз показав, що розглянуті системи водночас мають завищені вимоги до поведінки студентів (наприклад, відведення погляду від монітора вважається спробою списати) та слабкий рівень аналізу відеопослідовностей (обробка кадрів через невизначений проміжок часу ніяк не показує реальної картини того, як студент працює над екзаменаційним завданням). Ці проблеми в поєднанні з негнучкістю видів запитань для тестування призвели до того, що експерти рекомендують зменшити час, який витрачається на виконання тестових завдань, і зробити їх менш складними.

Це все погано впливає на рівень отримуваних знань. Вирішивши ці проблеми та покращивши прозорість обробки і зберігання даних, а також додавши контроль за рівнем стресу та фізичним здоров’ям студентів, системи прокторингу на основі штучного інтелекту можна вдосконалити, щоб забезпечити більш безпечне та сприятливе навчальне середовище.

Пропонується до наявного функціоналу додати функціонал відстеження емоційного стану та рухів студента. На основі цих змін функціональні вимоги включатимуть у себе комплексний аналіз роботи учня.

Однак варто уточнити, що, судячи з відгуків самих учнів, жорсткий контроль і завищені вимоги спричиняють у них підвищене відчуття стресу та страху. Завищені

вимоги до поведінки учня також дезінформують викладача, оскільки багато систем помилково спрацьовує на незначні рухи студента (якщо, звісно, саме в цей момент система вирішить обробити кадр із даних, отриманих із камери). Тому пропонується переглянути (Рис. 2) наявні вимоги до поведінки студента в бік їх пом'якшення.

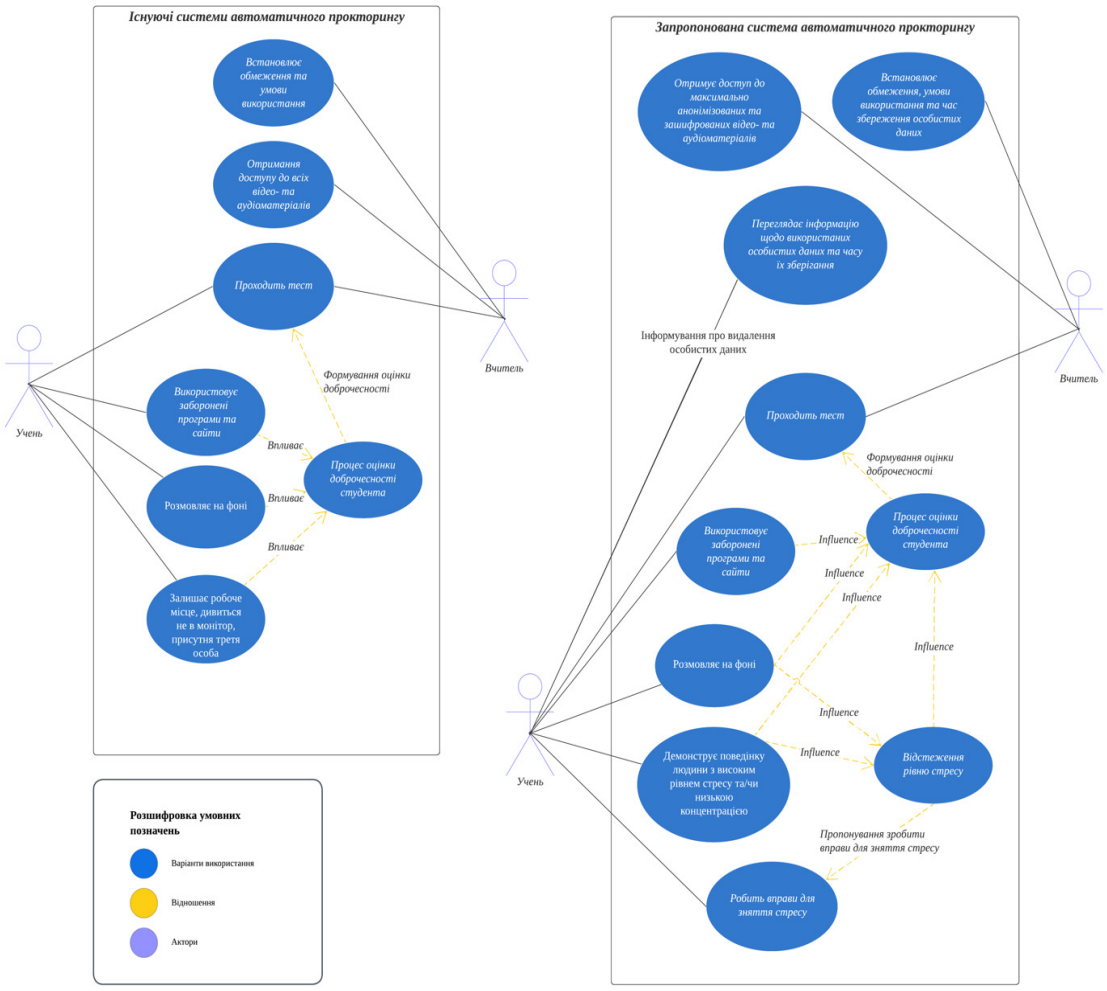



**Рис.2.** Діаграма варіантів використання автоматизованої системи онлайн-прокторингу

*Вдосконалення роботи з персональними даними учнів.* Загальний регламент про захист даних включає до себе такі основні ідеї роботи с персональними даними користувачів програмних продуктів

- користувач має бути проінформованим про те, які дані про нього планується збирати програмний продукт. Також він має отримати чітку інформацію щодо обсягів збору даних, правових підстав для обробки персональних даних, тривалості зберігання даних, передачі даних третім особам або за межі ЄС, а також будь-якого автоматичного прийняття рішень, що здійснюється на основі алгоритмів;
- користувачі мають право в будь-який час відкликати свою згоду на обробку даних, переглядати свої персональні дані та отримувати доступ до інформації про те, як ці дані опрацьовуються;
- користувачі мають право отримувати копії збережених даних, видаляти свої дані за певних умов, оскаржувати автоматизовані рішення, які здійснюються виключно на основі алгоритмів, і подавати скарги до органів захисту даних;
- алгоритми не повинні використовувати персональні дані користувачів, а самі дані повинні зберігатися у безпеці;

- Необхідно забезпечити, щоб обробка даних не була надмірною, а відбувалася лише з необхідною мірою. Це означає, що повинні оброблятися лише ті дані, які абсолютно необхідні для виконання певних цілей або завдань.

Виходячи з цих вимог була сформований алгоритм процесу роботи системи з персональними даними студентів у системі прокторингу, яка враховує всі перераховані вимоги (Рис. 3).

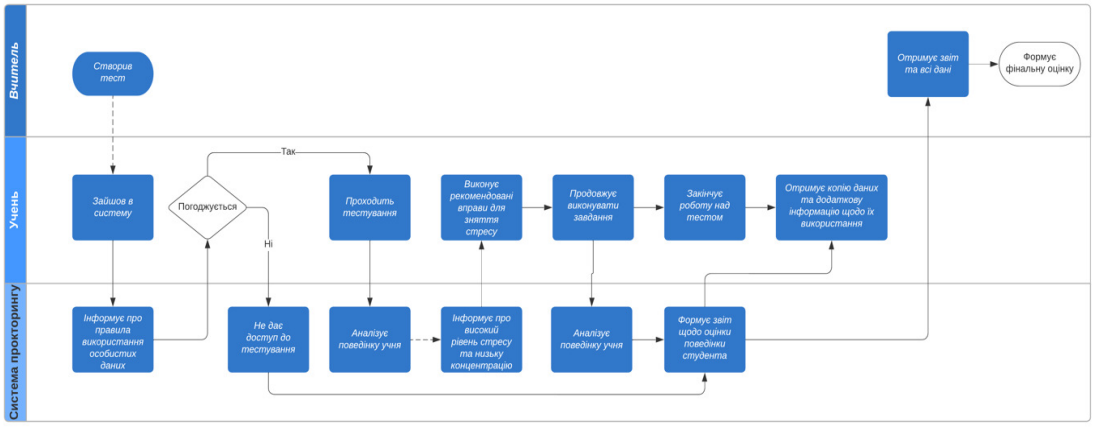



**Рис.3.** Swimlane діаграма комунікації акторів та системи і опис процесу роботи с особистими даними учнів

Для знеособлення даних також пропонується додавати шум до аудіо файлів, щоб не було чути усі деталі розмов, а також при формуванні звіту максимально розмивати відеоряд в місцях, де система машинного навчання з найбільшою вірогідністю констатувала доброчесність учня. Але приймаючи той факт, що для деяких іспитів мають бути представлені усі записи процесу роботи учня, пропонується робити таку постобробку даних за бажанням викладача при формуванні тестового завдання.

*Відстеження та контроль емоціонального та фізичного сану учня.* Емоційних стан учнів дуже важлива частина процесу навчання. Дослідниками був доказаний зв’язок між емоційним станом, його усвідомленням та результатами процесу навчання. Коли студенти усвідомлюють свої емоції та керуються певними стратегіями навчання, їхня навчальна успішність покращується щодо мотивації, залученості та саморегуляції. Так само ситуація складається і з вчителями: коли вчителі усвідомлюють емоційний стан учнів, їхнє ставлення та зворотний зв'язок стають більш ефективними та своєчасними [5].

Як вже було зазначено раніше, сучасні реалізації систем прокторінгу дуже суворо ставляться щодо виконання правил роботи з тестами, але одночасно задля зменшення навантаження на систему та економії ресурсів не аналізують детально відео, а скоріше наборі кадрів раз в декілька секунд чи навіть мінут. Ці жорсткі вимоги та слаба реалізація аналізу відеопослідовностей приводять до того, що системи не в силах адекватно оцінювати поведінку студента: вони то пропустять порушення правил здачі іспитів, то помилково зреагують на моргання або позіхання учня.

Також, як зазначали самі учні [6], інформованість про таку неадекватну поведінку систем та факт постійного спостереження за роботою викликає у людей додатковий стрес. Довга робота за комп’ютерним пристроїм також впливає на рівень стресу та фізичне здоров’я людини.

Враховуючи все вище написане, пропонується додати у такі система комплексний просторово-часовий аналіз поведінки студента, який буде включати як обробку дій учня, так і його/її емоції. Це дозволить своєчасно робити перерви для відновлення

емоціонального стану та допоможе знизити ризик розвитку проблем зі здоров'ям учнів.

Для опису та аналізу емоцій учнів пропонується використовувати виявлення система кодування лицьових (мімічних) рухів. Ця система детально описує рухові одиниці та дескриптори, пов'язані з людським обличчям. Рухові одиниці не залежать від будь-якої конкретної інтерпретації і можуть бути використані для визначення емоцій. ЕмСКЛід - Емоційну систему кодування лицьових рухів (Emotion Facial Action Coding System (EmFACS) [7] і Facial Action Coding System Affect Interpretation Dictionary (FACSAID) [7]), що розглядають рухи обличчя, пов'язані з емоціями. Комбінація кількох рухових одиниць визначає конкретну емоцію (Таб. 1, 2) [9].

**Таблиця 1.**

Приклад рухових одиниць та їх комбінацій у емоції

| Номер | Дії |
|---|---|
| 1 | Внутрішній підйом брів |
| 2 | Зовнішній підйом брів |
| 4 | Опускання брів |
| 9 | Зморщування носа |
| 43 | Очі заплющені |
| 45 | Моргання |

**Таблиця 2.**

Приклад об'єднань рухових лицьових одиниць у емоції

| Назва емоції | Комбінація рухових одиниць |
|---|---|
| Щастя | 6+12 |
| Страх | 1+2+4+5+7+20+26 |
| Сум | 1+4+15 |
| Гнів | 4+5+7+23 |

Ці класифікації дають можливість структурувати різноманіть людських емоцій та зробити опис емоційного стану студента.

Також треба зазначити необхідність впровадження вдосконалених наборів даних для навчання систем комп'ютерного зору розпізнаванню емоцій людей різних національностей, що для уникнення помилок при розпізнаванні обличь та емоцій людей.

Результати роботи модуля аналізу поведінки та емоційного сану учня будуть подальше виконуватися як для звіту для викладача, так і для системи стабілізації емоційного стану учня. Пропонується використовувати маркери високого стресу для рекомендації невеликого перериву та виконання відволікаючих вправ: від руханок [10] до тривіального до тривіального "подивитися у вікно і потерти очі" [11].

*Опис архітектури прототипу автоматизованої системи прокторингу*. Аналіз існуючих рішень показав, що найпростішими в використанні були системи, що інтегрувалися як плагіну в сучасні системи дистанційного навчання. Для прототипу пропонується розробити плагін для Google Forms та Web-додаток, що буде давати можливість викладачу проглядати та налаштовувати правила проведення іспитів. Це дасть можливість мінімізувати та полегшити роботу учнів щодо налаштуванні їх робочого пристрою [12], полегшить роботу викладачеві, якому не доведеться заново піднімати інфраструктуру свого класу тільки заради проведення тестових завдань та мінімізувати проблеми підтримки різноманітних операційних систем.

Фінальна запропонована архітектура системи автоматизованого онлайн прокторингу складається з трьох основних частин (Рис. 4):
- хмарний сервер, на якому буде проводитися аналіз та оцінка роботи учнів у процесі здачу тесту чи іспиту. В нього входять аналізатор метрик, модуль обробки запитів клієнта (API обробки), модуль обробки інтерфейсу користувача, модуль сервісу роботи з базами даних, а також модуль обробки даних користувача та пов'язаний з ним модуль машинного навчання, який саме буде аналізувати поведінку учня та давати ій комплексну оцінку;
- плагін на комп'ютері користувача, що складається з інтерфейсу, анонімайзера даних, інтерфейсу роботи з камерою та модулем роботи з хмарним сервісом через API;
- бази даних для ЄС та США будуть розділені для максимальної підтримки потенціальних користувачів у різних часових зонах, зниження вартості використання сховищ (кожне з них буде працювати у відведений час на більш дешевих тарифах, коли постійна безперервна робота сховищ коштує набагато дорожче []) та задовольняє вимогам роботи з особистими даними користувачів у цих частинах світу.

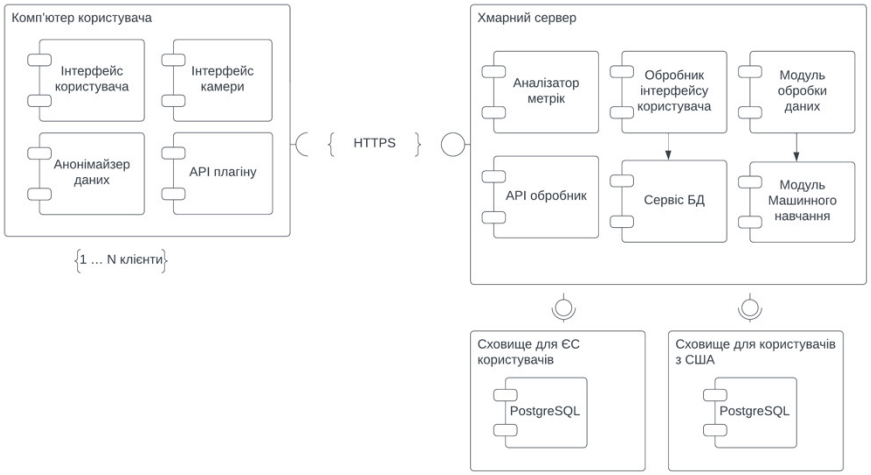



**Рис.4.** Діаграма компонент системи автоматизованого онлайн прокторінгу

**Висновки.** Були вдосконалені вимоги щодо систем автоматичного онлайн прокторингу. Для реалізації більш «людяної» роботи зі студентами було запропоновано пом'якшити вимоги до оцінки їхньої поведінки (наприклад, прибрати необхідність постійно дивитися в монітор). Також рекомендується впровадити систему контролю стресу і пропонувати навчанню виконувати вправи для його зняття. Такий контроль можна реалізувати більш примітивно, виставивши таймер перебування студента перед монітором. Однак у нашій системі пропонується відстежувати емоційний стан учня за допомогою аналізу емоцій і рухів з даних, отриманих з веб-камери.
Щодо безпеки з боку використання персональної інформації, було запропоновано максимально анонімізувати дані студентів після того, як вже була підтверджена їх особистість та за межами підозрілих ситуацій, які можуть свідчити про недоброчесність студента. Також було запропоновано надати студенту детальний доступ щодо умов використання його особистих даних та їх видалення з серверів системи.

**Список літератури**


1. Толмач, М. Цифрові технології в освіті: можливості й тенденції застосування. *ІТ-технології в освіті, мистецтві та культурі*. 17.12.2021. Vol. 4 No 2. P. 159–171. DOI: https://doi.org/10.31866/2617-796X.4.2.2021.247474.
2. Nigam, A, Pasricha, R, Singh, T, Churi, P. A systematic review on ai-based proctoring systems: past, present and future. *Educ Inf Technol (Dordr)*. 23.06.2021. Vol. 26 No 5. P. 6421-6445. DOI: 10.1007/s10639-021-10597-x.
3. ProctorU Abandons Business Based Solely on AI Code. URL: https://www.insidehighered.com/news/2021/05/24/proctoru-abandons-business-based-solely-ai (Дата доступу: 25.06.2023).
4. Critten, J. The Problem with Proctoring (Part 1). URL: https://www.cu.edu/blog/online-teaching-blog/problem-proctoring-part-1 (Дата доступу: 25.06.2023).
5. Arguedas, M., Xhafa, F., Daradoumis, T. Analyzing how emotion awareness influences students' motivation, engagement, self-regulation and learning outcome. *Intelligent and Affective Learning Environments: New Trends and Challenges.* 04.2016. Vol 19 No 2. P. 87-103. URL: https://www.jstor.org/stable/jeductechsoci.19.2.87.
6. Woldeab, D., Brothen, T. 21st Century assessment: Online proctoring, test anxiety, and student performance. *The International Journal of E-Learning & Distance Education.* 2019. Vol. 34 No. 1. URL: https://www.ijede.ca/index.php/jde/article/view/1106.
7. Friesen, W., Ekman, P. EMFACS-7: Emotional Facial Action Coding System. 1983.
8. Clark Elizabeth, A., Kessinger J'Nai, Duncan Susan, E., Bell Martha Ann, Jacob, L., Gallagher Daniel, L., O'Keefe Sean, F. The facial action coding system for characterization of human affective response to consumer product-based stimuli: a systematic review. *Frontiers in Psychology.* 2020. Vol 11. DOI: https://doi.org/10.3389/fpsyg.2020.00920.
9. Farnsworth, B. Facial action coding system (FACS) – A Visual Guidebook. URL: https://imotions.com/blog/learning/research-fundamentals/facial-action-coding-system/ (Дата доступу: 20.06.2023).
10. Руханки та вправи для зняття напруження. URL: https://mon.gov.ua/ua/osvita/doshkilna-osvita/suchasne-doshkillya-pid-krilami-zahistu/vseukrayinskij-naukovo-metodichnij-zhurnal-doshkilne-vihovannya/ruhanki-ta-vpravi-dlya-znyattya-napruzhennya (Дата доступу: 15.04.2020)
11. Safely using computers at work: Guidelines for using computers. URL: https://www.worksafe.govt.nz/topic-and-industry/work-related-health/musculoskeletal-disorders/ergonomics/safely-using-computers-at-work/ (Дата доступу: 01.07.2023)
12. Відгуки до одної с систем прокторінгу (ProctorU). URL: https://chrome.google.com/webstore/detail/proctoru/goobgennebinldhonaajgafidboenlkl (Дата доступу: 03.07.2023).
13. Amazon RDS pricing. URL: https://aws.amazon.com/rds/pricing/ (Дата доступу: 20.06.2023)

А.А. Брескіна

# DEVELOPMENT OF AN AUTOMATED ONLINE PROCTORING SYSTEM

A.A. Breskina

National Odesa Polytechnic University,
1, Shevchenka Ave. Odesa, 65044, Ukraine; e-mail: anastasia.breskina@gmail.com

The rapid development of machine learning technologies, the increasing availability of devices and widespread access to the Internet have significantly contributed to the growth of distance learning. Alongside distance learning systems, proctoring systems have emerged to assess student performance by simulating the work of a teacher. However, despite the development of image processing and machine learning technologies, modern proctoring systems still have limited functionality: some systems have implemented computer vision methods and algorithms so poorly (false positives when working with students of different nationalities) and classification of student actions (very strict requirements for student behaviour) that some software products have even refused to use modules that use elements of artificial intelligence. It is also a problem that current systems are mainly focused on tracking students' faces and gaze and do not track their postures, actions, and emotional state. However, it is the assessment of actions and emotional state that is crucial not only for the learning process itself, but also for the well-being of students, as they spend long periods of time at computers or other devices during distance learning, which has a great impact on both their physical health and stress levels. Currently, control over these indicators lies solely with teachers or even students themselves, who have to work through test materials and independent work on their own. An additional problem is the quality of processing and storage of students' personal data, as most systems require students to be identified using their identity documents and store full, unanonymised video of students' work on their servers. Based on the analysis of all these problems that impede the learning process and potentially endanger students' health in the long run, this article presents additional functional requirements for modern automated online proctoring systems, including the need to analyse human actions to assess physical activity and monitor hygiene practices when using computers in the learning process, as well as requirements for maximum protection of students' personal data.

**Keywords:** distance learning, automated online proctoring systems, personal data protection, analysis of human emotions, analysis of human actions.

**ІНФОРМАТИКА ТА МАТЕМАТИЧНІ МЕТОДИ В МОДЕЛЮВАННІ**

Том 13, номер 1-2, 2023. Одеса – 190 с., іл.

**ИНФОРМАТИКА И МАТЕМАТИЧЕСКИЕ МЕТОДЫ В МОДЕЛИРОВАНИИ**

Том 13, номер 1-2, 2023. Одесса – 190 с., ил.

**INFORMATICS AND MATHEMATICAL METHODS IN SIMULATION**

Volume 13, No. 1-2, 2023. Odesa – 190 p.

---

**Засновник:** Національний університет «Одеська політехніка»

Зареєстровано Міністерством юстиції України 04.04.2011р.
Свідоцтво: серія КВ № 17610 - 6460Р

Друкується за рішенням Вченої ради Національного університету «Одеська політехніка», (протокол № 9 від 26.04.2023р.)

**Адреса редакції:** Національний університет «Одеська політехніка»,
проспект Шевченка, 1, Одеса 65044 Україна

Web: www.immm.op.edu.ua (immm.opu.ua)

E-mail: immm.ukraine@gmail.com