МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Одеська політехніка»

ІНФОРМАТИКА ТА МАТЕМАТИЧНІ МЕТОДИ В МОДЕЛЮВАННІ

INFORMATICS AND MATHEMATICAL METHODS IN SIMULATION

Том 14, № 3

Volume 14, No. 3

Одеса – 2024
Odesa – 2024

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 КВ № 17610 - 6460Р of 04.04.2011

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

Оригінал-макет виготовлено редакцією журналу

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

ЗМІСТ/CONTENTS

DOI 10.15276/imms.v14.no3.123
УДК 004.04

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No.3, pp. 123-133

DEVICE AUTHENTICATION METHOD IN INTERNET OF THINGS NETWORKS

A.Ya. Davletova

West Ukrainian National University
11, Lvivska Str. Ternopil, 46009, Ukraine
Email: a7davletova@gmail.com

This work addresses the pressing issue of ensuring secure and reliable data transmission, along with efficient cryptographic key management in IoT networks. Typical authentication protocols, though providing secure communication, can be overly complex for many resource-constrained devices. This highlights the need to explore and identify efficient solutions suited to the capabilities of IoT devices, ensuring robust encryption and authentication. The paper presents a hybrid authentication algorithm, RHAA (RSA and Hamming Code-based Authentication Algorithm), designed to guarantee confidentiality, integrity, and authentication of data in IoT networks with numerous nodes. The key feature of the proposed algorithm is the combination of RSA-based asymmetric encryption with the error-correcting capabilities of Hamming code in finite fields. This approach ensures device authentication during information exchange through centralized key generation and management. Data protection from unauthorized access is achieved through re-encryption at each node. Data integrity is maintained by detecting and correcting errors at each transmission stage. This solution enhances IoT network security by reducing the risk of data leakage or loss. The work also includes an example of the RHAA algorithm implementation with real key values, demonstrating the system's response to errors and their correction. The proposed algorithm, optimized for resource-limited devices, can be applied to improve data protection in IoT networks.

Keywords: authentication, IoT networks, RSA algorithm, Hamming code in finite fields, encryption, data integrity, error detection and correction, privacy, data security.

Introduction. With the development of information systems and technologies, the importance of implementing effective methods for information protection and secure data exchange is becoming increasingly apparent. The active digitalization of key sectors of activity [1], the widespread use of Internet of Things (IoT) technologies, and the growing reliance on cloud services with remote access significantly increase the risks of misuse and cyberattacks [2].

The growing number of connected devices across various fields, from smart home automation to industrial systems, necessitates ensuring secure communication. The decentralized architecture complicates control over network devices and creates new opportunities for exploiting system vulnerabilities without the proper level of protection [3]. In such conditions, to prevent unauthorized access and avoid attacks aimed at spoofing or exploiting malicious devices, authentication mechanisms based on cryptography become crucial. The use of robust cryptographic methods ensures data confidentiality and the identification of communication participants. The properties of error-correcting codes guarantee the integrity of transmitted messages.

Many IoT devices are characterized by certain hardware limitations, such as low computational power, limited energy supply, and memory capacity [4]. This necessitates addressing the issue of managing security mechanisms, particularly cryptographic keys, including their generation, storage, and processing. A hybrid algorithm that combines various cryptographic methods and centralized security management could become an effective solution for enhancing the integrity of data transmission and the security of IoT device authentication.

Analysis of research and publications. IoT devices play an important role in data collection and processing across various fields, such as healthcare, industry, home automation, and more.

However, their widespread use exposes new vulnerabilities that can be exploited to gain unauthorized access to devices, intercept data, or perform other malicious actions [5]. Typically, IoT devices can be configured via remote control or through proprietary controllers developed by manufacturers [6].

This creates opportunities for sending malicious instructions directly to the device. Moreover, data from certain IoT devices are transmitted to edge servers for further processing, opening new channels for attacks [7].

By using specific types of attacks, such as radio attacks or reverse engineering attacks, malicious actors can compromise the security of the network [8]. Inadequate security levels can lead to serious consequences for data confidentiality and integrity. In response to these challenges, research in this field is focused on developing methods for securing data exchange and creating reliable authentication schemes.

In [9], a secure key exchange method and an authentication scheme are presented, aimed at enhancing communication security in the IoT environment. The proposed approach combines the security functions of elliptic curve cryptography (ECC) with the Elliptic Curve Diffie-Hellman (ECDH) key exchange mechanism. Research shows that the protocol is characterized by low energy consumption and computational costs, making it efficient for IoT nodes operating under resource constraints.

In [10], an enhanced authentication system for IoT is proposed, which includes the use of technologies such as blockchain, artificial intelligence, and biometrics. This approach provides a dynamic and adaptive authentication process, improving the security and accuracy of user and device identification. Research shows that the presented solution enhances security levels through the integration of multi-factor authentication, certificate-based authentication, robust encryption, identity management, and a zero-trust model, contributing to an increase in resilience against threats, including DDoS attacks, by up to 80%.

In [11], a re-encryption proxy server scheme for IoT environments is proposed. This approach utilizes a key update mechanism at specified time intervals, supporting user identity revocation. The proposed scheme ensures high functionality and security, demonstrating resilience against attacks based on the DBDH assumption, and does not require high computational costs, making it effective for fog-based cloud environments. However, the scheme has limitations regarding resilience to quantum attacks and may impose a burden on low-power or low-capacity IoT devices due to the complexity of encryption and decryption.

In [12], a security model for IoT networks is presented, which includes blockchain and a post-quantum secure identification scheme (PQ-IDS). The proposed digital signature mechanism, based on lattice-based cryptography, ensures their non-repudiation. Research shows that the proposed approach provides security properties such as unforgeability, non-repudiation, and non-transferability. Comparisons of efficiency and performance assessments indicate that the proposed PQ-IDS exhibits high effectiveness and practicality in IoT network applications.

The conducted analysis confirms that research and development of authentication methods and secure communication in IoT device networks are crucial for ensuring the confidentiality, integrity, and availability of data. Developing solutions that not only meet efficiency and adaptability requirements but also provide robust protection against potential threats is an urgent task.

The aim of this work is to develop a hybrid authentication algorithm that combines the properties of encryption and error-correcting codes, aimed at enhancing the reliability and security of data transmission, as well as the effective management of cryptographic keys in conditions of limited hardware resources.

Research on IoT Device Authentication Algorithms. Authentication for IoT devices is a crucial step and an integral part of security in cyber-physical systems. It ensures the protection of data and devices, as well as guarantees the integrity and reliability of the overall system. The goal of authentication is to establish a unified security policy, minimize risks, and ensure secure

interactions between devices, users, and servers within the IoT environment.

Among the common technologies and protocols that ensure security and efficiency in the IoT environment, MQTT, CoAP, DTLS, and PUF stand out. Their use facilitates secure communication between resource-constrained devices, enhancing the reliability and protection of IoT systems.

The MQTT (Message Queuing Telemetry Transport) messaging protocol is specifically designed for low-power devices and networks with limited resources [13]. It utilizes a "publish-subscribe" architecture (Fig. 1), allowing devices to exchange data without a constant connection. MQTT does not include built-in security mechanisms. For secure communication, external mechanisms such as TLS/SSL are used, which require additional computational overhead.

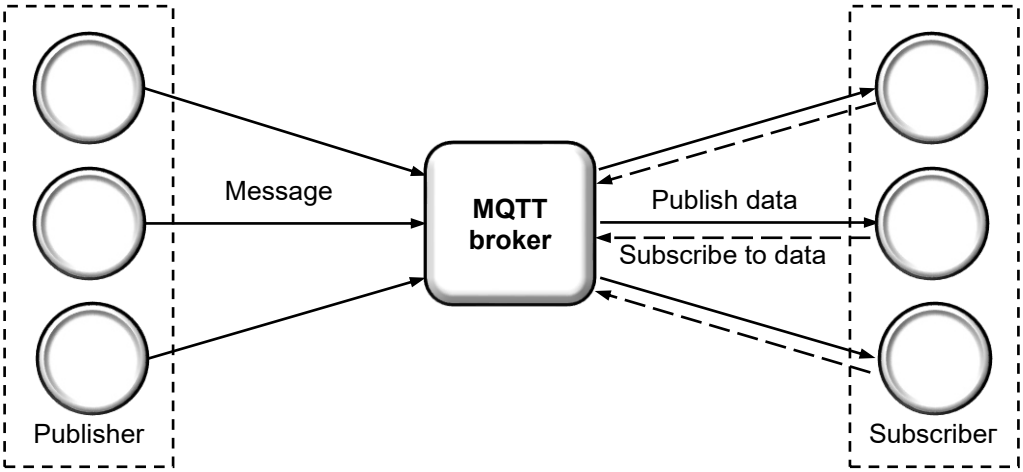


Fig. 1. MQTT Structure

Currently, the strategy of using a single broker is considered inefficient, making it advisable to implement a distributed strategy with multiple brokers in IoT networks, as this enhances the reliability and scalability of the system [14]. The widespread use of MQTT is attributed to its capabilities and features, including connectionless communication, high scalability, low energy consumption, and fast and reliable message delivery [15]. The protocol is applied in monitoring and management systems, as well as for effective routing in IoT [16].

The CoAP (Constrained Application Protocol) is used for interaction between IoT devices and servers. It provides unique addressing for each device, simplifying communication. CoAP is specifically designed for resource-constrained devices [17] and utilizes a simple request/response architecture (Fig. 2).

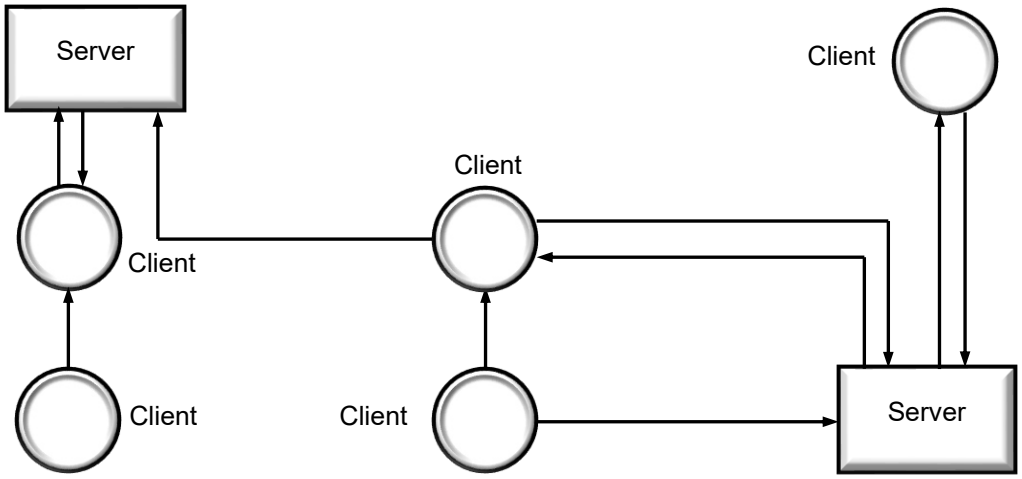


Fig. 2. CoAP Structure

CoAP is an adaptation of HTTP for use in IoT, as standard web communication

protocols are considered "heavy" for IoT devices due to their memory, computational power, and energy consumption requirements. Despite its advantages, such as low energy and bandwidth requirements, CoAP is vulnerable to distributed denial-of-service (DDoS) attacks. To ensure security, such as authentication or encryption, the protocol is often integrated with DTLS (Datagram Transport Layer Security), which provides transport security for CoAP [18].

DTLS is a security protocol that provides encryption and authentication for data transmitted over unreliable networks (Fig. 3). It is an adaptation of TLS (Transport Layer Security) designed to work with data transmitted in datagrams [19].

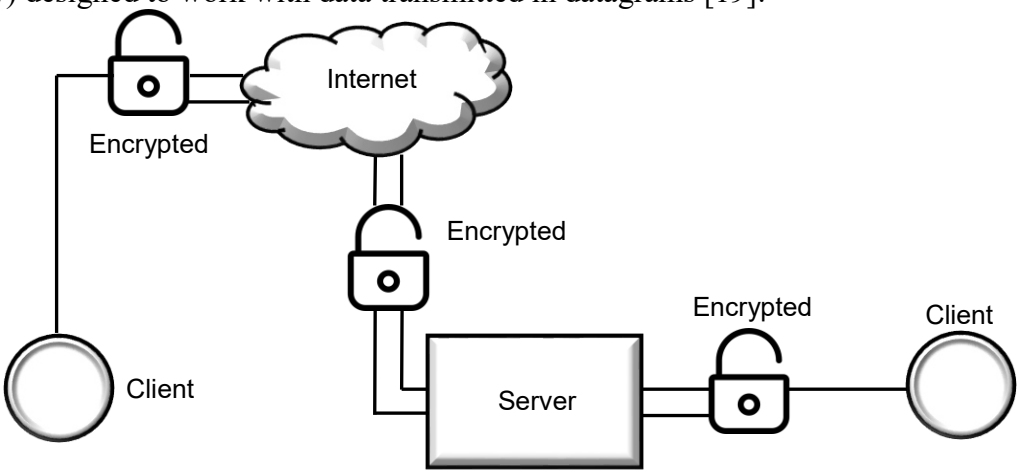


Fig. 3. DTLS Security Protocol

Although DTLS is optimized for constrained devices, it still requires significant computational resources to perform cryptographic operations, particularly the Diffie-Hellman key exchange and certificate verification, which ensure the authenticity of communications.

Authentication of IoT devices based on Physically Unclonable Functions (PUF) is a modern approach to ensuring security in IoT networks. A PUF is a hardware mechanism that utilizes the unique physical characteristics of each device to create a cryptographic identifier that cannot be copied or forged. Based on these unique characteristics, it can generate distinct responses to specific challenges (challenge-response pairs, CRP), which are used for device authentication [20].

The steps of authentication using PUF (Fig. 4) include an initial setup phase, during which each device generates a set of CRP that are stored in a secure repository. For authentication, the server sends a challenge to the IoT device, which uses its unique PUF to compute the response to the challenge. The response is sent back to the server, where it is compared with the stored data.

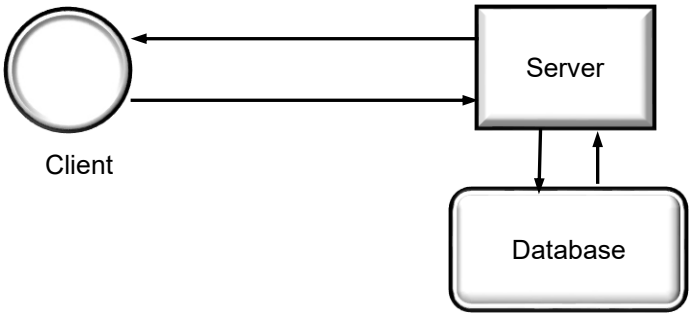


Fig. 4. PUF-Based Authentication Protocol in IoT

The advantages of this method include high resistance to spoofing and cloning, along with low computational resource requirements. However, it is sensitive to changes in the physical characteristics of the devices and vulnerable to challenge-response replay attacks [21].

The use of the discussed protocols enables secure communication between resource-constrained devices, enhancing the reliability and security of IoT systems. For IoT devices,

traditional protocols may be excessive. The high computational requirements necessitate significant resources and energy, which limits their use for low-power IoT devices. Often, IoT devices lack reliable mechanisms for securely storing cryptographic keys, creating a threat of security compromise.

Despite the fact that protocols like MQTT, CoAP, and DTLS can provide device authentication in IoT, they do not address the issue of data integrity during transmission. These protocols focus on ensuring secure communication and authentication; however, they often leave the protection of data from modifications or loss during transmission unaddressed. This creates potential risks for IoT systems, as data integrity breaches can lead to malfunctioning devices or vulnerabilities within the network.

Operation of the Proposed Hybrid Authentication Algorithm. The proposed hybrid authentication algorithm RHAA (RSA and Hamming Code-based Authentication Algorithm), by combining encryption and coding, contributes to ensuring the confidentiality and integrity control of data in systems with a large number of nodes. This approach enhances the security and reliability of transmitted data and can be implemented to protect IoT device networks from various cyber threats.

By using the RSA algorithm at each encryption stage, the data remains protected from unauthorized access [22]. RSA allows for the use of small key sizes, which reduces memory requirements for storage and ensures authentication and secure data transmission between IoT devices. The construction of the Hamming code $GF(n)$ provides the ability to detect and correct errors that may occur during transmission, thereby reducing the risk of data loss or corruption [23]. The error-correcting properties of the code offer advantages for IoT device networks, as the quality of communication may be unstable. The Hamming code is computationally simple, using minimal memory resources. Its implementation in finite fields GF(p) does not require additional overhead. The RHAA is flexible and scalable, allowing for the collection, processing, and transmission of large volumes of data from various sources.

In Figure 5, the structure of the proposed RHAA algorithm is presented.

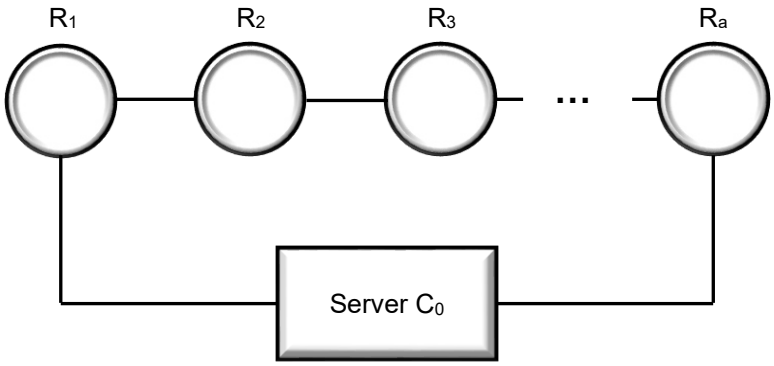


Fig. 5.– Structure of RHAA

The principle of operation of the proposed algorithm:

1. Setup Stage:

- Key Generation - for each node$R_1, R, \dots, R_a$ a pair of RSA keys is generated: public keys (e_i, n) and private keys (d_i, n). The public keys are transmitted to each corresponding node, while the private keys are stored securely on the central server C.

2. Message Preparation:

- Verification Initiation - at defined time intervals, the central server C initiates a verification process by sending a control message containing specially formatted data that must sequentially pass through all nodes.

- The outgoing message m is encrypted on the central server C using the public key of R_0 with the RSA algorithm

$$c_0 = m^{e_0} \, mod \, n.$$

- The encrypted message c_0 is transformed into a codeword x_0 by constructing a Hamming code in finite fields $GF(n)$, which ensures the detection and correction of errors during transmission.

3. Transmission of the codeword through intermediate nodes $R_1, R_2, \dots, R_a$. Each node R_i receives the codeword x_i for which the following occurs:

- Data integrity check and, if necessary, correction.
- Decoding of the message by removing the parity symbols, resulting in the encrypted message m_i.
- Re-encryption of m_i using the public key of the next node R_{i+1}

$$c_{i+1} = m_{i+1}^{e_{i+1}} \, mod \, n.$$

- Converting c_{i+1} to codeword x_{i+1}, which is then passed to the next node R_{i+1}. The last node R_a transmits the code word x_a to the center C.

4. Decryption and data integrity verification:

- The central server C receives the codeword x_a, checks and corrects errors, removes the parity symbols, resulting in the encrypted message m_a.
- The central server C sequentially decrypts the message using the corresponding private keys for each node (d_i, n)

$$m_i = c_i^{d_i} \, mod \, n.$$

- If all decryption stages are successful, the central server C restores the original message m.

5. Intrusion Detection:

- If the original message cannot be restored during the verification process, it indicates a potential compromise or intrusion. This means that at least one of the intermediate nodes may have been breached, resulting in data loss or tampering, which jeopardizes the integrity and confidentiality of the transmitted information.

Since the data passes through several intermediate nodes, ensuring its confidentiality and integrity is crucial. The integration of the RSA algorithm provides data encryption and node authentication, preventing unauthorized access to the information. The construction of Hamming codes in $GF(n)$ ensures additional encryption and guaranteed detection and correction of a single error at all stages of data transmission. The central server C performs regular checks to detect potential security threats. In the event of an intrusion detection, it is recommended to change access keys immediately, which will help prevent further access by malicious actors and assist in restoring the integrity of the system.

The algorithm for the operation of RHAA is shown in Fig. 6.

To illustrate the operation of the proposed RHAA algorithm, let's consider an example.

Suppose the following values are chosen for key generation: $p = 71$, $q = 101$.

Consequently, we obtain the values $\phi(n) = 7000, n = p * q = 7171$.

The number of nodes is $R = 4$.

The control message is $m = 12, 35, 151, 569, 74, 84, 357$.

As a result of the key generation phase, we obtain the following key pairs:

R_0: $e_0 = (9,\ 7171)$; $d_0 = (3889,\ 7171)$.

R_1: $e_1 = (11,\ 7171)$; $d_1 = (5091,\ 7171)$.

R_2: $e_2 = (13,\ 7171)$; $d_2 = (1077,\ 7171)$.

R_3: $e_3 = (17,\ 7171)$; $d_3 = (5353,\ 7171)$.

R_4: $e_4 = (19,\ 7171)$; $d_4 = (2579,\ 7171)$.

After encrypting the output message using the public key $e_0 = (9,\ 7171)$, we can calculate the encrypted message c_0 using the RSA encryption formula:

$$c_0 = 2038, 6300, 4516, 5539, 1223, 1711, 2145.$$

The result of converting c_0 into a codeword in the field $GF(7171)$ is:

$$x_0 = 2903, 1607, 2038, 2013, 6300, 4516, 5539, 5079, 1223, 1711, 2145.$$

The transmission of the codeword through intermediate nodes R_1 to R_4 may be

accompanied by the occurrence of errors. For example, the message received by node R_1 will look like this:

$$x_0' = 2903, 16, 2038, 2013, 6300, 4516, 5539, 5079, 1223, 1711, 2145.$$

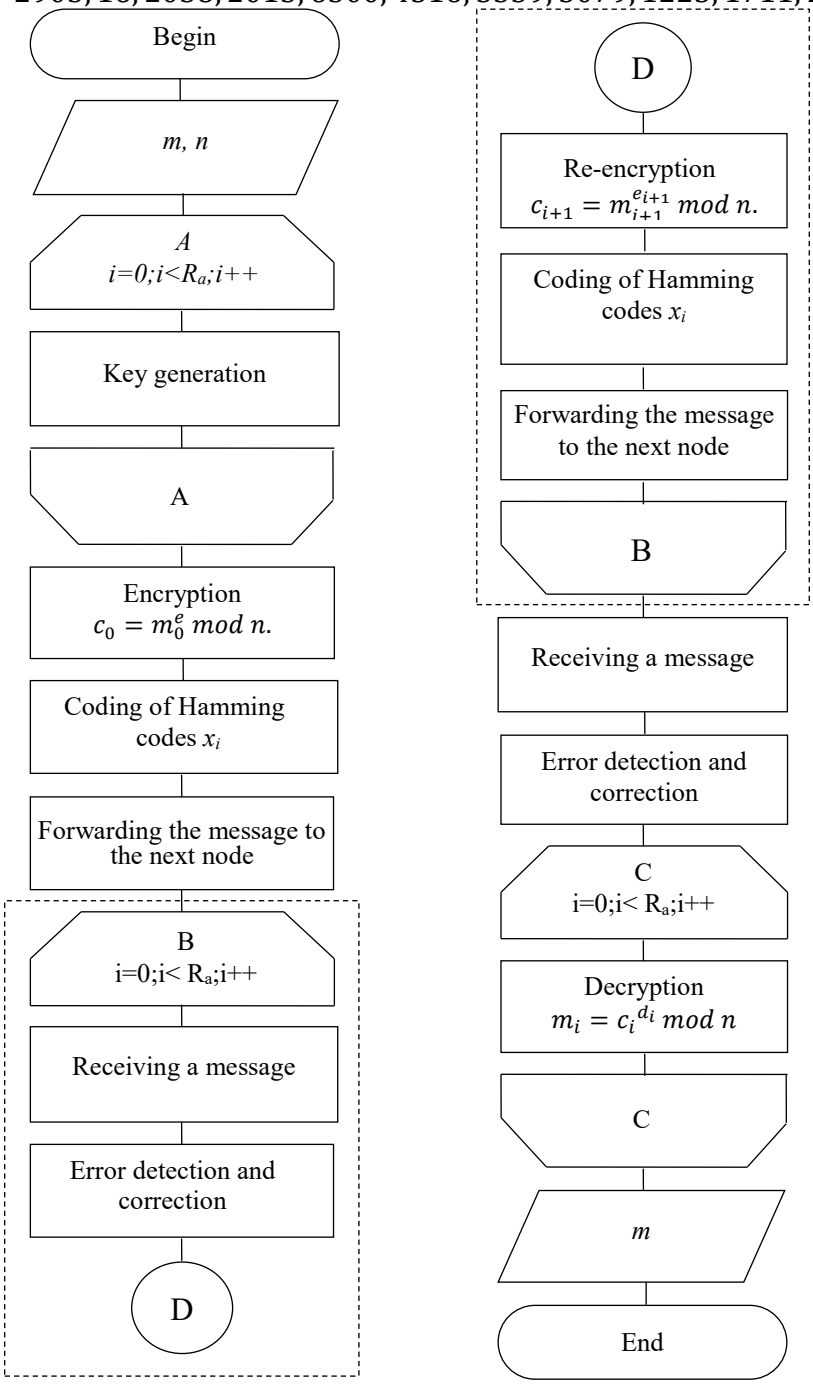


Fig. 6. Algorithm of RHAA Operation

As a result of the integrity check (Fig. 7), an error was detected in position 2, which corresponds to a check symbol that does not require correction.

```
Pariti bits with eror code from data (calculated)  [2903, 1607, 2013, 5079]
Pariti bits with eror code from      (received)    [2903, 16, 2013, 5079]
Result of compare                                  [False, True, False, False]
Error position  2
```

Fig. 7. Error Detection Process

The result of further decoding and encryption using the public key $e_1 = (11, 7171)$ will be the encrypted message:

$$c_1 = 2281, 6894, 6260, 4474, 86, 3226, 4199.$$

As a result of constructing the Hamming code in the field $GF(n)$ from the message c_1, we have:

$$x_1 = 3592, 6098, 2281, 3286, 6894, 6260, 4474, 340, 86, 3226, 4199.$$

The message received by node R_2, i considering potential distortion during transmission, will appear as follows:

$$x_1' = 3592, 6098, 2281, 3286, 105894, 6260, 4474, 340, 86, 3226, 4199.$$

As a result of the data integrity check, an error was detected in position 5. After correction, we obtain:

$$x_1 = 3592, 6098, 2281, 3286, 6894, 6260, 4474, 340, 86, 3226, 4199.$$

After decoding and re-encrypting using the public key $e_2 = (13, 71791)$, we obtain:

$$c_2 = 5010, 1022, 5241, 6036, 3528, 1905, 322.$$

In the message received by node R_3, no modifications were detected during the integrity check, so no correction is needed. Therefore, we will use the received message for further processing:

$$x_2 = 1576, 4172, 5010, 5128, 1022, 5241, 6036, 5755, 3528, 1905, 322.$$

As a result of decoding and decrypting using the public key $e_3 = (17, 7171)$, we obtain

$$c_3 = 714, 3865, 5567, 5752, 1927, 5448, 371.$$

The codeword for transmission to node R_4 will be as follows:

$$x_3 = 5458, 3510, 714, 842, 3865, 5567, 5752, 575, 1927, 5448, 371.$$

During transmission, the data were corrupted; therefore, the codeword received by node R_4 appears as follows:

$$x_3' = 5458, 3510, 714, 842, 3865, 5567, 1, 575, 1927, 5448, 371.$$

An error is detected in position 7, corresponding to the information symbol, which requires correction. This correction process is carried out similarly to the previous stages.

As a result of encrypting with the key $e_4 = (19, 7171)$, we obtain::

$$c_4 = 3345, 3469, 858, 569, 1084, 4528, 2478.$$

The central server C receives the codeword:

$$x_4' = 3774, 4607, 3345, 4896, 3469, 858, 9503, 919, 1084, 4528, 2478.$$

The integrity check detected an error in position 7, and after correction, we obtain:

$$x_4 = 3774, 4607, 3345, 4896, 3469, 858, 569, 919, 1084, 4528, 2478.$$

The check bits are no longer needed, and by discarding them, we obtain the message for decryption:

$$c_4 = 3345, 3469, 858, 569, 1084, 4528, 2478.$$

In order to detect an intrusion, data is decrypted step by step using private keys $d_0 - d_4$ at each step (Fig. 8).

```
Decripted message 1  [3345, 3469, 858, 569, 1084, 4528, 2478]
Using private key  (2579, 7171)
 private key  4
Decripted message  [714, 3865, 5567, 5752, 1927, 5448, 371]
Using private key  (5353, 7171)
 private key  3
Decripted message  [5010, 1022, 5241, 6036, 3528, 1905, 322]
Using private key  (1077, 7171)
 private key  2
Decripted message  [2281, 6894, 6260, 4474, 86, 3226, 4199]
Using private key  (5091, 7171)
 private key  1
Decripted message  [2038, 6300, 4516, 5539, 1223, 1711, 2145]
Using private key  (3889, 7171)
 private key  0
Decripted message  [12, 35, 151, 569, 74, 84, 357]
```

Fig. 8. Decryption process

The proposed RHAA algorithm, as a component of a comprehensive information protection system, ensures the confidentiality and integrity of data, providing a high level of protection and accuracy of information, as well as detecting anomalies, which helps to detect potential attacks. Using the proposed approach provides protection against the following types of attacks:

- Data Integrity Attacks - coding protects against data changes that occur when attempts are made to replace or modify data, for example, when traffic parameters are changed or part of the information is destroyed.

- Man-In-The-Middle (MITM) Attacks - encrypted data using the RSA algorithm is difficult to forge. Even if an attacker intercepts the message, altering the ciphertext will lead to errors that reveal the characteristics of the correction codes.;

- Data Interception - thanks to multiple encryption layers, an attacker will not be able to decrypt the intercepted data without the appropriate keys, which reduces the risk of losing confidential information.

- Unauthorized Access - in the case of node compromise, the system may detect a discrepancy in the keys, which signals a breach.

- Replay Attacks - regular checking ensures the detection of attempts to reuse encrypted messages.

- Confidentiality Attacks - encryption and coding ensure a high level of confidentiality, protecting data from unauthorized access during transmission.

The proposed RHAA is easy to adapt or expand according to growing requirements or changes in the environment. The proposed approach is more adaptive, compared to traditional authentication methods that focus only on identifying and confirming devices or users. RHAA actively monitors the quality of data transmission and provides solutions to eliminate problems. The combination of error detection and correction mechanisms with re-encryption improves efficiency in ensuring data integrity and security in cyber-physical systems with a large number of nodes, where the risk of errors is significantly increased.

Conclusions. The proposed RHAA algorithm provides detection of device compromise thanks to key loss monitoring and node authentication functions. This allows timely response to potential threats and unauthorized access attempts. The use of correction codes ensures the detection of errors of any size in the symbols of the code word and performs the correction of single errors in the data block. Re-encryption and coding make it possible to increase the security level by R times, which makes RHAA more reliable and adaptable to modern data transmission conditions.

The combination of RSA and Hamming code construction in finite fields provides an effective authentication scheme for IoT devices, with the correct choice of encryption parameters, for example, using keys of a given length, to minimize IoT resources.

References

1. Shopina I. Information security of digital transformation. *Scientific Bulletin of the Lviv State University of Internal Affairs. Legal series*. 2023. No. 1. P.28-35. doi: https://doi.org/10.32782/2311-8040/2023-1-4.
2. Pöhn D., Hommel W. Towards an Improved Taxonomy of Attacks related to Digital Identities and Identity Management Systems. *Security and Communication Networks*. 2024. doi: 10.48550/arXiv.2407.16718.
3. Mumin A., Hammoudeh M., Alrawashdeh R., Alsulaimy B. A Survey on Security, Privacy, Trust, and Architectural Challenges in IoT Systems. *IEEE Access*. 2024. PP.(99): 1-1. doi: 10.1109/ACCESS.2024.3382709.
4. Srivastava N., Pandey P. Internet of things (IoT): Applications, trends, issues and challenges. *Materials Today: Proceedings.* 2022. V. 69/2. P. 587-591. doi: 10.1016/j.matpr.2022.09.490
5. Ibibo J.T. IoT Attacks Countermeasures: Systematic Review and Future Research Direction. *BDTA 2023, LNICST 555.* 2023. P. 95–111. doi: 10.1007/978-3-031-52265-9_7.

6. Arora R., Muqeem M., Saxena M.. Developing a Comprehensive Security Framework for Detecting and Mitigating IoT device Attack. 2024. doi: 10.21203/rs.3.rs-5165811/v1.
7. Yilmaz S., Dener M. Security with Wireless Sensor Networks in Smart Grids: A Review. *Symmetry.* 2024. 16(10): 1295. doi: 10.3390/sym16101295
8. Kalaria R., Kayes A.S.M., Rahayu W., Pardede E., Salehi S. A. IoTPredictor: A security framework for predicting IoT device behaviours and detecting malicious devices against cyber attacks. *Computers & Security.* 2024. V. 146. doi: 10.1016/j.cose.2024.104037.
9. Peivandizadeh A., Haitham Y. A., Molavi B., Mohajerzadeh A., Al-Badi H. A. A Secure Key Exchange and Authentication Scheme for Securing Communications in the Internet of Things Environment. *Future Internet* 2024. V.16. No. 16(10). P. 357. doi: 10.3390/fi16100357
10. Maiwada U.D., Danyaro K.U., Janisar A.A., Abdullahi M. Enhancing Security of 5G-Enabled IoT Systems through Advanced Authentication Mechanisms: A Multifaceted Approach. *UMYU Scientifica*. 2024. V. 2. No. 4. P. 201-2011. doi: 10.56919/usci.2324.025.
11. Lin H.-Y., Chen P.-R. Revocable and Fog-Enabled Proxy Re-Encryption Scheme for IoT Environments. *Sensors.* 2024. V.24. No. 19: 6290. doi: 10.3390/s24196290
12. Zhang Y., Tang Y., Li C., Zhang H., Ahmad H. Post-Quantum Secure Identity-Based Signature Scheme with Lattice Assumption for Internet of Things Networks. *Sensors* 2024. V.24, No. 13: 4188. doi: 10.3390/s24134188
13. Abdulelah H., Mohammad I. Effective Feature Engineering Framework for Securing MQTT Protocol in IoT Environments. *Sensors*. 24. 1782. 2024. doi: 10.3390/s24061782.
14. Hmissi F., Ouni Se. TD-MQTT: Transparent Distributed MQTT Brokers for Horizontal IoT Applications. *2022 IEEE 9th International Conference on Sciences of Electronics, Technologies of Information and Telecommunicatio*ns (SETIT)*,* Hammamet, Tunisia. 2022. P. 479-486. doi: 10.1109/SETIT54465.2022.9875881.
15. Lakshminarayana S., Praseed A., Thilagam P. Securing the IoT Application Layer from an MQTT Protocol Perspective: Challenges and Research Prospects. *IEEE Communications Surveys & Tutorials*. 2024. P. 1-1. doi: 10.1109/COMST.2024.3372630.
16. Sonam, Johari R., Garg S., Bawa P., Aggarwal D. MIAWM: MQTT based IoT Application for Weather Monitoring. *Journal of High Speed Networks*. 2024. V.30. P.1-22. doi: 10.3233/JHS-230008.
17. Tariq M.A., Khan M., Khan M.T.R., Kim D. Enhancements and Challenges in CoAP-A Survey. *Sensors*. 2020. V.20. P.6391. doi: 10.3390/s20216391.
18. Westphall J., Loffi L., Merkle Westphall C., Martina J. CoAP + DTLS: A Comprehensive Overview of Cryptographic Performance on an IOT Scenario. *IEEE Sensors Applications Symposium (SAS)*. 2020. P.1-6. doi: 10.1109/SAS48726.2020.9220033.
19. Restuccia G., Tschofenig H., Baccelli E. (2020). Low-Power IoT Communication Security: On the Performance of DTLS and TLS 2020. 1.3. doi: 10.48550/arXiv.2011.12035.
20. Tun W.N., Mambo M. Secure PUF-Based Authentication Systems. *Sensors*. 2024. 24(16): 5295. doi: 10.3390/s24165295.
21. Rajput S., Dofe J. Secure Dynamic PUF for IoT Security. Internet of Things. *Advances in Information and Communication Technology*. 2023. P.454-462. doi: 10.1007/978-3-031-45878-1_33.
22. Rivest R.L. Shamir A., Adleman L.M.A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*. 1978. V. 21. No. 2. P. 120-126. doi: 10.7551/mitpress/12274.003.0047
23. Davletova A. Побудова кодів Хеммінга в скінченних полях Галуа. *Herald of Khmelnytskyi National University. Technical Sciences. 2024.* V.333(2). P.28-34. doi: 10.31891/2307-5732-2024-333-2-4.

МЕТОД АВТЕНТИФІКАЦІЇ ПРИСТРОЇВ В МЕРЕЖАХ ІНТЕРНЕТ РЕЧЕЙ

А.Я. Давлетова

Західноукраїнський національний університет
11, Львівська, м. Тернопіль, 46009, Україна
Email: a7davletova@gmail.com

Робота присвячена вирішенню актуальної задачі забезпечення безпечної та надійної передачі даних та ефективного управління криптографічними ключами в мережах IoT. Типові протоколи автентифікації, хоча й забезпечують захищену комунікацію, можуть бути занадто складними для багатьох пристроїв з обмеженими ресурсами. Це підкреслює необхідність дослідження та пошукуефективних рішень, які відповідають ресурсним можливостям пристроїв IoT і дозволять забезпечити надійне шифрування та автентифікацію. Представлено гібридний алгоритм автентифікації RHAA (RSA and Hamming Code-based Authentication Algorithm), розроблений для забезпечення конфіденційності, цілісності та автентифікації даних у мережах IoT з великою кількістю вузлів. Особливістю запропонованого алгоритму є поєднання асиметричного шифрування на основі RSA з використанням корегуючих властивостей коду Хеммінга в скінченних полях. Такий підхід гарантує автентифікацію пристроїв, які беруть участь у процесі обміну інформацією, за рахунок централізованої генерації та управління ключами. Захист даних від несанкціонованого доступу досягається повторним шифруванням інформації на кожному з вузлів. Цілісність даних забезпечується шляхом виявлення та виправлення помилок під час кожного з етапів передачі. Таке рішення підвищує рівень безпеки даних в мережах IoT , знижуючи ризики витоку чи втрати інформації. У роботі наведено приклад реалізації алгоритму RHAA із застосуванням реальних значень ключів та показано, як система реагує на появу помилок і виконує їх корекцію. Запропонований алгоритм, оптимізований для обмежених пристроїв, може бути використаний для покращення захисту даних у мережах IoT.

Ключові слова: автентифікація, мережі IoT, алгоритм RSA, код Хеммінга в скінчених полях, шифрування, цілісність даних, виявлення та корекція помилок, конфіденційність, безпека передачі даних.

DOI 10.15276/imms.v14.no3.134
UDK: 681.518

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 134-145

A METHOD FOR IMPROVING THE QUALITY OF IMAGE ANNOTATION IN SEMANTIC MONITORING GIS OF BUSINESS PROCESSES

R.M. Pasichnyk, L.V. Babala, M.V. Machuliak

West Ukrainian National University
11, Lvivska Str. Ternopil, 46009, Ukraine
Emails: Roman.pasichnyk@gmail.com, Ludaduma7@gmail.com, Mvmach9@gmail.com

This article addresses the pressing issue of automating the image labeling process for computer vision systems in agriculture. The authors investigate methods for creating image datasets and configuring parameters for image classification models using neural networks based on the TensorFlow framework. The scientific significance of the work lies in developing new approaches to automated collection of thematic image collections and formalizing the methodology for parametric training of classification models. The practical value of the research is expressed in improving the efficiency of the image labeling process for geoinformation systems in the agricultural sector. The research methodology includes analyzing existing approaches to image labeling, developing an algorithm for automated formation of thematic image collections, formalizing a method for parametric training of the classification model, and experimental verification of the proposed approaches. Main results of the work: 1. An algorithm for automated formation of thematic image collections has been developed. 2. A method for parametric training of the image classification model using the TensorFlow framework has been formalized. 3. The dependence of classification accuracy on the size of the training sample and image augmentation parameters has been experimentally established. The study showed that with optimal selection of augmentation parameters and using 48 images per label in the training sample, it is possible to reduce the classification error to an acceptable level of 8%. The work makes a significant contribution to the development of automated image processing methods for agricultural geoinformation systems. The practical significance of the results lies in improving the efficiency of monitoring and management processes in the agricultural sector.
Keywords: computer vision, image labeling, neural networks, TensorFlow, geoinformation systems, agriculture, image classification.

Introduction. Modern agriculture faces the challenge of increasing production efficiency while reducing negative environmental impact. In this context, Geographic Information Systems (GIS) serve as a powerful tool revolutionizing the agricultural sector. GIS allows processing large volumes of geospatial data, creating detailed field maps, and analyzing various factors affecting plant growth, which in turn facilitates informed decision-making in agricultural production.

Particular attention in this study is given to the integration of computer vision methods into GIS for analyzing data obtained using drones. This technology allows for automatic identification and classification of objects in images, which significantly increases the efficiency of monitoring agricultural lands.

The article will examine modern approaches to implementing computer vision systems, particularly the application of convolutional neural networks, which provide hierarchical learning of features from basic (edges, corners) to complex (specific objects of interest).

Neural networks are an effective means of image classification. However, when using this apparatus, a number of difficulties arise. These lie in the great variety of network architectures and model training methods. The situation has been simplified with the introduction of the TensorFlow framework, where recommendations have been developed for certain subject areas regarding the choice of network architectures, methods for their rapid implementation, and effective training. However, researchers still face open questions about

selecting model parameters and automating the formation of image collections for model training. This work is dedicated to the study of these issues.

Fig. 1 shows the structure of the automated monitoring system for agricultural business processes:

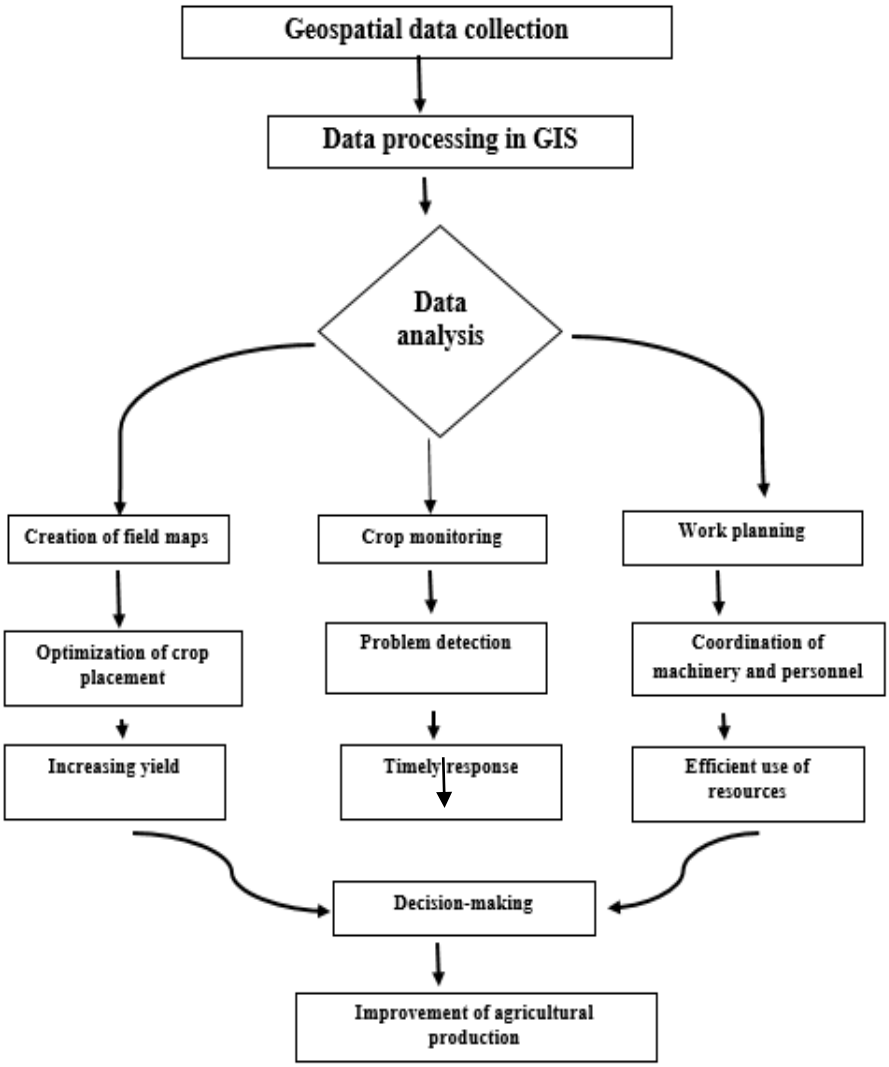


Fig. 1. Automated monitoring systems for agricultural business processes

Literature Review. Among the basic publications in the chosen research direction, we highlight reviews of approaches to image labeling problems [1] [2] [3] and approaches to building a system of semantic classes based on image segmentation [4], improving classification quality using cross-entropy similarity loss function [5], and building a semantic dictionary for image classification using deep learning methods [6].

In [1], basic approaches to forming semantic labels based on a keyword dictionary that solve the task of assessing the nature of images in the required aspect are analyzed. As these approaches are labor-intensive, various methods were used to reduce manual annotation costs. Three different types of image annotation are highlighted: free text annotation, keyword annotation, and ontology-based annotation.

Keyword annotation involves annotating images using a list of related keywords. There are two options for choosing keywords. In particular, this includes using arbitrary keywords, as well as using words from a predefined list. This information can be provided at two levels of specificity: firstly, a list of keywords related to the full image, listing what is depicted in the image. Secondly, image segmentation along with keywords associated with each segmentation area. Additionally, keywords describing the whole image can be provided. Often, segmentation

consists simply of a rectangular area drawn around the area of interest, or dividing the image into foreground and background pixels.

Ontological annotation is used when formed ontologies that cover a given subject area are available. For example, in the field of image description, ICONCLASS is a very detailed ontology for researching and documenting iconography images, designed to index or catalog the iconographic content of works of art, reproductions, literature, etc., and contains over 28,000 definitions organized in a hierarchical structure. Each definition is described by an alphanumeric code accompanied by a textual description.

For arbitrary text annotation, the user can use any combination of words or sentences. This facilitates annotation but complicates the use of annotation later for image search. Often this option is used in addition to keyword selection or ontology. Any concepts that cannot be adequately described using keyword selection are simply added in free-form description.

In [2], the search process in a system of medical articles is investigated. The developed labeled image dictionary is considered as the base vocabulary, images from which are compared with images from the analyzed document. Two image matching methods have been developed: one based on image intensity projections on coordinate axes, and the other on normalized cross-correlation. If image similarity thresholds are not reached, the analyzed image is skipped.

In [3], it is noted that to ease the load of manually labeling a large number of images, certain parts of the process can be automated where a computer vision algorithm performs preliminary annotation, and then a human user reviews and corrects the proposed labels. Accordingly, the human's role changes to supervision with the sole tasks of filtering, selection, and updating.

In the development of automated labeling, the methodology of interactive labeling is used, which is part of the "human-in-the-loop" methodology. Its goal is to reduce the limitations of fully automated labeling through purposeful interaction with the user. The method allows the user to label an initial batch of examples, trains a model, and then constantly asks the user for corrections. A developed strategy of active user feedback is used, which minimizes errors in subsequent labeling iterations and maximizes the expected information gain.

Works [4]-[6] are devoted to image labeling using semantic classes. In publication [4], the main idea is to identify areas that provide stable classification using an entropy measure. Minimizing entropy for different image segments gives its representation as a region adjacency graph. For each test image, generated contextual information is represented by a co-existence matrix.

In [5], the average similarity between the prediction and given labels is considered as a measure of semantic similarity in classification. With this type of evaluation, analysis of the loss function, which includes both cross-entropy similarity loss and deep feature loss, can improve the semantic similarity between the prediction and actual labels. It is shown that by analyzing the loss function, a model that produces more accurate predictions can be obtained. It is shown that cross-entropy similarity loss improves superclass similarity. It is proved that the average vocabulary similarity, which is a more accurate indicator, is also improved.

In [6], it is shown that although objects in the foreground of single-label images are one-level, this assumption is usually incorrect for multi-label images. Moreover, the different composition and interaction between objects in multi-label images also increases the informativeness of classifying such images. It is noted that multi-label image classification is more practical and complex than single-label image classification.

The paper presents a new end-to-end approach to multi-label image classification called Deep Semantic Dictionary Learning (DSDL), which considers the problem of multi-label image classification as a dictionary learning task. It uses an autoencoder to generate a semantic dictionary aligned with visual space, with class-level semantics. Unlike traditional approaches to multi-label image classification, DSDL not only uses correlations between label and vision spaces but also aligns relationships between label, semantic, and vision spaces.

Thus, some basic approaches to image labeling have been considered. Based on their generalization, we will form an approach to the announced automatic labeling within the framework of a geographic information system.

Semantic Label System. From the analysis of literature sources, it can be established that the basic methods of automatic labeling are based on autocorrelation image comparison or using artificial neural networks. In our view, neural networks are a more flexible tool that well supports the adaptation of the approach to detected deviations in semantic classification. Therefore, we prefer image labeling methods based on neural networks.

In particular, deep neural networks, especially convolutional neural networks (CNN), have achieved significant success in this field. Machine learning frameworks greatly simplify the process of creating and training neural networks. They provide ready-made tools, optimizations, and interfaces for interacting with data and models. In this regard, TensorFlow is one of the most popular machine learning frameworks developed by Google. It is particularly well-suited for working with deep neural networks, including convolutional neural networks. This involves the following main steps of using TensorFlow for image classification: data preparation, model formation, training, validation, and testing.

Storing trained models and quickly using them at the right moment with linking to corresponding geographical locations requires the development of a specialized geographic information system. Let's outline its basic structures that emerge from the types of information we plan to store.

From this perspective, we present the information models Im of the geographic information system being created as follows:

$$Im = \langle Bt, Ls, Gl, Md, Trs, \ Tss, Mit \rangle \quad (1)$$

where Bt - is the type of business process, Ls - is the image labeling system, Gl - is the geographic localization of the business Md - process, Trs - is the description of the method for building the image classification model, Tss - is the set of images for training, Mit - is the set of images for testing, are the types of images that the model classifies incorrectly.

$$Bt = \langle IdBt, \ NmBt \rangle \quad (2)$$

where, $IdBt$ - is the business process identifier, $NmBt$ - is the name of the business process.

$$Ls = \langle IdLi, \ NmLi, \Pr\_IdLi, IdBt \rangle \quad (3)$$

where, $IdLi$ - is the labeling element identifier, $NmLi$ - is the name of the labeling element, $\Pr\_IdLi$ - is a reference to the parent element of the hierarchy (identifier of the corresponding labeling element).

$$Gl = \langle IdLoc, \ NmLoc \rangle \quad (4)$$

where, $IdLoc$ - is the location identifier, $NmLoc$ - is the name of the location.

$$Md = \langle IdMd, \ TxtMd, \ IdBt \rangle \quad (5)$$

where, $IdMd$ - is the method identifier, $TxtMd$ - is the textual description of the method.

$$Trs = \langle IdTrs, TrPath, TrVol, Df, IdLi, IdBt \rangle \quad (6)$$

where, $IdTrs$ - is the identifier of the training image set, $TrPath$ - is the path to the training image set, $TrVol$ - is the volume of images in the training image set, Df - is the date and time of the image set formation.

$$Tss = \langle IdTss, TsPath, TsVol, Df, IdLi, IdBt \rangle \quad (7)$$

where, $IdTss$ - is the identifier of the testing image set, $TsPath$ - is the path to the training image set, $TsVol$ - is the volume of images in the testing image set.

$$Mit = \langle IdMit, IdLoc, IdLi, IdBt \rangle \quad (8)$$

where, $IdMit$ - is the identifier of the types of images that the model classifies incorrectly.

Method of Image Set Formation. After building structures for storing information, let's consider preparing a collection of images for training the neural network. We will start by building a list for labeling images from the subject area, entering it into the Ls structure. To build an image labeler for this subject area, we form a collection of images that will contain sets of images for each element of the constructed list. In the case of drones availability, the

image collection can be built from observations of objects that need to be labeled. If the system is just being formed, we create the image collection by extracting them or scraping them from the Web network.

Image scraping is the process of automatically collecting images from web pages. The Python library Beautiful Soup is often used for image extraction. It is characterized by ease of use, can work with various HTML and XML formats, and provides a wide range of methods for searching, navigating, and manipulating DOM elements.

To use the Beautiful Soup library, it is necessary to specify a Web page that contains suitable images for extraction. It can be selected by analyzing the output of a Web - search engine for a term query in the $NmLi$ image category.

Presenting the algorithm for scraping images from a multi-page structure in the form of certain stages. In particular, in the first stage, we set the tool, source, and parameters of scraping. In the second stage, we load the contents of the selected number of pages from a certain source and select the contents of image tags for a certain class from it. In the third stage, we select unique links to images from the selected tag content. And finally, in the fourth stage, we record images in a specified folder with corresponding names. Let's detail the implementation of these stages using the following steps and operators.

1. We form ***driver*** - the constructor of the *webdriver.Chrome()* class from the ***selenium*** library, which initializes a new instance of the web driver for the Chrome browser:

$$driver = webdriver.Chrome() \quad (9)$$

2. We fix ***url*** - a link to the site chosen for downloading and the ***npages*** we aim to download.

3. We fix ***class*** - the *css* class of image tags that contain links to the needed images

4. We organize a loop for the number of pages to download:

$$for\ page\ in\ range(page1, npage + 1) \quad (10)$$

5. We form ***url_page*** - a link to the next page:

$$url\_page = url + "?page = " + str(page) \quad (11)$$

6. Getting the contents of the next page:

$$driver.get(url\_page) \quad (12)$$

7. We analyze the contents of the page using an HTML parser and build a DOM - document object model:

$$content = driver.page\_source \quad (13)$$

$$soup = BeautifulSoup(content, "html.parser") \quad (14)$$

8. Loop through the elements of the image tag (IMG) with the specified class:

$$for\ image\ in\ soup.findAll("img", \{"class" : class\}) \quad (15)$$

9. Selecting a link from an element if it has not been used yet:

$$if\ image['src']\ not\ in\ results: \quad (16)$$

$$results.append(image['src']) \quad (17)$$

10. Loop through elements from downloaded images:

$$for\ b\ in\ results: \quad (18)$$

11. Getting an image by link:

$$image\_content = requests.get(b).content \quad (19)$$

$$image\_file = io.BytesIO(image\_content) \quad (20)$$

$$image = Image.open(image\_file).convert("RGB") \quad (21)$$

12. Writing the image to a specified folder with a numerical name in the order of download:

$$file\_path = Path("tractors\_plow2", str(numb).zfill(length) + ".png") \quad (22)$$

$$image.save(file\_path, "PNG", quality = 80) \quad (23)$$

$$numb = numb + 1 \quad (24)$$

Method Training Model Classification Image. We plan image scraping to have sufficient images for forming training and test sets for each marker NmLi of a specific business process NmBt. Thus, we create folders for individual business processes NmBt with subdirectories for text markers NmLi within them. For each marker, we form image directories for training Tr and testing Ts with corresponding volumes TrVol and TsVol. The number of such directories can be expanded with a link to the date and time of formation Df. Thus, paths to image directories for training Pitr(NmBt,NmLi,Df) and testing Pits(NmBt,NmLi,Df) are formed, which are recorded in the information system:

$$TrPath = Pitr(NmBt, NmLi, Df), \tag{25}$$

$$TsPath = Pits(NmBt, NmLi, Df). \tag{26}$$

After forming or replenishing sets of image directories, it is necessary to build and train the model according to a specific algorithm, which can be presented as the following sequence of stages. In the first stage, we set the values of the main algorithm parameters and form a dataset for model training. In the second stage, we create data processing pipelines that shuffle elements in the training and validation datasets. In the third stage, we create a neural network model for classification with random image transformations. In the fourth stage, we set the parameters of the method that regulates the learning process, carry out the learning process, and save the resulting model. The stages are implemented through the following steps:

1. Set the number of epochs, i.e., cycles during each of which the model uses each data sample once, and the extracted_dir directory containing photos for training, the fraction of data vs that will be allocated for the validation set

$$extracted\_dir = TrPath(NmBt, NmLi, Df) \tag{27}$$

2. Create a path object that represents the path to the directory from which images will be obtained

$$data\_dir = pathlib.Path(extracted\_dir) \tag{28}$$

3. Create a dataset of images from the directory containing photos for training and validation

$$train\_ds = tf.keras.utils.image\_dataset\_from\_directory(data\_dir, validation\_split = vs, subset = "training", seed = 123, image\_size = (img\_height, img\_width), batch\_size = batch\_size) \tag{29}$$

$$val\_ds = tf.keras.utils.image\_dataset\_from\_directory(data\_dir, validation\_split = vs, subset = "validation", seed = 123, image\_size = (img\_height, img\_width), batch\_size = batch\_size) \tag{30}$$

where the function $tf.keras.utils.image\_dataset\_from\_directory$ expects images to be organized in subdirectories, each corresponding to one class. Subdirectory names will be used as labels for images. The function recursively traverses the specified directory, loads all images and converts them into tensors that can be used for neural network training. All loaded images and their labels are combined into a *Dataset* object that can be used for iteration and feeding data into the model;

$validation\_split$ determines the fraction of data vs that will be allocated for the validation set;

$subset$ specifies which part of the data to return: training or validation;

$seed$ is used to set the initial value of the random number generator. If we set the same seed value for different runs, the data will be divided into training and validation sets in the same way;

$batch\_size$ defines the size of the data batch, i.e., a subset of data that is fed into the model simultaneously for gradient computation and weight updates. A larger batch size usually allows using larger mini-batch sizes, which can speed up training.

4. Increase the dimension of each element in the $train\_ds$ dataset. If previously each element was a pair (image, label), now it will become a triple (image, label, two identical integers).

$$counter = tf.data.Dataset.counter() \tag{31}$$

$$train\_ds = tf.data.Dataset.zip((train\_ds, (counter, counter))) \quad (32)$$

where $tf.data.Dataset.counter()$ creates simple datasets for testing or debugging models, used to create an infinite dataset where each element is a sequential integer starting from 0;

$tf.data.Dataset.zip()$ combines elements from two or more datasets into one new dataset. Elements from corresponding positions in the original datasets will be joined into tuples. A new dataset $train\_ds$ is created, which contains tuples of three elements: an image from the original train_ds set, the first integer from the counter set, and the second integer from the counter set. These additional integers can be used as indices for elements in the batch for learning algorithms.

5. We create data processing pipelines that shuffle elements in the $train\_ds$ and val_ds datasets, apply augmentation to increase data diversity for the training set, combine data into batches for efficient learning, and prefetch batches in advance to minimize model downtime

$$\begin{aligned} train\_ds = (\ & train\_ds \\ & .shuffle(1000) \\ & .map(augment) \\ & .batch(batch\_size) \\ & .prefetch(tf.data.AUTOTUNE)) \end{aligned} \quad (33)$$

$$\begin{aligned} val\_ds = (\ & val\_ds \\ & .map(resize\_and\_rescale, num\_parallel\_calls = tf.data.AUTOTUNE) \\ & .batch(batch\_size)\ ) \\ & .prefetch(tf.data.AUTOTUNE)) \end{aligned} \quad (34)$$

6. We describe a sequence of transformations for images that will be applied during model training, which will randomly change images in each training epoch

$$\begin{aligned} data\_augmentation = & keras.Sequential([\\ & layers.RandomFlip("horizontal", \\ & input\_shape = (img\_height, \text{img\_width}, 3)) \\ & layers.RandomRotation(alf1), \\ & layers.RandomZoom(alf2),]) \ ) \end{aligned} \quad (35)$$

There $keras.Sequential()$ creates a sequential model where each layer is applied to the output of the previous one;

$layers.RandomFlip("horizontal")$ randomly flips the image horizontally;

$input\_shape$ indicates the shape of input images (height, width, number of channels).

$layers.RandomRotation(alf1)$ randomly rotates the image by an angle from -$alf1$ to $alf1$ degrees.

$layers.RandomZoom(alf2)$ randomly increases or decreases the image scale by a value from –$alf2\%$ to $alf2\%$.

7. We create a sequential neural network model for image classification, consisting of several layers

$$\begin{aligned} model = Sequential([\\ & data\_augmentation, \\ & layers.Rescaling(1./255), \\ & layers.Conv2D(16, 3, padding='same', activation='relu'), \\ & layers.MaxPooling2D(), \\ & layers.Conv2D(32, 3, padding='same', activation='relu'), \\ & layers.MaxPooling2D(), \\ & layers.Conv2D(64, 3, padding='same', activation='relu'), \\ & layers.MaxPooling2D(), \\ & layers.Dropout(0.2), \\ & layers.Flatten(), \\ & layers.Dense(128, activation='relu'), \\ & layers.Dense(num\_classes) \quad]) \end{aligned} \quad (36)$$

There $keras.Sequential()$ creates a sequential model where each layer is applied to the output of the previous one;

$data\_augmentation$ - this layer (which we've already discussed) applies random transformations to images for data augmentation;

$layers.Rescaling(1./255)$ - normalizes pixel values of images to the range 0-1. This is important for most neural networks;

$layers.Conv2D(16, 3, padding = 'same', activation = 'relu')$ - applies a two-dimensional convolution with 16 filters of size 3x3;

$padding = 'same'$ - ensures that the output image size remains the same as the input;

$activation = 'relu'$ - applies the ReLU (Rectified Linear Unit) activation function to the output values;

$layers.MaxPooling2D()$ - the model contains two more convolution and pooling layers with a larger number of filters, allowing the model to extract more complex features from images;

$layers.Dropout(beta)$ - randomly turns off a fraction beta of neurons in this layer during training. This helps prevent overfitting;

$layers.Flatten()$ - transforms a three-dimensional tensor (height, width, channels) into a one-dimensional vector;

$layers.Dense(128, activation = 'relu')$ - applies a connected layer with 128 neurons and ReLU activation, allowing the model to extract abstract features;

$layers.Dense(num\_classes)$ - applies the last connected layer with the number of neurons equal to the number of classes. The output of this layer will be used for image classification.

8. Setting the optimizer, loss function, and metrics for model training

model.compile(optimizer='adam',
loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True), (37)
metrics=['accuracy'])

There $optimizer = 'adam'$ - a method that regulates the learning process by changing the weights of the neural network to minimize the loss function;

$tf.keras.losses.SparseCategoricalCrossentropy$ - a loss function for classification tasks that measures how well the model predicts results on training data;

$metrics = ['accuracy']$ - the simplest metric that shows the percentage of correctly classified samples, which allows evaluating the quality of the model during training and validation.

9. Initiate the neural network training process

history = model.fit(train_ds, validation_data=val_ds, epochs=epochs) (38)

with training sample $train\_ds$, validation sample val_ds, and the number of epochs of training sample passes during one iteration.

10. Save the result of model training in the file mt

$$model.save(mt) \quad (39)$$

Thus, IC $(data\_dir, vs, alf1, alf2, beta)$ (formalizes the process of building and training an image recognition and classification model in the *Tensorflow* package, the main parameters of which are the set of training image $data\_dir$ directories, the fraction of images vs that will be allocated for the validation set, the limits $alf1$ of random image rotation, the limits $alf2$ of random image scaling, the fraction beta of randomly turning off network neurons during training.

Experimental Results. We experimentally assess the quality of the built image classification model and its adaptation to the proposed data. In the first stage, if necessary, we adapt the list of semantic labels to the set of images selected for training. If the quality of image recognition for a certain semantic label is sufficiently low, this label is subordinated to the one closest to it semantically. In the second stage, we optimize the accuracy of the image labeling model by

selecting the parameters of the model training algorithm and, if possible, adjusting the volume of the training set for image classification.

Let's consider the process of labeling images of field cultivation in an agricultural enterprise. Let the system of semantic labels be defined by the following one-level list:

$$NmLi = [tractors\_plow, tractors\_cultivation, tractors\_seeding, tractors\_watering\_plants, tractor\_cutting\_grass, combining] \quad (40)$$

In the first stage, we download classified images from Web resources, for example, from the site *https://www.istockphoto.com.* First, using a thematic query, we find Web sites of thematic images with the ability to select photos based on queries that correlate with semantic labels. We form such a set of images using the described method of forming an image set, in particular using the css class "yGh0CfFS4AMLWjEE9W7v". The volume of downloaded collections should be selected as significant; in this example, volumes of 1200 photos were set. Unfortunately, when viewing the collapsed images, it turns out that not all of them correspond to the queries based on which they were selected by the Web site. Therefore, automatic downloading should be supplemented with visual manual filtering. This is a labor-intensive process, so we aimed to form collections for semantic labels in volumes of 12, 24, or 48 specimens, calculating that these sets will be divided into training and validation parts. In the next stage, we train the corresponding neural networks according to the described algorithm. Randomly selected image samples used for training are shown in the following figure.

Fig. 2. Sample images used for training

The quality of training was evaluated using test samples that were not used for training. They were formed from the set of downloaded images with a volume of 5 and 10 images per semantic label.

It was established that the classification accuracy primarily depends on the size of the training sample as well as on the parameters of image collection augmentation during training. The main results of the calculations are presented in the following table.

Table 1.

Image Classification Accuracy

n	***TrVol***	***|NmLi|***	***α1***	**α2**	**ε(%)**
1	12	6	0.3	0.3	43
2	12	6	0.4	0.4	63
3	12	6	0.5	0.5	58
4	24	5	0.2	0.2	26
5	24	5	0.3	0.3	18
6	24	5	0.4	0.4	38
7	48	5	0.2	0.2	32
8	48	5	0.1	0.1	18
9	48	5	0.05	0.05	8

In the table, ***TrVol*** denotes the size of the training sample, $|NmLi|$ - the number of semantic labels in the model, $\alpha 1$, $\alpha 2$ - parameters of image collection augmentation during training, ε - relative error obtained when testing the model. The presented results indicate a low level of classification accuracy with small training sample sizes. Poor differentiation of the semantic label $tractors\_seeding$ was also recorded, which had to be combined with the label, $tractors\_cultivation$ leading to a reduction in their number from 6 to 5. With a successful selection of augmentation parameters and using 48 images per label in the training set, it was possible to reduce the classification error to an acceptable level of 8%. The dynamics of errors of the image classification model during its parameter selection are shown in the following figure. Only one level of recorded errors can be considered acceptable.

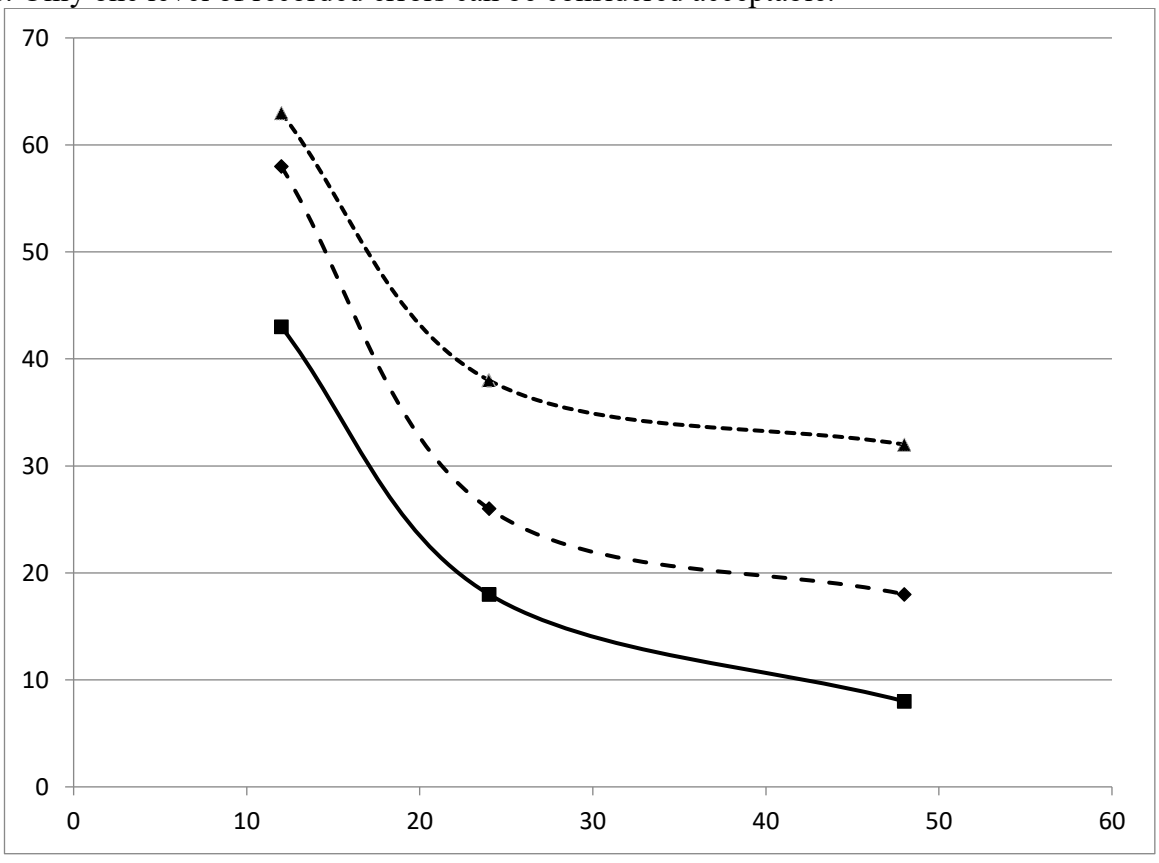


Fig.3. Dynamics of errors of the image classification model during its tuning

Conclusions. The paper investigates the issue of automating the formation of image collections and the formation of a methodology for tuning the parameters of the image classification model

using neural networks with the TensorFlow framework. An algorithm for automated formation of a thematic image collection is proposed. A method of parametric training of the thematic image classification model using the TensorFlow framework is also formalized. Experimental studies have confirmed the effectiveness of the proposed approaches.

References

1. Hanbury A. A survey of methods for image annotation. *Journal of Visual Languages and Computing*. 2008. No.19. P. 617–627.
2. Chachra S.K., Xue Z, Antani S., Demner-Fushman D., Thoma G.R. Extraction and Labeling High-resolution Images from PDF Documents. Lister Hill National Center for Biomedical Communications Bethesda, MD: U. S. National Library of Medicine,2023. 20894
3. Sager Ch., Janiesch Ch., Zschech P. A survey of image labelling for computer vision applications. *Journal of Business Analytics*. 2021. V.4. No.2, P. 91-110, DOI: 10.1080/2573234X.2021.1908861
4. Kluckner S., Mauthner T., Roth P.M., Bischof H. Semantic Image Classification Using Consistent Regions and Individual Context. URL: https://www.researchgate.net/publication/ 221259742_Semantic_Image_Classification_Using_Consistent_Regions_and_Individual _Context
5. Yu.Y. Learning Semantics of Classes in Image Classifcation. URL: https://www.zora.uzh.ch/id/eprint/259312/1/yu2023master.pdf
6. Zhou F., Huang S., Xing Y. Deep Semantic Dictionary Learning for Multi-label Image Classification. URL: https://cdn.aaai.org/ojs/16472/16472-13-19966-1-2-20210518.pdf

МЕТОД ПІДВИЩЕННЯ ЯКОСТІ РОЗМІТКИ ЗОБРАЖЕНЬ СЕМАНТИЧНОГО МОНІТОРИНГУ БІЗНЕС-ПРОЦЕСІВ ГІС

Р.М. Пасічник, Л.В. Бабала, М.В. Мачуляк

Західноукраїнський національний університет
11, Львівська, м. Тернопіль, 46009, Україна
Emails: Roman.pasichnyk@gmail.com, Ludaduma7@gmail.com, Mvmach9@gmail.com

Стаття присвячена актуальній проблемі автоматизації процесу маркування зображень для систем комп'ютерного зору в сільському господарстві. Автори досліджують методи формування наборів зображень та налаштування параметрів моделей класифікації зображень з використанням нейронних мереж на базі фреймворку TensorFlow. Наукова значущість роботи полягає у розробці нових підходів до автоматизованого збору тематичних колекцій зображень та формалізації методики параметричного навчання моделей класифікації. Практична цінність дослідження виражається у підвищенні ефективності процесу маркування зображень для геоінформаційних систем у сільськогосподарській галузі. Методологія дослідження включає аналіз існуючих підходів до маркування зображень, розробку алгоритму автоматизованого формування тематичних колекцій зображень, формалізацію методу параметричного навчання моделі класифікації та експериментальну перевірку запропонованих підходів. Основні результати роботи: 1. Розроблено алгоритм автоматизованого формування тематичних колекцій зображень. 2. Формалізовано метод параметричного навчання моделі класифікації зображень з використанням фреймворку TensorFlow. 3. Експериментально встановлено залежність точності класифікації від розміру навчальної вибірки та параметрів аугментації зображень. Дослідження показало, що при оптимальному підборі параметрів аугментації та використанні 48 зображень на мітку в навчальній вибірці можливо знизити помилку класифікації до прийнятного рівня 8%. Робота вносить значний внесок у розвиток методів автоматизованої обробки зображень для сільськогосподарських геоінформаційних систем. Практичне значення результатів полягає у підвищенні ефективності процесів моніторингу та управління в аграрному секторі.

Ключові слова: комп'ютерний зір, маркування зображень, нейронні мережі, TensorFlow, геоінформаційні системи, сільське господарство, класифікація зображень.

DOI 10.15276/imms.v14.no3.146
UDC 004.424.5

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 146-153

EFFICIENCY OF SORTING ALGORITHMS IN TYPESCRIPT

O. G. Trofymenko[1], Yu. V. Prokop[2], A. I. Dyka[1], O. S. Karahuts[1]

[1] National University "Odesa Law Academy"
23, Fontans'ka doroga st., Odesa, 65009, Ukraine
Email: trofymenko@onua.edu.ua
[2] National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
Email: prokop.y.v@op.edu.ua

Since developers often need to organize data, choosing the fastest and most efficient sorting algorithm depending on the size and other properties of the data, as well as the programming language, is relevant. In some cases, processing the data directly in the browser is necessary due to the need for high data confidentiality. The growing popularity of TypeScript in web development over the past year makes it topical to study the effectiveness of various sorting algorithms in this language. This paper investigates the speed and performance of twelve sorting algorithms using the modern web development language TypeScript: Bubble, Selection, Insertion, Shell, Merge, Quick, TimSort, Smooth, Introspective, Gravity, Radix, and built-in. We compared the actual runtime of each algorithm for sets of pseudorandom integers from 1000 to 100,000,000 elements. Although the built-in sort() TS method is flexible and adapts to different situations, the study results show that it gives the best results and can only be a good choice on data up to 1000 items. The built-in method loses to Quick Sort, Introspective Sort, Timsort, and Merge Sort algorithms on larger arrays and may not be the best choice. Therefore, studying the efficiency and features of sorting algorithms is very relevant. The applied aspect of the study is to find out which algorithm, when implemented in TypeScript, will optimally sort an array of pseudorandom numbers depending on its size and other properties. The results can help effectively choose one algorithm under certain conditions and data. The study confirmed that each sorting algorithm we considered has advantages and disadvantages. The choice of an appropriate sorting algorithm for a particular development task depends mainly on the size and specific characteristics of the data and the programming language. The choice is also influenced by the desired level of sorting efficiency and the stability requirements of the algorithm.

Keywords: sorting algorithms, efficient algorithms, running time, performance, sorting, testing, TypeScript.

Introduction. Software developers frequently use sorting algorithms to organize numeric and textual data. Effective sorting is also essential for optimizing the implementation of other algorithms, such as search and data merging algorithms, which require sorted lists to work correctly. All modern programming languages have built-in sorting methods. However, the algorithms used in these methods are only sometimes the most efficient. Therefore, the problem of choosing the optimal one from a wide range of sorting algorithms is relevant.

Sorting efficiency depends not only on the algorithm but also on the programming language in which it is implemented and on the properties of the data being sorted [1]. Therefore, no optimal sorting algorithm exists for all programming languages and data sets. Studying the effectiveness of algorithms for a particular programming language is relevant.

Analysis of research and publications. Many works have been devoted to studying sorting algorithms implemented in different programming languages. For example, using the Python programming language, the paper [2] investigated the performance of five sorting algorithms (Quick, Heap, Merge, Introspective, and Radix). In [3] and [4], the performance of two and three sorting algorithms in Java was compared, respectively. The study [5] compares the performance of algorithms for sorting sets of pseudorandom numbers from 10,000 to 100,000 elements using three languages: Python, C++, and Java. The paper [1] investigates the

efficiency of nine popular sorting algorithms in six programming languages: Python, C++, Java, JavaScript, PHP, and C#. The paper [6] studies the effectiveness of five popular sorting algorithms (Bubble, Selection, Insertion, Merge, and Quick) using C++ and Java for random number sets ranging from 10,000 to 50,000. The paper [7] explains the work of three less common sorting algorithms in TypeScript (Bloom, Shell, Heap) but does not compare the efficiency of these algorithms. The analysis of publications revealed, on the one hand, interest in the search for practical sorting algorithms in different languages and, on the other hand, the lack of studies on algorithms in TypeScript and JavaScript. Usually, data sorting is performed on the server side. However, in some cases, processing the data directly in the browser is necessary due to the need for high data confidentiality. The growing popularity of TypeScript in web development over the past year makes it relevant to study the effectiveness of various sorting algorithms in this language.

Research Objective. The main goal of the work is to compare different sorting algorithms implemented in TypeScript. The applied aspect of the study is to identify how the implementation in this language affects the algorithm's execution time for arrays of pseudorandom numbers of different sizes.

Choosing a programming language for research. The reason for selecting TypeScript (TS) for this study is its rapidly growing popularity in web development. TS is an add-on for JavaScript (JS), as TypeScript code needs to be compiled into JS to run in a browser. On the other hand, this makes TS compatible with any browser and JS engine and ensures its compatibility with existing JS libraries and frameworks. The official website (https://www.typescriptlang.org/) states, "TypeScript is a strongly typed programming language that builds on JavaScript, giving you better tooling at any scale". Most programming language rankings consider TS and JS to be different languages, with TS rapidly catching up with JS in popularity. Thus, in the DOU 2024 ranking of programming languages, TypeScript (15%) came in second place after JavaScript (JS) (15.3%) [8], almost equalizing its leadership position. At the same time, over the past year, JS has lost 2.8% of users (professionals) who use the language for development. TypeScript has risen in the ranking by 10.7%, becoming the language of the year in popularity growth. Front-end developers often prefer TypeScript. The share of TS supporters in the front-end development has increased by 15.3% over the past year. As for the back-end and full-stack areas, fans have also increased significantly over the year, although not as rapidly – by 2.5% and 9.4%, respectively. TypeScript has become more widely used both for desktop applications (up 3.9%) and for mobile application development, both cross-platform (up 7.1%) and operating system-specific: mobile Android (up 1.9%), mobile iOS (up 3.5%), and embedded (up 0.4%).

Comparative analysis of the speed and performance of sorting algorithms. In the practical part of the study, we implemented 11 sorting algorithms using the TypeScript language and the Node.js platform: Bubble, Selection, Insertion, Shell, Merge, Quick, Timsort, Smooth, Introspective, Gravity, and Radix. We compared the actual runtime of each algorithm for sets of pseudorandom integers from 1000 to 100,000,000 elements in increments of 10n (n is the number of bits in the number). We calculated the time as the arithmetic mean of five measurements of the algorithm's running time, as this approach is commonly used in research [1] – [6]. The units of measurement are milliseconds (ms). The hardware and software components of the study are as follows: MacBook Pro 13" laptop based on the Apple M2 processor and 8 GB of RAM; TS – 5.1.6, Node – v18.13.0.

At the first stage of analyzing the performance of these sorting algorithms, we excluded from the comparison the algorithms that showed low performance. For example, the Bubble Sort for 1,000,000 items took 1,953,523 ms, i.e. 32.5 minutes. The Selection Sort algorithm showed 3.6 times better results than the bubble algorithm (536,735 ms) but was also very slow (almost 9 minutes). The Insertion Sort showed results (264,248 ms, i.e., 4.5 minutes) twice as good as the Selection Sort. However, all these three algorithms are inefficient, especially for

large datasets, compared to the results of much more efficient algorithms. The time complexity of all three algorithms is $O(n^2)$, and the space complexity is O(1).

Gravity, or Bead sorting, is a relatively new and not-so-common sorting algorithm. We considered it in our study because the algorithm's complexity can theoretically reach O(n) for sorting natural numbers [9]. Practical implementation showed that sorting 100,000 items takes 3918 ms, which is worse than the performance of the Insertion Sort algorithm. Still, unlike this algorithm, the Gravity sort requires large memory consumption during $O(n^2)$ operation, so it is unsuitable for sorting large data sets. When sorting 1,000,000, the program crashes due to a lack of memory – a stack overflow (Fig. 1). Therefore, we had to remove this algorithm from the further comparison of the performance of the sorting algorithms.

```
Array(10) was sorted with Bead sort in 0.8835 ms.
Array(100) was sorted with Bead sort in 6.198834 ms.
Array(1000) was sorted with Bead sort in 9.420167 ms.
Array(10000) was sorted with Bead sort in 139.405083 ms.
Array(100000) was sorted with Bead sort in 3917.76275 ms.

<--- Last few GCs --->

[19227:0x140078000]    14076 ms: Mark-sweep (reduce) 2043.3 (2084.2) -> 2043.3 (2084.2) MB,
 requested
[19227:0x140078000]    15038 ms: Mark-sweep (reduce) 2043.3 (2084.2) -> 2043.3 (2084.2) MB,
 requested


<--- JS stacktrace --->

FATAL ERROR: CALL_AND_RETRY_LAST Allocation failed - JavaScript heap out of memory
 1: 0x104e7d4fc node::Abort() [/Users/klapeks/.nvm/versions/node/v18.16.0/bin/node]
 2: 0x104e7d6ec node::ModifyCodeGenerationFromStrings(v8::Local<v8::Context>, v8::Local<v8::
```

Fig. 1. The result of Gravity sorting.

The Shell Sort algorithm sorted 1,000,000 items in 243 ms and 100,000,000 items in 62 seconds, 1230 times faster than the Insertion Sort algorithm. The time complexity varies depending on the choice of gap sequence. It is typically $O(n^{1.5})$ or $O(n^2)$. And the space complexity is O(1). The Shell Sort algorithm shows better results on partially sorted data. Its efficiency decreases on datasets greater than 10^6 elements.

The Merge Sort was even faster: it sorted 1,000,000 items in 159 ms and 100,000,000 in 28 seconds. The time complexity of the Merge Sort algorithm is O(n log n), and the space complexity is O(n). This algorithm can be efficient for massive data sets of up to 10^9 elements. The Merge Sort is a stable algorithm, e.g., equal elements retain order when sorting.

The Quick Sort algorithm has a spatial complexity of log(n) and a time complexity of O(n log n) operations on average, and in the worst case, it makes $O(n^2)$ comparisons [10]. This algorithm is unstable. In practice, the Quick Sort showed the best results among the compared algorithms, even for large data sets. For example, it processed 1,000,000 items in 89 ms and 100,000,000 in 13.7 seconds.

The Timsort hybrid sorting algorithm combines Insertion and Merge sorting. Its time complexity is O(n log n), and its space complexity is O(1). The practical implementation of this algorithm with TS has shown better results than the Shell and Merge algorithms. The Timsort sorted 1,000,000 items in 124 ms and 100,000,000 in 24.5 s. This algorithm is stable and shows promising results on partially sorted data.

Smooth Sort is a variation of the Heap Sort algorithm proposed by E. Dijkstra. The advantage of Smooth Sort is that its performance approaches O(n) if the input data is partially ordered. In contrast, the performance of the Heap Sort is constant and does not depend on the state of the input data. Smooth sorting in TS was slower than Shell sort: it sorted 1,000,000 items in 274 ms and 100,000,000 in 101 s.

Radix Sort is a fast, stable algorithm for organizing data with a time complexity of O(n+k) (where n is the number of elements in the array; k is the number of characters in the alphabet;

for decimal numbers, k = 10) and a space complexity of O(n+k). There are as many ordering cycles as bits in the maximum element. The Radix Sort is suitable for sorting numeric data when the range of values is not too wide. The results of this algorithm in TS turned out to be slower than Shell's: it sorted 1,000,000 elements in 391 ms and 100,000,000 in 64 s.

Introspective Sort uses Quick Sort and switches to Heap Sort if the recursion depth exceeds some predefined level (e.g., the logarithm of the number of sorted items). This approach combines the advantages of both methods with a worst-case time complexity of O(n log n) and performance comparable to Quick Sort. The memory consumption during Introsort execution is O(n). In practice, Introspective Sort is faster than Merge Sort but slower than Quick Sort: it sorted 1,000,000 items in 139 ms and 100,000,000 in 21.8 seconds.

Using the built-in sort() method in TypeScript gave mixed results. For small arrays of up to 1000 elements, its results were the best among all twelve sorting algorithms compared. However, with an increase in the number of the array elements, its performance deteriorated: with 10,000 elements, the built-in sort was twice as slow as the Quick Sort algorithm, and when sorting 1,000,000 elements, its performance was slower than the Shell algorithm, and ranked sixth among the twelve (Table 1). These results can be explained by the fact that the sort() method is reconfigurable and uses different sorting algorithms "under the hood": Quick, Heap, Merge, or other. Among them, the most commonly used algorithm in the JS sort() method is

Table 1.

Running time of TypeScript sorting algorithms for different array sizes (in ms)

Sorting Algorithm	*Number of elements*					
	10^3	10^4	10^5	10^6	10^7	10^8
Bubble	1,85	84,18	15739	1953523	–	–
Selection	1,84	58,48	5433	543963	–	–
Insertion	1,37	32,40	2647	299032	–	–
Gravity	8,50	146,0	3617	–	–	–
Smooth	0,93	8,32	34,1	274	5931	101523
Radix	0,42	4,94	33,1	391	3820	64110
Shell	1,45	2,24	18,5	243	3345	62264
Built-in	0,22	1,39	17,4	258	2310	31398
Merge	0,63	1,75	20,5	159	1881	28209
Timsort	0,54	4,51	15,1	124	1478	24510
Introspective	0,36	1,67	12,2	139	1621	21799
Quick	0,39	0,77	7,8	89	982	13742

The choice of algorithm for the sort() method can depend on various factors, such as the size of the array, the type of data, the optimization strategy of the sorting mechanism, and even the browser.

The built-in sort() method in JS is designed to handle various scenarios and ensure stable and efficient operation on different data types. To achieve this, it employs more sophisticated pivot selection strategies like the median of three through GetThirdIndex(). However, the call to this function requires additional computation to find the third index and compare values to select the pivot. This overhead can become noticeable on large arrays or with frequent calls. If the data is unordered or has a complex structure (e.g., objects instead of simple numbers), handling it through GetThirdIndex() might require more time for comparisons. Therefore, in some cases, especially with specific data types or distributions of values, this can slow down the sorting process compared to a more straightforward Quick Sort implementation.

As it turned out, the built-in sorting is not the best in speed and performance. TS, like JS, is a scripting programming language. Since TS is an add-on for JS, most JS libraries are compatible with TS. The mechanisms of these languages are usually implemented as part of

web browsers, server platforms, or standalone TS and JS runtimes [12]. Browsers have different JavaScript engines, which are a way of executing JavaScript code. That's why different browsers and platforms use different methods and strategies, including the sort() method. Over time, browsers themselves revise their approaches to using their engines to optimize and improve them, and therefore, the JavaScript toolkit is transforming. For example, Chrome uses the V8 engine, and Mozilla Firefox uses SpiderMonkey. Firefox uses Merge Sort but switches to Insertion Sort on small arrays [12]. Desktop Chrome and Safari on the V8 engine use Timsort and Quick Sort, but on iOS, they use Merge Sort. Since Node.js uses V8 (the engine from Chrome), it also uses primarily Quick Sort. Older Opera uses Merge Sort. Konqueror and Rekonq use a binary and red-black tree and their variations [13].

The choice of sorting algorithm is mainly guided by the Big O notation indicators [14]. This notation provides a general rule of thumb for the dependence of expected performance on algorithm scalability. Still, it cannot consider all the variable factors that affect the performance and speed of sorting algorithms.

Fig. 2 shows a graphical representation of the performance comparison of eight sorting algorithms listed in Table 1.

The implementation of the algorithms showed a fairly wide variation in speed and performance despite their identical profiles in the Big O notation. Sometimes, sorting algorithms with worse O-notations of time complexity showed better speed and performance than algorithms with worse Big O notations. For example, in the worst case, Quick Sort has a time complexity of O(n2), unlike many other algorithms with better notations, showing the highest speed and performance for arrays of 10,000 elements or more.

As shown above, four sorting algorithms (Bubble, Selection, Insertion, and Gravity) are unproductive for large arrays; thus, we excluded them from the comparison.

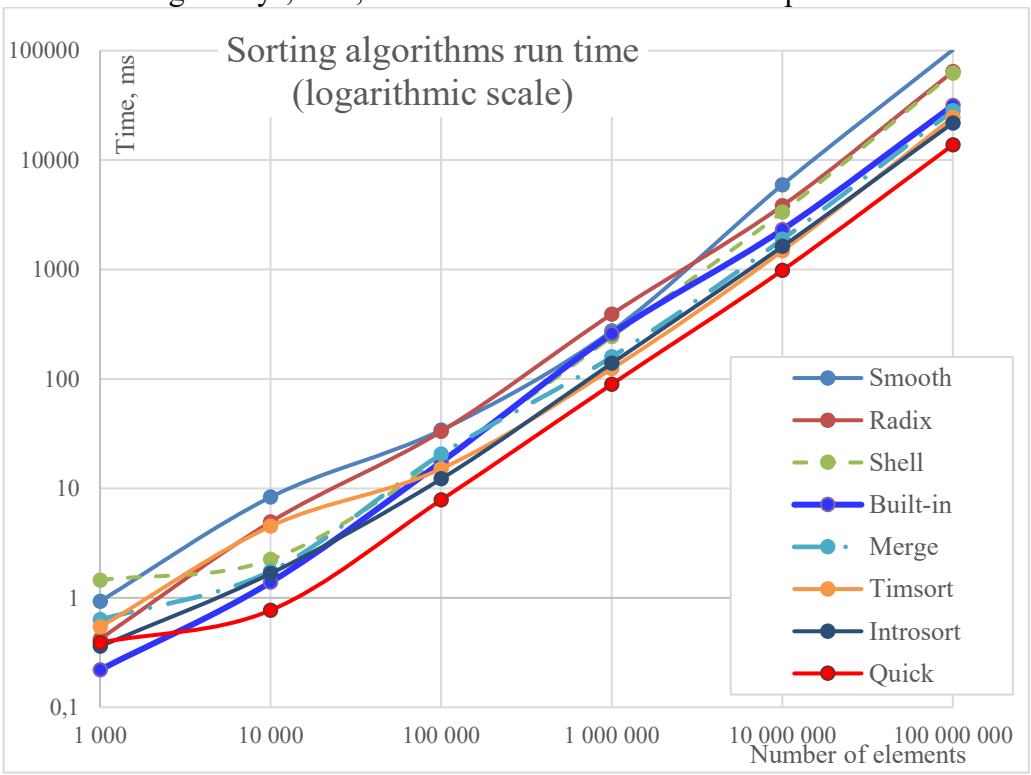


Fig. 2. Speed comparison of TypeScript sorting algorithms.

Comparing the speed (Fig. 2) and performance of the mentioned algorithms (Fig. 3), we found out that the size of the arrays significantly affects the work of different sorting algorithms.

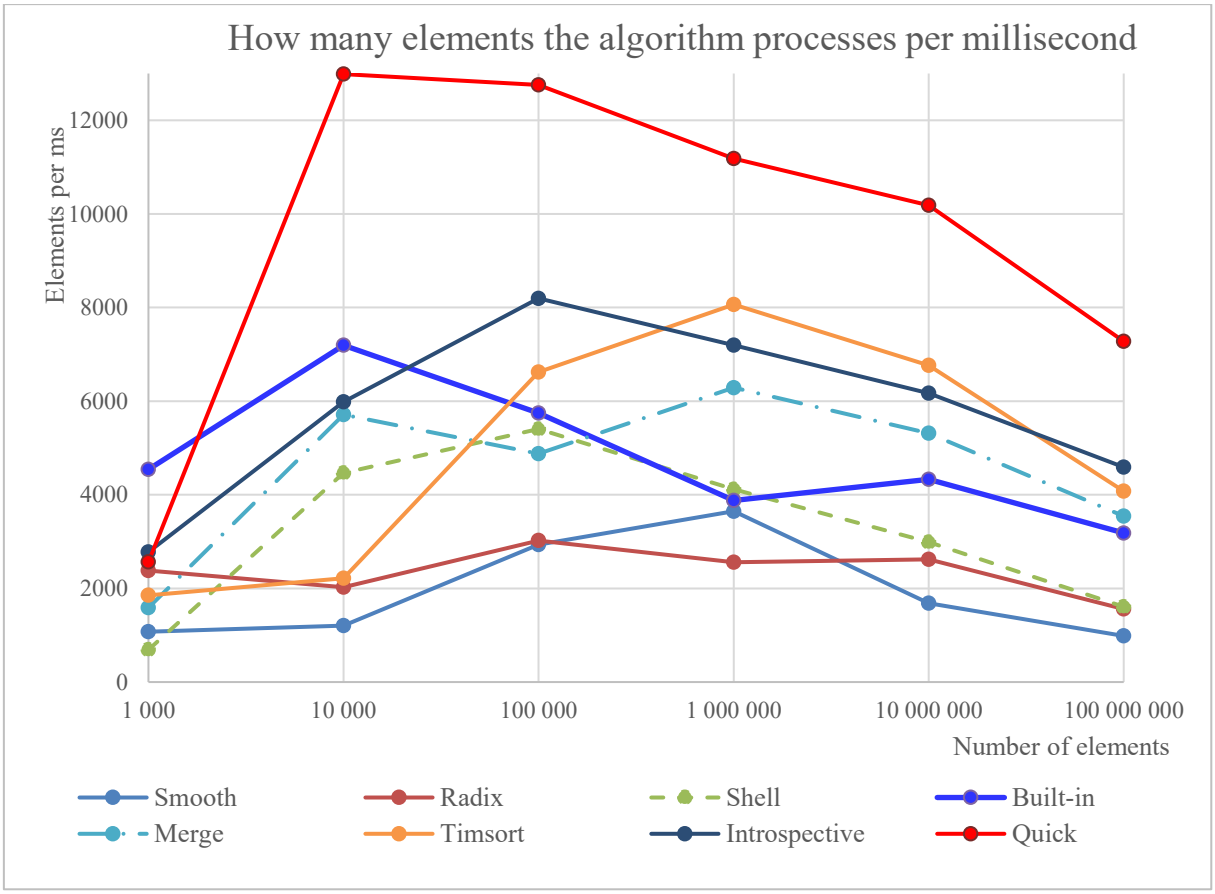


Fig. 3. Performance comparison of TypeScript sorting algorithms.

As we can see from the graphs, the built-in sorting method is only sometimes the best choice since there are algorithms with much better performance and productivity. For example, the built-in TS sort has the highest speed and performance for up to 1,000 item arrays. For arrays with more than 1,000 items, the Quick Sort algorithm showed the best speed and performance. The size of the array significantly influences the speed and performance of the Shell and Merge algorithms. For large array sizes, the Merge Sort is more efficient. The practical implementation of the Timsort algorithm by TS tools has shown better results than the Shell and Merge algorithms. Smooth and Radix sorting showed the worst performance for massive data sets among the eight sorting algorithms compared in Table 1. For arrays up to 1000 elements, Introsort, Timsort, Radix Sort, and Quick Sort have good performance. The built-in sort() method can be the best choice in this case.

Considering the results obtained, we can conclude that Quick Sort is one of the most efficient sorting algorithms for large datasets. It generally outperforms many other algorithms. However, there are certain cases and data structures where Quick Sort may not be the best choice. In such situations, different methods should be preferred. For small arrays of up to 50 elements, consider using Insertion Sort or Shell Sort; for up to 1000 elements, consider Introsort, Timsort, or built-in sort. For the already sorted or nearly sorted data, consider using Timsort or Merge Sort. If stability is critical or you want to preserve the order of equal elements when using secondary sorting criteria, stable algorithms like Merge Sort or Timsort can be the better choice. When sorting in environments with limited stack size or handling massive datasets that might cause deep recursion, consider using an iterative version of Quick Sort or another algorithm like Merge Sort.

Conclusions. Since developers frequently need to organize data, selecting a fast, efficient sorting algorithm is relevant. This paper investigates the speed and performance of twelve

sorting algorithms using the modern web development language TypeScript: Bubble, Selection, Insertion, Shell, Merge, Quick, TimSort, Smooth, Introspective, Gravity, Radix, and built-in.

Although the built-in sort() TS method is flexible and adapts to different situations, the study results show that it gives the best results and can only be a good choice on data up to 1000 items. The built-in method loses to Quick Sort, Introspective Sort, Timsort, and Merge Sort algorithms on larger arrays and may not be the best choice. Therefore, studying the efficiency and features of sorting algorithms is very relevant.

Bubble, Selection, Insertion, and Gravity algorithms showed significantly worse speed and performance, so we excluded them from consideration.

We analyzed the performance differences of the other eight sorting algorithms. The results can help effectively choose one algorithm under certain conditions and data. The study confirmed that each sorting algorithm we considered has advantages and disadvantages.

The choice of an appropriate sorting algorithm for a particular development task depends mainly on the size and specific characteristics of the data and the programming language. The choice is also influenced by the desired level of sorting efficiency and the stability requirements of the algorithm.

References

1. Трофименко О.Г., Прокоп Ю.В., Чепурна О.Є., Корнійчук М.М. Порівняння швидкодії алгоритмів сортування у різних мовах програмування. *Кібербезпека: освіта, наука, техніка*. 2023. № 1(21). С. 86-98. DOI: https://doi.org/10.28925/2663-4023.2023.21.8698
2. Marcellino M., Pratama D. W., Suntiarko S. S., Margi K. Comparative of Advanced Sorting Algorithms (Quick Sort, Heap Sort, Merge Sort, Intro Sort, Radix Sort) Based on Time and Memory Usage. *Proceedings 1st International Conference on Computer Science and Artificial Intelligence (ICCSAI'2021).* 2021. Vol. 1. P. 154-160. DOI: https://doi.org/10.1109/ICCSAI53272.2021.9609715.
3. Ali I., Lashari H., Keerio I., Maitlo A., Chhajro M., Malook M. Performance Comparison between Merge and Quick Sort Algorithms in Data Structure. *Proceedings International Journal of Advanced Computer Science and Applications*. 2018. Vol. 9. P. 192-195. DOI: https://doi.org/10.14569/IJACSA.2018.091127.
4. Rabiu A., Garba E., Baha B., Malgwi Y., Dauda M. Performance Comparison of three Sorting Algorithms Using Shared Data and Concurrency Mechanisms in Java. *Arid-zone Journal of Basic & Applied Research.* 2022. Vol. 1. P. 155-64. DOI: https://doi.org/10.55639/607fox.
5. Durrani O. K., Hayan S. Asymptotic performances of popular programming languages for popular sorting algorithms. *Semiconductor Optoelectronics*. 2023. Vol. 42. P. 149-169. URL: https://www.researchgate.net/publication/369196272_asymptotic_performances_of_popular_programming_languages_for_popular_sorting_algorithms.
6. Durrani O. K., Farooqi A. S., Chinmai A. G., Prasad K. S. Performances of Sorting Algorithms in Popular Programming Languages. *Proceedings International Conference on Smart Generation Computing, Communication and Networking (SMART GENCON'2022),* Bangalore, India. P. 1-7. DOI: https://doi.org/10.1109/smartgencon56628.2022.10084261.
7. Korzhenko V. Exploring Lesser-Known Sorting Algorithms in TypeScript. URL: https://medium.com/@vitaliykorzenkoua/exploring-lesser-known-sorting-algorithms-in-typescript-1c0a2ecff57
8. DOU. Ranking of programming languages 2024. URL: https://dou.ua/lenta/articles/language-rating-2024
9. Nagaraju N. A better implementation of bead-sort. URL: https://medium.com/@vini.the.pooh/a-better-implementation-of-bead-sort-7ca7352de036
10. Srivastava, P. Stable Sorting Algorithms. URL: https://www.baeldung.com/cs/stable-sorting-algorithms.

11. Riordan G. JavaScript Sort – How to Use the Sort Function in JS. 2023. URL: https://www.freecodecamp.org/news/how-does-the-javascript-sort-function-work/.
12. Three Common Sorting Algorithms with JavaScript. URL: https://blog.javascripttoday.com/blog/sorting-algorithms-with-javascript/.
13. Grzybek M. Ultimate guide to sorting in Javascript and Typescript. URL: https://dev.to/maciekgrzybek/ultimate-guide-to-sorting-in-javascript-and-typescript-4al9..
14. Big-O Cheat Sheet. URL: https://www.bigocheatsheet.com.

ЕФЕКТИВНІСТЬ АЛГОРИТМІВ СОРТУВАННЯ В TYPESCRIPT

О. Г. Трофименко[1], Ю. В. Прокоп[2], А. І. Дика, О. С. Карагуц

[1] Національний університет «Одеська юридична академія»
23, Фонтанська дорога, м. Одеса, 65000, Україна
Email: trofymenko@onua.edu.ua
[2] Національний університет «Одеська політехніка»
1, Шевченка пр., м. Одеса, 65044, Україна
Email: prokop.y.v@op.edu.ua

У статті порівняно дванадцять різних алгоритмів сортування, реалізованих у TypeScript. Оскільки розробникам часто потрібно впорядковувати дані, актуальним є вибір швидкого та ефективного алгоритму сортування в залежності від розміру та інших властивостей даних, а також мови програмування. Через потребу забезпечення високої конфіденційності даних іноді доводиться обробляти їх безпосередньо в браузері. Зростання популярності TypeScript у веброзробці робить актуальним вивчення ефективності різних алгоритмів сортування цією мовою. Прикладний аспект дослідження полягає у з'ясуванні алгоритму, реалізованого у TypeScript, який оптимально сортуватиме масив псевдовипадкових чисел, залежно від його розміру та інших властивостей. Досліджено швидкість і продуктивність дванадцяти алгоритмів сортування за допомогою сучасної мови веброзробки TypeScript: Bubble, Selection, Insertion, Shell, Merge, Quick, TimSort, Smooth, Introspective, Gravity, Radix і вбудованого методу sort(). Було порівняно фактичний час виконання кожного алгоритму для наборів псевдовипадкових цілих чисел від 1000 до 100 000 000 елементів. Хоча вбудований метод TypeScript гнучко адаптується до різних ситуацій, результати дослідження показали, що він не завжди дає найкращі результати та може бути хорошим вибором лише для даних до 1000 елементів. Вбудований метод програє алгоритмам швидкого сортування, інтроспективного сортування, сортування злиттям на великих масивах. Дослідження підтвердило, що кожен розглянутий в роботі алгоритм сортування має переваги та недоліки. Вибір відповідного алгоритму сортування для конкретного завдання розробки залежить від розміру та конкретних характеристик даних, а також від мови програмування. На вибір також впливає бажаний рівень ефективності сортування та вимоги до стабільності алгоритму. Здобуті результати дослідження дозволяють ефективно вибирати ефективний алгоритм за певних умов і даних.

Ключові слова: алгоритми сортування, ефективні алгоритми, час роботи, продуктивність, сортування, тестування, TypeScript.

DOI 10.15276/imms.v14.no3.154
УДК 004.056.5

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 154-162

ДОСЛІДЖЕННЯ СТАТИСТИЧНИХ ХАРАКТЕРИСТИК АЛГОРИТМУ ПЕРЕМІШУВАННЯ ЕЛЕМЕНТІВ ДВОВИМІРНИХ МАТРИЦЬ ЯК ОСНОВА ДЛЯ ФОРМУВАННЯ ЗМІННИХ S-БЛОКІВ

Г.В. Ахмаметьєва[1], А.І. Гарбуз[2]

[1] Національний університет «Одеська юридична академія»
28,. Рішельєвська вул, Одеса, 65000, Україна
[2] ВСП «Фаховий коледж вимірювань» ДУІТЗ
13, Спиридонівська вул., Одеса, 65020, Україна
Emails: anna.odessitka@gmail.com[1], garbuzartem@okv.suitt.edu.ua[2]

Наводиться опис та аналіз структури алгоритму формування стеганографічного ключа, основою якого є перемішування рядків і стовпців двовимірної матриці, що містить неповторювані числові значення як послідовність вибору пікселів/блоків зображення для вбудовування повідомлення. Охарактеризована структура алгоритму з точки зору застосування перемішаних двовимірних матриць для формування змінних S-блоків в криптографічному перетворенні, виявлені переваги і недоліки структури та можливі напрями удосконалення процедури перемішування і підвищення швидкодії алгоритму. Проведено тестування експериментальних S-блоків розміром 16×16, що здійснюють заміну 8-бітових блоків новим 8-бітовим значенням, статистичними тестами NIST, яке показало, що більшість сформованих таблиць відповідає вимогам надійних псевдовипадкових послідовностей. З метою подальшого застосування розробленого раніше алгоритму перемішування двовимірної матриці для задач криптографічних перетворень були розраховані статистичні властивості S-блоків - алгебраїчна степінь нелінійності, відстань нелінійності, коефіцієнти кореляції S-блоку, лавинні властивості S-блоку, період повернення S-блоку. Результати експерименту показали, що майже для всіх таблиць досягається значення алгебраїчної степені нелінійності 7, що теоретично дозволяє використання алгоритму перемішування для формування змінних S-блоків, але коливання інших показників вимагає подальшої модифікації алгоритму та дослідження впливу параметрів алгоритму на статистичні характеристики сформованих матриць замін.

Ключові слова: стеганографічний ключ, перемішування, двовимірна матриця, криптографічне перетворення, S-блок заміни, статистичні властивості, псевдовипадкова послідовність

Вступ. Сучасні технології та Інтернет-комунікації сприяють миттєвому обміну інформацією між користувачами різних країн світу. Засобами вебсайтів, електронної пошти, месенджерів, файлообмінників, хмарних сховищ можна передавати будь-які дані, як то текстові документи, аудіозаписи, фото, відео, тощо. Однак далеко не всі інструменти Інтернет-комунікації є безпечними. За умови передачі несекретної та загальнодоступної інформації існуючі системи моніторингу не створюють особливих проблем, хоч і можуть відслідковувати активність та зацікавленість користувачів. А ось необхідність відправити конфіденційну інформацію при відсутності захищених каналів зв’язку створює велику загрозу збереженню в таємниці персональної або комерційної інформації.

Вирішити цю задачу дозволяє застосування методів стеганографії, які забезпечують приховану передачу конфіденційних даних всередині нічим не примітного на перший погляд контейнеру. В якості контейнерів найбільш розповсюдженими є зображення, аудіо та відео через наявність в них надмірної інформації, що дозволяє забезпечити високу пропускну спроможність прихованого каналу зв’язку, тобто можливість вбудовування в контейнер значний обсяг даних.

Переваги використання стеганографії полягають в широкому виборі методів та їх програмних реалізацій, переважна більшість яких здійснює вбудовування секретної інформації в цифрові зображення. В більшості стеганографічних застосунків не приділяється увага формуванню стеганографічного ключа - послідовності вибору елементів контейнера для занурення в нього даних. Іноді стеганографічне перетворення поєднують з криптографічним закриттям конфіденційних даних [1-3]. У зв'язку з чим в роботі [4] були проаналізовані підходи до формування стеганографічних ключів інших авторів [5-12] та розроблено алгоритм формування стеганографічного ключа на основі перемішування рядків і стовпців двовимірної матриці. В роботі [13] було проведено дослідження характеристик перестановки графічними тестами, які показали достатньо надійний псевдовипадковий розподіл числових елементів на площині та якісні характеристики автокореляційної функції.

Метою даної роботи є проведення більш детального дослідження статистичних характеристик розробленого в [4, 13] алгоритму, в основу якого покладено перемішування рядків і стовпців двовимірної матриці.

Задачами статті є:

1. Визначити переваги і недоліки в структурі алгоритму формування стеганографічного ключа;
2. Провести дослідження якості перестановок при застосування оригінального алгоритму [4] статистичними тестами NIST [14];
3. Проаналізувати можливість використання основи алгоритму стеганографічного ключа для формування змінних S-блоків для криптографічних перетворень.

Основна частина. Алгоритм формування стеганографічного ключа [4] передусім спрямований на те, щоб задати псевдовипадкову послідовність вибору елементів контейнеру (блоків, пікселів) для вбудовування повідомлення, а з цього випливає, що елементи такої послідовності не можуть повторюватись, оскільки перезапис нового біту повністю видаляє попередній, що припускає можливість застосування даного алгоритму для формування змінних S-блоків заміни для криптографічних перетворень. Крім того, алгоритм побудований таким чином, що не потребує збереження матриці ключа окремим файлом, оскільки його можна відтворити на основі секретного паролю та вхідних параметрів, що забезпечує неможливість відтворення секретного ключа без знання паролю.

В роботі [4] описано послідовність обчислень та пояснення щодо математичних операцій, однак не наведено основні кроки алгоритму, які в повній мірі задають перемішування елементів матриці. Для подальшого розуміння структури алгоритму та виявлення його недоліків наведемо основні кроки алгоритму [13].

Крок 1. Отримання вхідних даних:

- розмір зображення-контейнера $H \times W$,
- розмір блоку $m \times n$,
- пароль key (послідовність з не менш як восьми символів),
- алгоритм хешування $algorithm$ (за умовчанням MD5),
- кількість перемішувань $perturbations$ (за умовчанням 30).

Крок 2. Підготовка матриці стеганографічного ключа.

2.1. Обчислення розміру матриці стеганографічного ключа

$$M = \left\lfloor \frac{H}{m} \right\rfloor, N = \left\lfloor \frac{W}{n} \right\rfloor,$$

де $\lfloor \bullet \rfloor$ - округлення до найменшого цілого.

2.2. Заповнення матриці послідовністю чисел від 1 до MN

$$table_{i,j} = j + (i-1)N.$$

Крок 3. Формування додаткових ключів.

3.1. Отримання результату хешування паролю key

$$hash_l = algorithm(key), \overline{l = 1, L},$$

$$algorithm \in \{MD5, SHA1, SHA256, SHA384, SHA512\},$$

де $hash$ - вектор з десятковими значеннями хеш-функції.

3.2. Обчислити:

$$t_k = \left\lfloor \frac{L}{Z_k} \right\rfloor \text{ або } t_k = \left\lceil \frac{L}{Z_k} \right\rceil,$$

де L - довжина вектору $hash$, $\lfloor \bullet \rfloor$ - округлення до найменшого цілого, $\lceil \bullet \rceil$ - округлення до найбільшого цілого, $Z_k \in \square$, $Z_k > 1, k = \overline{1,8}$.

3.3. Визначити значення додаткових ключів:

$$key1 = \left| \left(hash_{t_8} - hash_{t_6} \right) \left(hash_{t_1} + hash_{t_5} \right) \right|,$$

$$key2 = \left| \left(hash_{t_2} - hash_{t_7} \right) \left(hash_{t_3} + hash_{t_8} \right) \right|,$$

$$key3 = \left| \left(hash_{t_7} - hash_{t_4} \right) \left(hash_{t_5} + hash_{t_2} \right) \cdot hash_{t_6} \cdot hash_{t_1} \right|,$$

$$key4 = \left| \left(hash_{t_2} - hash_{t_1} \right) \left(hash_{t_7} + hash_{t_6} \right) \cdot hash_{t_4} \cdot hash_{t_3} \right|.$$

Крок 4. Формування параметрів лінійного конгруентного генератора і генерація псевдовипадкових послідовностей $SeqI$ і $SeqJ$.

4.1. Обчислити модулі

$$p_1 = M \cdot perturbations,$$

$$p_2 = N \cdot perturbations.$$

4.2. Визначення параметрів двох лінійних конгруентних генераторів.

4.2.1. Для модулів p_1 і p_2 визначити прості дільники.

4.2.2. Обчислити добуток унікальних простих дільників, де під унікальними простими дільниками розуміємо такі значення, що не повторюються. Результат - A_1, A_2.

4.2.3. Якщо $p_1 \equiv 0 (\mathrm{mod}\, 4)$, то $A_1 = 2A_1 + 1$, інакше $A_1 = A_1 + 1$.

Якщо $p_2 \equiv 0 (\mathrm{mod}\, 4)$, то $A_2 = 2A_2 + 1$, інакше $A_2 = A_2 + 1$.

4.2.4. Обчислити $B_1 = \mathrm{mod}(p_1 \cdot X, Y)$ і $B_2 = \mathrm{mod}(p_2 \cdot X, Y)$, де $X, Y \in \square$.

4.2.5. Якщо $(B_1, p_1) = 1$, то $B_1 = B_1$, інакше доки $(B_1, p_1) \neq 1$, $B_1 = B_1 - 1$.

Якщо $(B_2, p_2) = 1$, то $B_2 = B_2$, інакше доки $(B_2, p_2) \neq 1$, $B_2 = B_2 - 1$.

4.3. Генерація псевдовипадкових послідовностей

$$prg_i^1 = A_1 prg_{i-1}^1 + B_1 (\mathrm{mod}\, p_1), \overline{i = 1, p_1}, prg_0^1 = key1,$$

$$prg_j^2 = A_2 prg_{j-1}^2 + B_2 (\mathrm{mod}\, p_2), \overline{j = 1, p_2}, prg_0^2 = key2.$$

Крок 5. Перемішування матриці *table*.

Для $\overline{i = 1, p_1}$, $\overline{j = 1, p_2}$:

5.1. Визначення номеру рядка і стовпця:

$$x = \mathrm{mod}(prg_i^1, M), y = \mathrm{mod}(prg_j^2, N).$$

5.2. Визначення величини циклічного зсуву рядка і стовпця:

$$shiftI = rezI(\mathrm{mod}\, N),\ rezI = XOR^{mod}(prg_i^1, key3, mn),$$
$$shiftJ = rezJ(\mathrm{mod}\, M),\ rezJ = XOR^{mod}(prg_j^2, key4, nn),$$

де XOR^{mod} - операція складання за модулем mod, $mod = mn$ - максимальна цифра у prg_i^1, $mod = nn$ - максимальна цифра у prg_j^2.

5.3. Циклічний зсув рядка і стовпця матриці $table$:

$$table_{i,:} = circshift(table_{i,:}, shiftI),$$
$$table_{:,j} = circshift(table_{:,j}, shiftJ),$$

де $circshift$ - операція циклічного зсуву.

Результат – матриця $table$ з перемішаними номерами блоків $m \times n$ для зображення розміром $H \times W$.

В даному алгоритмі слід детально пояснити сутність операції XOR^{mod}, оскільки вона є нестандартною та не використовується в криптографічних та стохастичних алгоритмах. Операція XOR^{mod} передбачає порозрядне сумування двох десяткових цифр *x* та *y* за модулем деякої третьої цифри *w*, де значення *w* становить максимальну цифру числа псевдовипадкової послідовності. Наприклад, маємо число псевдовипадкової послідовності $X = \{x_7, x_6, x_5, x_4, x_3, x_2, x_1, x_0\}$ та ключ $Y = \{y_7, y_6, y_5, y_4, y_3, y_2, y_1, y_0\}$. Нехай максимальною цифрою у X є x_5, тоді порозрядним модулем буде $w = x_5$. Тоді результатом операції XOR^{mod} буде число $Z = \{z_7, z_6, z_5, z_4, z_3, z_2, z_1, z_0\}$, кожний розряд якого обчислюється за формулою

$$z_i = x_i + y_i \left(\mathrm{mod}\, w\right), i = \overline{0,7}.$$

І оскільки псевдовипадкове число *X* змінюється на кожній ітерації алгоритму, то і модуль *w* буде змінним, що забезпечує певну непередбачуваність у величинах зсуву рядків та стовпців.

Послідовність і взаємозв'язок обчислень алгоритму можна подати у вигляді схеми, наведеної на рис.1 [13, 15], де скорочення ППВЧ означає послідовність псевдовипадкових чисел, суцільна лінія стосується обчислень для циклічного зсуву рядків, пунктирна лінія – для стовпців.

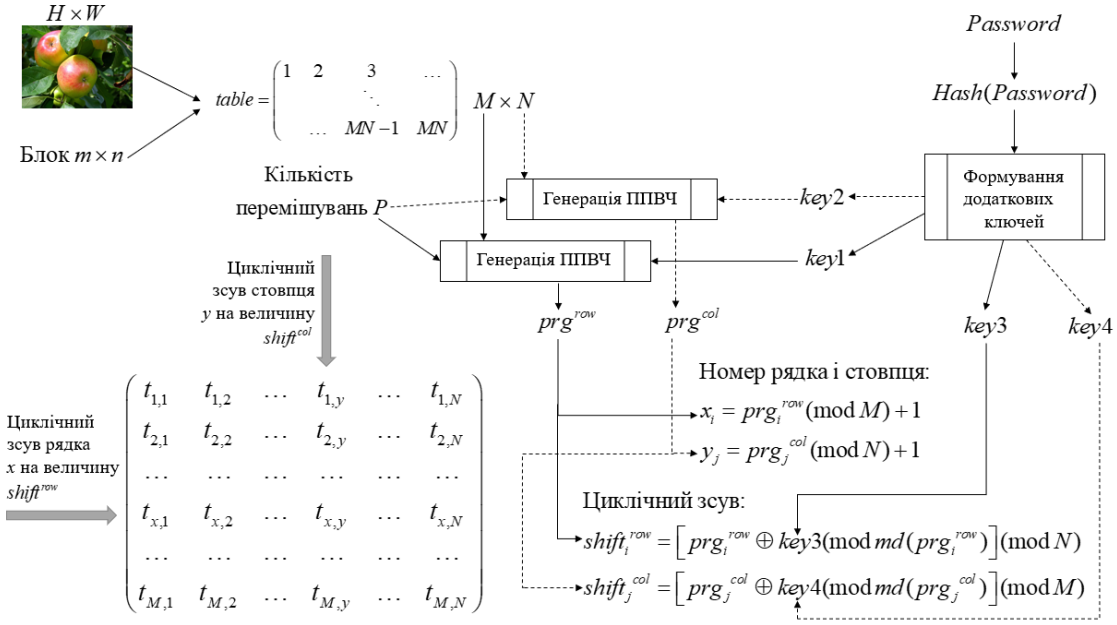


Рис.1. Структурна схема алгоритму формування стеганографічного ключа

Слід зазначити, що в алгоритмі міститься багато переваг, які дозволяють гарантувати захищеність стеганографічного ключа та можливість його відтворення на основі знання секретного паролю та параметрів перетворення, зокрема:

– в перемішуванні двовимірної матриці відбувається циклічний зсув як рядків, так і стовпців, причому змінними є як величина зсуву для кожного рядку/стовпця на кожному кроці, так і вибір самого рядка/стовпця, який задається псевдовипадковою послідовністю;

– всі внутрішні параметри прив'язані до секретного ключа –хеш-значення паролю та числа перемішувань P, тобто заміна ключа призводить до повної зміни результатів перестановки та гарантує непередбачуваний характер процедури перемішування;

– використання нестандартної операції модульного порозрядного сумування XOR^{mod} з одного боку є перевагою, з іншого недоліком, оскільки обмежує можливість реалізації алгоритму лише програмними засобами та потребує обчислень в десятковій системі числення, що вимагає більших обчислювальних затрат порівняно з двійковою системою числення та негативно впливає на швидкодію;

– запропонований алгоритм забезпечує рівномірний розподіл перших 5-10% елементів з послідовності від 0 до MN–1 по всій матриці [4] вже при $P = 10$.

До недоліків (з погляду адаптації алгоритму до задачі формування змінних S-блоків заміни криптографічних перетворень) можна віднести наступні:

– занадто складна процедура обчислення додаткових ключів (крок 3) містить константи, які задають вибір складових хешу, та подальше обчислення математичних виразів, обґрунтування яких в [4, 13] не наводиться;

– на кроці 4 при підборі параметрів лінійних конгруентних генераторів для забезпечення максимального періоду слід вірно підібрати константи A і B, для яких необхідно розкласти числа p_1 і p_2 на прості множники, тобто слід застосувати обчислювально складні операції факторизації складених чисел.

В сучасних криптографічних алгоритмах приділяється увага формуванню надійних S-блоків заміни, які забезпечують нелінійний шар криптографічного перетворення, що в значній мірі ускладнює можливість застосування диференціального, лінійного та інших методів криптоаналізу. В більшості алгоритмів S-блоки заміни є складовою частиною структури і є незмінними, деякі передбачають можливість використання сторонніх таблиць заміни, зокрема національний стандарт шифрування ДСТУ 7624:2014 [16, 17].

Оскільки S-блок представляє собою бієктивну матрицю заміни, де кожному входу єдиним способом ставиться у відповідність вихідний елемент, тобто S-блок суть є перестановкою значень від 0 до 2^N–1, поданої у вигляді квадратної матриці, а отже описаний вище алгоритм можна адаптувати та використовувати для формування змінних S-блоків в криптографічних шифрах.

З метою подальшої модифікації алгоритму перестановки для задач формування змінних S-блоків криптографічних перетворень, проаналізуємо статистичні властивості двовимірних матриць, отриманих запропонованим в [4] алгоритмом. Оскільки сучасні криптографічні стандарти, зокрема AES та ДСТУ 7624:2014, здійснюють байтові заміни, будемо розглядати S-блоки розміром 16×16, елементами якого є числа від 0 до 255.

За допомогою описаного вище алгоритму формування стеганографічного ключа було сформовано 1024 S-блоків розміром 16×16, з яких 512 з використанням хеш-функції SHA-256 та 512 з використанням хеш-функції SHA-256 при числі перемішувань $P = 12$. Отримані матриці подані для тестування статистичними тестами NIST [14], результати яких наведено в таблиці 1.

З таблиці 1 видно, що більшість послідовностей, отриманих перемішуванням запропонованим алгоритмам, проходять статистичні тести та можуть бути використані в

задачах криптографії для формування змінних S-блоків або для генерування псевдовипадкових послідовностей для потокових шифрів.

З'ясуємо, чи можна розглянутий алгоритм (можливо з деякими модифікаціями) використовувати як основу для перемішування даних в S-блоках заміни. Для цього був проведений обчислювальний експеримент на основі:

– група 1 – 512 S-блоків розміром 16×16, отриманих 12-ми циклами перемішування рядків і стовпців матриці з використанням хеш-функції SHA-256;

– група 2 – 512 S-блоків розміром 16×16, отриманих 18-ми циклами перемішування рядків і стовпців матриці з використанням хеш-функції SHA-256;

– група 3 – 512 S-блоків розміром 16×16, отриманих 12-ми циклами перемішування рядків і стовпців матриці з використанням хеш-функції SHA-512;

– група 4 – 512 S-блоків розміром 16×16, отриманих 18-ми циклами перемішування рядків і стовпців матриці з використанням хеш-функції SHA-512.

Таблиця 1.

Результати стохастичних тестів NIST

№	Тест	S-блоки (SHA-256)		S-блоки (SHA-512)		Кількість S-блоків, що пройшли тест, %	
		P-value	Pass rate	P-value	Pass rate	S-блоки (SHA-256)	S-блоки (SHA-512)
1	Monobit test	1	+	1	+	100	100
2	Frequency within block test	0,073	+	0,435	+	99,4	99,6
3	Runs test	0,979	+	0,731	+	100	100
4	Longest run ones in a block test	0,857	+	0,219	+	100	99,4
5	Binary matrix rank test	0,412	+	0,718	+	99,4	98,6
6	DFT test	0,096	+	0,882	+	99,02	99,2
7	Non overlapping template matching test	1	+	0,999	+	100	100
8	Overlapping template matching test	0,707	+	0,998	+	99,8	99,6
9	Maurers universal test	0,444	+	0,162	+	99,6	99,8
10	Linear complexity test	0,507	+	0,179	+	99,2	98,8
11	Serial test	0,991	+	0,964	+	100	100
12	Approximate entropy test	0,999	+	0,998	+	100	100
13	Cumulative sums test	0,719	+	0,784	+	100	100
14	Random excursion test	0,065	+	0,321	+	97,2	96,1
15	Random excursion variant test	0,046	+	0,088	+	98	97,1

Для кожної групи були обчислені показники алгебраїчної степені нелінійності, відстань нелінійності, коефіцієнти кореляції S-блоку, лавинні властивості S-блоку, та період повернення S-блоку, наведені в таблиці 2.

З таблиці 2 видно, що експериментальні набори S-блоків задовольняють показникам алгебраїчної степені нелінійності, де майже всі S-блоки мають значення 7, та коефіцієнтам кореляції S-блоку, де всі коефіцієнти наближені до 0. Значення показників відстані нелінійності для значної кількості матриць не досягають 100, яке на практиці вважають орієнтиром, а лавинні властивостей S-блоку мають достатньо значні відхилення від 128. Період повернення S-блоку також є різним для експериментальних

блоків, і така помітна різниця пояснюється використанням великого числа псевдовипадкових чисел в структурі алгоритму та нерівномірним розподілом перестановок.

Таблиця 2.

Оцінка статистичних властивостей S-блоків

Показник	Значення показника	Кількість S-блоків з відповідними значеннями показників, %			
		Група 1	Група 2	Група 3	Група 4
Алгебраїчна степінь нелінійності	7 6	99.6% 0.4%	99.4% 0.6%	99.2% 0.8%	100% 0%
Відстань нелінійності	$x<96$ $96<=x<100$ $x>=100$	6.4% 36.1% 57.5%	7.2% 36.5% 56.3%	6.6% 38.3% 55.1%	4.7% 35.7% 59.6%
Коефіцієнти кореляції S-блоку	середнє значення	від -0.0237 до 0.0244	від -0.0227 до 0.0222	від -0.0249 до 0.0222	від -0.0205 до 0.0273
Лавинні властивості S-блоку	середнє значення	від 124 до 132.625	від 124 до 132.375	від 123.8125 до 132.8125	від 124.875 до 133.25
Період повернення S-блоку	середнє значення	4834855.51	15072582.9	37296991.63	13132147.81

Висновки. Незважаючи на певні просідання в значеннях окремих показників та тестів, вважаємо доцільними подальші дослідження розглянутого алгоритму перемішування за умови певних модифікацій в його структурі, а саме:

– замість секретного паролю та його хеш-значення використовувати повноцінний ключ довжиною не менше 256 бітів;

– розроблення зрозумілої та обґрунтованої процедури розширення ключа для отримання величин зсуву рядків і стовпців матриці;

– спрощення порядку вибору рядків і стовпців матриці та використання обчислювально легких операції для збільшення швидкодії перемішування.

Також має сенс дослідити мінімальну кількість ітерацій перемішування, яка забезпечувала б відповідність всім показникам статистичних властивостей S-блоків.

Оскільки в більшості криптографічних алгоритмах використовується більше однієї таблиці замін, а принаймні чотири, то слід провести дослідження властивостей комплексу, який включає від чотирьох і більше S-блоків. Також можливим є використання алгоритму перемішування в потоковому шифруванні для генерації гами за умови забезпечення високої швидкодії процедури перемішування. В подальших публікаціях означені питання будуть розглянуті більш детально.

Список літератури

1. Abbas Z., Saeed M.Q. Image Steganography using Cryptographic Primitives. *International Conference on Cyber Warfare and Security (ICCWS).* 2021, P 124-131. DOI: 10.1109/ICCWS53234.2021.9703017.
2. Alanzy M.; Alomrani R.; Alqarni B.; Almutairi S. Image Steganography Using LSB and Hybrid Encryption Algorithms. *Appl. Sci*. 2023, No.13. P. 11771. URL: https://doi.org/10.3390/app132111771
3. Bahaddad A.A., Almarhabi K.A. Sayed Abdel-Khalek. Image steganography technique based on bald eagle search optimal pixel selection with chaotic encryption. *Alexandria*

Engineering Journal. 2023. V. 75, P.41-54. URL: https://doi.org/10.1016/j.aej.2023.05.051

4. Ахмаметьєва Г.В., Бойко Н.В. Розробка алгоритму формування стеганографічного ключа для цифрових зображень. *«Шляхи розвитку науки в сучасних кризових умовах»: І Міжнародна науково-практична інтернет-конференція.* 2020. Т.1. С.32-35.
5. Vinodhini R.E., Vimalkumar K., Malathi P., Gireeshkumar T. A Highly Secured Image Steganography using Bernoulli's Chaotic Map and Binary Hamming Code. *International Journal of Pure and Applied Mathematics*. 2018. V. 118. No. 7. P.159-164.
6. Sahil, Sinwar D. A Steganography Technique based on chaos for Pseudo-Random LSB Images. *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*. 2018. V. 6. No. II. P.436-440.
7. Al-Bahadili H.. A secure block permutation image steganography algorithm. *International Journal on Cryptography and Information Security (IJCIS)*. 2013. V.3. No. 3. P.11-22.
8. Sunder R., Eswaran P., Nagalinga R. A. High capacity image steganography in the spatial domain using Lehmer code. *International Journal of Advance Research In Science And Engineering (IJARSE)*. 2015. V.4. No.5. P.91-99.
9. Nagalinga R., Sunder R. Hiding text in digital images using permutation ordering and compact key based dictionary. *ICTACT Journal on Image and Video Processing*. 2017. V.7. No.4. P.1497-1504.
10. Bassam H.S., Elsamani A.E.A., Gafar Z.A.S., Abdelmajid H.M. A Spatial Domain Image Steganography Technique Based on Pseudorandom Permutation Substitution Method using Tree and Linked List. *International Journal of Engineering Trends and Technology* (IJETT). 2015. V.23. No 4. P.209-217.
11. Nazari S., Eftekhari-Moghadam A.M., Mohammad-Shahram M. A novel image steganography scheme based on morphological associative memory and permutation schema. *Security and Communication Networks*. 2015. No.8. P.110–121.
12. Shakir M. Hussain, Naim M. Ajlouni. Key Based Random Permutation (KBRP). *Journal of Computer Science*. 2006. No. 2 (5). P.419-421.
13. Бойко Н.В. Удосконалення стеганографічного методу для цифрових зображень. Розробка алгоритму формування стеганографічного ключу: квал. роб. бак. Одеса: НУ «одеська політехніка», 2020. 67 с.
14. A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generators for Cryptographic Applications. *NIST Special Publication 800-22.* 2001.
15. Ахмаметьєва Г. Дослідження алгоритму формування стеганографічного ключа для побудови змінних криптографічних S-блоків. *Міжнародна науково-практичної конференція «Кіберпростір в умовах війни та глобальних викликів XXI століття: теорія та практика»*. 2023. С. 13-17.
16. ДСТУ 7624:2014. Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного блокового перетворення. К.: Мінекономрозвитку України, 2015.
17. Горбенко І. Д., Олійников Р.В., Казимиров О.В., Руженцев В.І., Кузнєцов О.О., Горбенко Ю.І., Дирда О.В., Долгов В.І., Пушкарьов А.І., Мордвінов Р.І. Симетричний блоковий шифр «Калина» – новий національний стандарт України. *Радіотехніка*. 2015. Вип. 181. С. 5–22.

RESEARCH OF THE STATISTICAL CHARACTERISTICS OF THE ALGORITHM FOR SHUFFLING THE ELEMENTS OF TWO-DIMENSIONAL MATRICES AS A BASIS FOR THE FORMATION OF VARIABLE S-BLOCKS

A.V. Akhmametieva[1], A.I. Garbuz[2]

[1] National University «Odesa Law Academy»
28, Rishilievska str., Odesa, 65000, Ukraine
[2] SSS «Professional College of Measurements of State University of Intellectual Technologies and Communications»
13, Spiridonivs'ka str., Odesa, 65020, Ukraine
Emails: anna.odessitka@gmail.com[1], garbuzartem@okv.suitt.edu.ua[2]

The article provides a description and analysis of the structure of the algorithm for formation of the steganographic key. Proposed algorithm is based on the shuffling of rows and columns of a two-dimensional matrix containing unique numerical values as a sequence of image pixel/block selection for embedding of the secret message. The structure of the algorithm was characterized from the point of view of the using of mixed two-dimensional matrices for the formation of variable S-blocks in cryptographic transformation. The advantages and disadvantages of the algorithms structure, possible directions for improving the mixing procedure and increasing the speed of the algorithm are revealed. Experimental S-blocks of size 16×16, which replace 8-bit blocks with new 8-bit values, were tested by NIST statistical tests, which showed that most of the formed tables meet the requirements of reliable pseudo-random sequences. In order to further apply the previously developed two-dimensional matrix shuffling algorithm for cryptographic transformation problems, the statistical properties of S-blocks were calculated - the algebraic degree of nonlinearity, the distance of nonlinearity, correlation coefficients of the S-block, avalanche properties of the S-block, and the return period of the S-block. The results of the experiment showed that for almost all tables, the value of the algebraic degree of nonlinearity is 7, which theoretically allows the use of the shuffling algorithm for the formation of variable S-blocks, but the fluctuation of other indicators requires further modification of the algorithm and the study of the influence of the algorithm parameters on the statistical characteristics of the formed substitution matrices.

Keywords: steganographic key, shuffling, two-dimensional matrix, cryptographic transformation, S-block of substitution, statistical properties, pseudorandom sequence.

DOI 10.15276/imms.v14.no3.163
УДК 004.94

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 163-171

КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ ІНВЕСТИЦІЙНОГО ПОРТФЕЛЮ КРИПТОВАЛЮТ НА ОСНОВІ БАЗ ДАНИХ ВІДКРИТОГО ІНТЕРЕСУ

Л.В. Бовнегра[1], Ю.І. Бабич[2], М.І. Бабич[3], В.В. Вознюк[4]

Національний університет «Одеська політехніка»
1, Шевченка пр., Одеса, 65044, Україна
Emails: dlv5@ukr.net[1], babich.u.i@op.edu.ua[2], babich.tiger@gmail.com[3], vavozniuk@gmail.com[4]

Проведено аналіз стратегій біржової торгівлі, заснованих на обсягах торгів (відкритого інтересу) та їх придатність для торгівлі криптовалютами. З розвитком криптовалютного ринку, де волатильність та ліквідність значно відрізняються від традиційних фінансових ринків, виникає необхідність адаптації класичних підходів до нових умов. Аналіз обсягу торгів та тікових даних дозволяє трейдерам оцінювати настрої ринку, знаходити точки входу та виходу, а також передбачати потенційні ринкові рухи. Значною перевагою криптовалютних бірж у порівнянні з класичними є наявність доступу в режимі реального часу до усіх ринкових даних, в тому числі відкритого інтересу. Котирування криптовалют та інформація про відкритий інтерес будуть основою вхідних даних для створеної в ході роботи кросплатформенної автоматичної системи моделювання портфелю криптовалют. Важливу роль відведено фільтрації активів, які використовуються зловмисниками для пампінгу, щоб уникнути потенційно значних збитків або взагалі ліквідації рахунку. Цінність даної роботи полягає в описі усіх моделей для фільтрації небезпечних активів, а також включення до інвестиційного портфелю найбільш безпечних та передбачуваних по своїй динаміці криптовалют. Ці моделі стали основою системи, що дає змогу інвестиційним фондам та іншим установам, які інвестують у криптовалюти, в режимі реального часу (24/7) корегувати інвестиційний портфель, максимізуючи його безпечність. Також була створена автоматична система моделювання інвестиційного портфеля, яка була перевірена на серверах під управлінням різних ОС і показала себе зручною в переносі та стабільною в роботі.

Ключові слова: портфель інвестицій, криптовалюти, пампінг, відкритий інтерес, скрипт, база даних, моделювання.

Вступ. Протягом кількох років більшість інвестиційних фондів у складі своїх інвестиційних портфелів мають значну кількість різноманітних альткоінів. Найвідоміші факти активності інвестиційних фондів в сегмент криптовалюти:

– Pantera Capital – один з найстаріших крипто-хедж-фондів, відомий своїми інвестиціями в альткоїни. Pantera інвестує в токени, пов'язані з блокчейн-проєктами, DeFi та часто бере участь в ICO (Initial Coin Offerings).

– Polychain Capital – фонд активно вкладає в альткоїни та спеціалізується на ранніх блокчейн-проєктах. Вони інвестують як у великі, так і в дрібні криптовалюти, а також беруть участь у запуску нових токенів та платформ.

– Multicoin Capital – спеціалізується на альткоїнах, включаючи Solana, Ethereum та інші. Їх інвестиції зосереджені на нових секторах, таких як DeFi, Web3 та смарт-контрактні платформи. Частка альткоїнів у їх портфелі може досягати від 30 до 50 %, залежно від ринкових умов.

Але, нажаль, усі фінансові ринки, так чи інакше використовуються для відмивання «брудних коштів», саме пампінг та дампінг є яскравими представниками цього процесу [4]. Коли зловмисники роблять пампінг криптовалюти, то

спостерігається значне і дуже динамічне зростання вартості цієї криптовалюти, яке може раптово зупинитись і навіть різко знизитись на той ціновий рівень, який був у криптовалюти до початку пампінгу [5]. Такий процес є дуже небезпечним для всіх учасників ринку, оскільки може спричинити значні збитки на торговому рахунку або взагалі привести до ліквідації рахунку. Наслідками цього може бути маржин кол. Саме тому дуже важливо мати простий і надійний механізм фільтрації криптовалют, який дозволить виключати зі списку інвестиційних інструментів ті криптовалюти, на яких відбувається пампінг, а також обирати ті криптовалюти, на яких можна безпечно торгувати, аналізуючи відкритий інтерес по цих криптовалютах.

Метою роботи є розробка автоматичної системи моделювання портфелю криптовалют на основі фільтрації пампінгу та аналізу відкритого інтересу.

Для досягнення поставленої мети в роботі вирішуються наступні **задачі**:

– створення скриптів на мові програмування Python, які будуть забезпечувати регулярне зчитування, запис в базу даних та аналіз інформації;

– розробка максимально простих алгоритмів на базі лінійних математичних моделей, які будуть швидко та надійно розподіляти криптовалюти в чорний список або в інвестиційний портфель;

– розробка бази даних для зберігання проміжних результатів по кожній криптовалютній парі;

– формування інвестиційного портфель криптовалют;

– аналіз ефективності роботи автоматичної системи моделювання портфелю криптовалют.

Об’єктом дослідження в роботі є процеси пампінгу та росту перспективних криптовалют на основі відкритого інтересу.

Предметом дослідження є методи виявлення процесів пампінгу, а також виявлення криптовалют, які торгуються по загальновідомим ринковим законам.

Рис.1. Фрагмент пампінгу на валютній парі LPT/USDT, кілька випадків у лютому 2024 року

Рис.2. Фрагмент пампінгу на валютній парі BCH/USDT, у лютому-березні 2024 року

На рис.1 та рис.2 представлено фрагменти пампінгу криптовалюти, які можна побачити навіть оком недосвідченого інвестора, порівнявши середній розмір добового приросту курсу криптовалюти та їх прирости або падіння, які були у дні пампінгу. Одночасно на популярних біржах впродовж дня може бути задіяно приблизно 5-10 різних криптовалют у пампінгу, найчастіше він зустрічається у ф'ючерсному розділі біржі [6]. Тому, виходячи з цього, необхідно регулярно за допомогою відкритого API зчитувати список всіх валют, доступних у ф'ючерсному розділі біржі, а потім зчитувати цінові дані по денним тайм-фреймам для цих активів.

В процесі моніторингу та фільтрації було проведено багатоступеневі розрахунки на основі нижче наведених моделей.

Для розрахунку середнього арифметичного об'ємів торгів:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad (1)$$

Для подальших розрахунків використовувалось співвідношення поточного об'єму торгів до середнього.

$$R_i = x_i / \bar{x} \qquad (2)$$

З використанням вищенаведених лінійних моделей було створено скрипт моніторингу ф'ючерсів на криптовалюти (рис.3),

```
client = Client(api_key, api_secret)
futures_list_file = 'fl.txt'
with open(futures_list_file, 'r') as file:
    futures_symbols = [line.strip() for line in file.readlines()]
for symbol in futures_symbols:
  try:
    klines = client.get_klines(symbol=symbol, interval=Client.KLINE_INTERVAL_1DAY)
    csv_file_path = f'{symbol}.csv'
    with open(csv_file_path, 'w', newline='') as csvfile:
        csv_writer = csv.writer(csvfile,delimiter=";")
        headers = ['timestamp', 'open', 'high', 'low', 'close', 'volume']
        csv_writer.writerow(headers)
        for kline in klines:
            timestamp = datetime.utcfromtimestamp(kline[0] / 1000).strftime('%Y-%m-%d %H:%M:%S')
            row_data = [timestamp, kline[1], kline[2], kline[3], kline[4], kline[5]]
            csv_writer.writerow(row_data)
```

Рис.3. Фрагмент програмного коду моніторингу

Результати роботи скрипта на основі математичних моделей (1),(2) наведено у таблиці 1.

Таблиця 1.

Список найбільш значних випадків пампінгу з 2021 по 2024 рік

Валюта	Коефіцієнт зростання об'єму	Коефіцієнт зростання ціни	Ціна (USDT)	Дата
XVGUSDT	52.41	25.88	0.007365	03.07.2023 00:00
ONGUSDT	47.33	15.48	0.3205	11.07.2023 00:00
NMRUSDT	47.25	25.8	19.73	02.09.2023 00:00
PEOPLEUSDT	43.75	17.16	0.02194	03.01.2024 00:00
FLMUSDT	42.85	17.7	0.0996	05.09.2023 00:00
ALPHAUSDT	40.64	7.04	0.1493	17.12.2023 00:00
USTCUSDT	38.73	3.15	0.0452	27.11.2023 00:00
BADGERUSDT	38.4	23.33	5.151	09.11.2023 00:00
IOTAUSDT	34.37	18.05	0.2553	29.11.2023 00:00
BNTUSDT	33.79	13.45	0.8503	13.11.2023 00:00
SRMUSDT	32.74	3.92	0.28404	15.11.2022 00:00
POWRUSDT	32.4	14.17	0.4938	07.01.2024 00:00
BTCSTUSDT	30.41	8.84	60.83	07.10.2021 00:00
ONTUSDT	30.09	6.39	0.2955	06.04.2023 00:00

З результатів розрахунку отримано дані, які підтверджують тези дослідників фінансових ринків:

– пампінг супроводжується значним сплеском об’ємів торгів (в десятки разів перевищуючим середній денний об’єм);

– в більшості випадків у якості валют для пампінгу вибирають валюти з дуже низькою ринковою вартістю.

На основі формул (1), (2) було розроблено алгоритм формування чорного списку криптовалют, які необхідно виключити не лише з поточного портфелю, але й прибрати усі умовні ордери з ринку, щоб випадково не сталося їх спрацювання у випадку різкого пампінгу.

Для того, щоб скрипт адекватно формував чорний список на основі перевищення поточного об’єму торгів, необхідно ввести порогове значення об’єму торгів, яке і буде тригером для переведення криптовалюти у список потенційних «жертв» пампінгу. Для цього створено скрипт моніторингу середніх об’ємів торгів по BTCUSDT за результатами денних тайм фреймів і глибиною історії 1 рік. Даний скрипт запускатиметься щоденно і видаватиме середній обсяг торгів за формулою (1), який буде використано у якості порогового значення [1].

Після запуску скрипта результат середнього коефіцієнту приросту об’єму для BTCUSDT, становив 7. Саме тому в роботу скрипта включено фільтрацію пампінгу величиною порогу – 7, і всі валютні пари, у яких денний приріст об’єму торгів більше цього порогу, буде направлено до чорного списку, а всі інші – до білого списку. Це дозволить виконувати моніторинг відкритого інтересу та отримання торгових сигналів по даних криптовалютах [2].

```
for filename in os.listdir(quotes_folder):
    if filename.endswith('.csv'):
        file_path = os.path.join(quotes_folder, filename)
        df = pd.read_csv(file_path,delimiter=";")
        open = df['open'].copy()
        close = df['close'].copy()
        volume = df['volume'].copy()
        timestamp = df['timestamp'].copy()
        sum=0
        sumv=0
        qv=0
        for i in range(1, len(open)):
         x1=abs(close.values[i]-open.values[i])
         sum=sum+(x1/open.values[i]*100)
         sumv=sumv+volume.values[i]
        adr=round(sum/i,2)
        adrv=round(sumv/i,2)
```

Рис.4. Фрагмент програмного коду розрахунку середнього коефіцієнту приросту

Для розробки системи моніторингу використовувалось API криптобіржі, а саме запити на отримання відкритого інтересу (списку проведених торгових операцій, в яких вказаний тип операції та її об'єм).

Різні криптобіржі називають ці запити по різному, але усі вони передають наступну інформацію:

– біржа Bybit – market/recent-trade;
– біржа Binance – /api/v3/trades;
– біржа KuCoin – /market/historie;
– аналітичний сервіс crypto.com – /public/get-trades.

Для зберігання даних по кожній валютній парі було розроблено базу даних. Структуру полів однієї з таблиць бази наведено в таблиці 2.

Таблиця 2.

Структура таблиці бази даних

Назва поля	Тип даних	Призначення
execId	текстовий	ключ таблиці, зберігає унікальний ідентифікатор торгової операції
symbol	текстовий	код криптовалюти
side	текстовий	напрямок (buy або sell)
size	числовий	об'єм операції
price	числовий	ціна операції
time	дата/час	час операції

Розробка бази даних та скриптів, які регулярно її заповнюють, дозволили отримати дані стосовно кількості торгових операцій на ф'ючерсному ринку по різним валютним парам. Таким чином, щоденно здійснюється близько 2 млн. торгових операцій по кожній валютній парі. Максимальне значення в період дослідження становило 4.9 млн. Слід зазначити, що основний параметр, який впливає на напрямок руху ціни у розрізі дослідження відкритого інтересу – це співвідношення об'ємів продавців та покупців на внутрішньо денних тайм фреймах (приблизно 60 хвилин) та змінами ціни за цей же часовий період [3].

Для оцінки зв'язку приросту ціни по відношенню до об'ємів продавців та покупців використовуємо коефіцієнт кореляції:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \quad (3)$$

Фільтр другого рівня обиратиме для торгівлі лише ті валютні пари, у яких є позитивне значення коефіцієнта кореляції більше, ніж 0.5. Усі інші валютні пари не враховуватимуться, оскільки в них не має статистично підтвердженого зв'язку між приростами ціни (а вони і визначають напрямок її руху) та співвідношенням обсягів торгів учасників ринку [7].

Результати найкращих 10 валютних пар представлено у таблиці 3.

Таблиця 3.

Список валютних пар з найбільшою кореляцією

Валютна пара	Коефіцієнт кореляції
ETH/USDT	0.81
SOL/USDT	0.78
ADA/USDT	0.75
BNB/USDT	0.71
AVAX/USDT	0.65
DOT/USDT	0.64
LINK/USDT	0.61
MATIC/USDT	0.59
LTC/USDT	0.56
SHIB/USDT	0.53

Слід зазначити, що більше 80 % валютних пар мають коефіцієнт кореляції менше 0.5, і близько 10 % з них – у вигляді від'ємного числа.

Було проведено також аналіз середніх величин приросту об'ємів торгів (на базі роботи першого скрипта вказаного в роботі), який показав, що всі рекомендовані валютні пари внаслідок моделювання портфелю не включені до чорного списку і мають низькі середні величин приросту об'ємів торгів.

Таблиця 4.

Список валютних пар з низькими середніми приростами об'ємів торгів

Валютна пара	Середня величина приросту об'ємів торгів
ETH/USDT	4.2
SOL/USDT	2.5
ADA/USDT	3.2
BNB/USDT	2.3
AVAX/USDT	3.1
DOT/USDT	2.8
LINK/USDT	2.4
MATIC/USDT	3.3
LTC/USDT	2.65
SHIB/USDT	2.2

В результаті аналізу коефіцієнта кореляції та середніх величин приросту об'ємів торгів, було сформовано інвестиційний портфель, в який увійшли всі 10 валютних пар, які пройшли перевірку. Частка кожної валютної пари в інвестиційному портфелі буде однаковою, оскільки не має достовірних даних стосовно впливу відкритого інтересу на потенційну дохідність криптовалют.

Таблиця 5.

Структура отриманого інвестиційного портфелю криптовалют

Валютна пара	Частка в інвестиційному портфелі, %
ETH/USDT	10
SOL/USDT	10
ADA/USDT	10
BNB/USDT	10
AVAX/USDT	10
DOT/USDT	10
LINK/USDT	10
MATIC/USDT	10
LTC/USDT	10
SHIB/USDT	10

Розроблена автоматична система є кросплатформенною. На протязі декількох місяців вона працювала на сервері без втручання оператора під управлінням операційної системи Debian Linux 12, потім – під управлінням ОС Windows Server 2016. Для перевірки стабільності та надійності роботи системи було обрано спеціальну конфігурацію з не великим обсягом оперативної пам'яті – 512 мегабайт та одноядерним процесором.

Для аналізу щоденних котирувань валютних пар та відкритого інтересу в середньому завантажувалось близько 20 гігабайт даних. За цей час не було виявлено витоків пам'яті внаслідок роботи скриптів, що є основою автоматичної системи, swap-файл жодного разу не збільшувався до 500 мегабайт.

Максимальна зручність переносу між різними серверами досягається за рахунок використання мови програмування Python 3.9 та мінімуму бібліотек в коді скриптів (os, math, pandas , datetime, csv, bybit).

Висновки. В роботі вирішено важливі науково-практичні задачі:

– запропоновано прості та ефективні лінійні моделі фільтрації криптовалют, які запобігають включенню ризикованих активів до інвестиційного портфелю;

– розроблено максимально прості алгоритми, які будуть швидко та надійно розподіляти криптовалюти в чорний список або в інвестиційний портфель;

– розроблено базу даних для зберігання проміжних результатів по кожній криптовалютній парі;

– сформовано збалансований інвестиційний портфель з однаковими частками кожної криптовалюти;

– запропоновано моделі відбору криптовалют, які доцільно включити в інвестиційний портфель;

– сформовано інвестиційний портфель з однаковими частками по кожній з криптовалют;

– розроблено автоматичну кросплатформену систему моделювання портфелю криптовалют з використанням мови Python та бази даних. Така система може стабільно працювати в режимі реального часу на серверах з мінімальними технічними характеристиками та під управлінням різних ОС. В процесі розробки даної системи було проведено аналіз різних мов програмування, оптимальним вибором стала мова програмування Python через велику кількість офіційних API від популярних криптобірж, а також через широкі можливості бібліотеки Pandas для обробки різних видів даних.

Також паралельно було проведено аналіз популярних криптобірж та їх API. Аналізувались такі параметри, як швидкодія, стабільність, великі ліміти денних запитів до серверу біржи, зручність парсингу отриманих даних, низький рівень надлишковості вхідних даних. Оптимальною виявилась біржа Bybit.

Список літератури

1. Velez O., Capra G. The Complete Guide to Market Breadth Indicators. Marketplace Books, 2011. 240 c.
2. Foresi S., Markese J. (). Volume Spread Analysis Explained. Technical Analyst Press, 2004. 180 c.
3. Xu J., Livshits B., Gervais A. The Anatomy of a Cryptocurrency Pump-and-Dump Scheme. *Proceedings of the 28th USENIX Security Symposium.* 2019
4. Le Pennec G., Fiedler I., Ante L. Pump and Dump schemes in the cryptocurrency market. *Economics Letters*. 2020. V.197. 109646.
5. Silva A. F., Kim M. (). Pump-and-Dump in Cryptocurrency Markets: Empirical Evidence. *Journal of Financial Markets*, 2020. V.54. P. 100-123.
6. Houben R., Snyers A. Cryptocurrencies and blockchain: Legal context and implications for financial crime, money laundering and tax evasion. European Parliament, 2018.
7. Zohar A. (). Bitcoin: under the hood. *Communications of the ACM*. 2015. V.58(9). P.104-113.

COMPUTER MODELING OF INVESTMENT PORTFOLIO OF CRYPTOCURRENCIES BASED ON DATABASES OF OPEN INTEREST

L.V. Bovnegra[1], Y.I. Babych[2], M.I. Babych[3], V.V. Vozniuk[4]

National Odesa Polytechnic University
1, ShevchenkoAve., Odesa, 65044, Ukraine
Emails: dlv5@ukr.net[1], babich.u.i@op.edu.ua[2], babich.tiger@gmail.com[3], vavozniuk@gmail.com[4]

The article provides an in-depth analysis of trading strategies based on trading volumes (open interest) and their suitability for cryptocurrency trading. With the development of the cryptocurrency market, where volatility and liquidity differ significantly from traditional financial markets, there is a need to adapt classical approaches to new conditions. Analyzing trading volumes and tick data enables traders to assess market sentiment, identify entry and exit points, and predict potential market movements. A significant advantage of cryptocurrency exchanges compared to traditional ones is the availability of real-time access to all market data, including open interest. Cryptocurrency quotes and open interest data will form the basis of the input for the cross-platform automated cryptocurrency portfolio modeling system developed in the course of this research. A key role is assigned to filtering assets used by malicious actors for pump schemes to avoid potentially significant losses or even account liquidation. The value of this work lies in describing all models for filtering risky assets and incorporating the safest and most predictable cryptocurrencies into the portfolio based on their dynamics. These models form the foundation of a system that allows investment funds and other institutions investing in cryptocurrencies to adjust their portfolios in real-time (24/7), maximizing their safety. Additionally, an automated portfolio modeling system was created, which was tested on servers running different operating systems and proved to be portable and stable.

Keywords: investment portfolio, cryptocurrencies, pump schemes, open interest, script, database, modeling.

DOI 10.15276/imms.v14.no3.172
УДК 004.056.53

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 172-182

ЗАСТОСУВАННЯ ОБФУСКАЦІЇ ДЛЯ ЗАХИСТУ МЕТАДАНИХ ФАЙЛІВ ВІД НЕСАНКЦІОНОВАНОГО ДОСТУПУ

Є.С. Булгаков, Н.І. Кушніренко, В.В. Подуфалов, В.О. Назаров

Національний університет «Одеська політехніка»
1, Шевченка пр., Одеса, 65044, Україна
Email: infsec2011@gmail.com

Розглянуто проблему захисту метаданих файлів від несанкціонованого доступу за допомогою обфускації. Метадані відіграють ключову роль у сучасних цифрових системах, забезпечуючи важливу інформацію про файли, таку як авторство, дата створення, геолокація, тип пристрою та інші атрибути, що допомагають у їх ідентифікації та класифікації. Однак ці дані можуть стати вразливими для кібератак, оскільки зловмисники можуть використовувати метадані для збору конфіденційної інформації або здійснення атак на користувачів та організації. Детально проаналізовано різні методи обфускації метаданих, зокрема шифрування, маскування та фальсифікацію, кожен з яких має свої переваги і недоліки. Шифрування дозволяє забезпечити високий рівень захисту, однак вимагає управління ключами, що може бути складним для великих організацій. Маскування передбачає заміну реальних значень метаданих псевдонімами або випадковими значеннями, зберігаючи при цьому функціональність файлів. Фальсифікація метаданих полягає у створенні неправдивої інформації для введення зловмисників в оману. Окрім того, у статті запропоновано концепцію розробки спеціалізованого програмного забезпечення для автоматизованого захисту метаданих, яке дозволяє користувачам автоматично обфускувати або видаляти метадані файлів під час їх обробки чи передачі через мережу. Програмне забезпечення також включає можливість групової обробки файлів, що є важливим для організацій, які працюють із великими обсягами даних. Такі рішення є актуальними в умовах сучасних кіберзагроз, оскільки забезпечують високий рівень конфіденційності та захисту даних. Важливим аспектом є те, що запропоноване програмне забезпечення не лише обфускує дані, але й інтегрується з іншими системами для автоматизації процесів захисту.

Таким чином, у роботі підкреслено важливість застосування обфускації як інструменту для підвищення рівня інформаційної безпеки та захисту конфіденційної інформації. Запропоноване програмне рішення є перспективним кроком у вирішенні проблеми витоків даних через метадані та може знайти застосування у різних галузях, включаючи медицину, освіту, архітектуру, геймдев, де захист даних відіграє ключову роль.

Ключові слова: захист метаданих, обфускація, шифрування, маскування, фальсифікація, програмне забезпечення.

Вступ. У сучасному світі інформаційні технології стають дедалі важливішими в усіх сферах людської діяльності, що призводить до безпрецедентного зростання обсягів даних, які передаються, зберігаються та обробляються. Відомі випадки, коли вразливість метаданих призводила до значних фінансових втрат. Наприклад, в одному з кейсів компанія з Лондона втратила близько 500 000 доларів США через компрометацію конфіденційних метаданих їхніх файлів під час передачі через мережу [1]. В іншому дослідженні, проведеному в області захисту метаданих, було показано, що понад 80% конфіденційних документів, які зберігаються в хмарних системах, містять незахищені метадані, що може бути використано зловмисниками для атак [2]. Поряд із вмістом файлів, значну роль відіграють метадані — інформація про файли, яка використовується для забезпечення їх ідентифікації, класифікації, індексації та впорядкування. Метадані містять такі відомості, як ім'я автора, дата створення або редагування файлу, тип пристрою, на якому був створений файл, геолокація тощо. У багатьох випадках метадані

несуть не менш важливу, а іноді навіть критично важливу інформацію, яка може бути використана для отримання доступу до приватних або конфіденційних даних.

Однією з основних проблем, що виникають у зв'язку з використанням метаданих, є їхня вразливість до несанкціонованого доступу та використання. Наприклад, зловмисники можуть отримати доступ до метаданих документів або мультимедійних файлів, щоб виявити інформацію про автора, місцезнаходження або інші важливі дані, які не призначені для загального доступу. Це може призвести до порушення приватності, конфіденційності або навіть безпеки користувача.

Одним із ефективних способів захисту метаданих є застосування методу обфускації, який передбачає перетворення або приховування метаданих таким чином, щоб вони ставали нерозбірливими або недоступними для зловмисників. Обфускація метаданих дозволяє забезпечити конфіденційність інформації, що міститься у файлах, зберігаючи при цьому їхню функціональність для законних користувачів. Серед програмних рішень для обфускації можна виділити такі інструменти, як ExifTool та Metadata Anonymization Toolkit (MAT), які вже успішно застосовуються для захисту метаданих у великих корпораціях [3]. Проте вони мають свої недоліки, такі як недостатня автоматизація або повне видалення метаданих, що може знижувати зручність використання для кінцевих користувачів. Це створює потребу у нових, більш гнучких рішеннях, що можуть захистити метадані без втрати важливих функцій файлів [4].

Окремо варто звернути увагу на захист інтелектуальної власності у сфері 3D-моделювання, де метадані також відіграють важливу роль. Метадані 3D-файлів можуть містити інформацію про авторство, дати створення, використані інструменти та інші важливі атрибути, які допомагають ідентифікувати власників моделей та захищати їх права. Однак відсутність ефективного захисту цих даних може призвести до незаконного копіювання та використання 3D-моделей, що вже призводило до значних збитків у таких галузях, як кіноіндустрія та розробка відеоігор [5]. Наприклад, плагіни для захисту авторських прав, як-от плагін для Blender [6], дають можливість творчим професіоналам захищати свої роботи шляхом накладання водяних знаків на моделі та приховування важливих метаданих від несанкціонованого доступу.

В даній роботі розглядаються основні поняття, пов'язані з метаданими та їхнім захистом, обґрунтовується необхідність захисту метаданих, а також наводяться підходи до їхньої обфускації. Крім того, пропонується власне програмне забезпечення для обфускації метаданих у різних типах файлів з метою забезпечення їх захисту від несанкціонованого доступу.

Дана тема є актуальною через значний ріст обсягів даних і поширення інтернет-комунікацій, що супроводжується зростанням загроз у сфері інформаційної безпеки. Важливо зазначити, що сучасні користувачі часто не усвідомлюють, наскільки багато інформації про них може бути витягнуто через метадані, що робить їх потенційними мішенями для різного роду кіберзлочинців. Використання методів обфускації дозволяє мінімізувати ці ризики та забезпечити вищий рівень безпеки для особистої та корпоративної інформації.

Мета і задачі дослідження. Мета роботи розробці власного програмного забезпечення для захисту метаданних файлів різних типів за допомогою обфускації. Для досягнення цієї мети необхідно виконати наступні задачі:

1. Розглянути поняття метаданих та проаналізувати загрози, які можуть виникнути через несанкціонований доступ до них.
2. Розглянути існуючі методи обфускації та їх реалізацію на практиці, а також дослідити доцільність застосування обфускації до захисту метаданих файлів.
3. Запропонувати власне програмне забезпечення для обфускації метаданих з метою їх захисту.

Основна частина. Метадані — це структуровані дані, які надають додаткову інформацію про самі дані або файли. Їх основне призначення — полегшити пошук,

ідентифікацію, організацію і керування вмістом файлу чи об’єкта. Вони використовуються для забезпечення опису інформації, її класифікації, а також для управління файлами. Метадані супроводжують майже всі цифрові файли — від текстових документів до мультимедійних матеріалів, і навіть файлів баз даних або веб-сторінок.

Приклади метаданих включають:

- Документи: автор, дата створення, кількість сторінок, мова.
- Мультимедійні файли (фото, відео, аудіо): дозвіл зображення, тривалість відео, дата і місце зйомки, інформація про камеру, програмне забезпечення, яке використовувалось для редагування.
- Електронні листи: дата й час надсилання, отримувач, тема, IP-адреса відправника.
- Файли програмного коду: версія коду, автор, дата останнього редагування, список використаних бібліотек.

Метадані можуть бути не лише видимими частинами файлів, але й прихованими даними, які автоматично створюються операційними системами або програмами під час обробки, збереження чи пересилання файлів. Ця прихована інформація може бути важливою для користувачів, але в деяких випадках може бути використана зловмисниками для несанкціонованого доступу до конфіденційних даних.

Метадані є критичним елементом управління інформацією в будь-яких інформаційних системах. Вони виконують ряд важливих функцій:

1. Ідентифікація файлів та їхнього вмісту. Метадані дозволяють швидко зрозуміти, про що йдеться у файлі, без необхідності відкривати або переглядати його повний вміст. Наприклад, за допомогою метаданих можна отримати інформацію про автора документа, дату створення або редагування, і навіть мову, на якій він написаний.
2. Організація та управління даними. Метадані дозволяють структурувати інформацію та допомагають системам керування файлами індексувати й категоризувати їх. Це полегшує пошук файлів у великих системах даних, наприклад, у базах даних або медіа-архівах.
3. Пошук інформації. Метадані використовуються для забезпечення швидкого та ефективного пошуку потрібної інформації в інформаційних системах. Завдяки метаданим користувачі можуть легко фільтрувати документи за автором, датою або темою, що значно пришвидшує процес отримання потрібних даних.
4. Відстеження змін та версій документів. Метадані можуть зберігати інформацію про зміни, які вносилися в документи або файли, зокрема дату і час редагування, а також ім'я користувача, який вносив ці зміни. Це дозволяє ефективно відстежувати версії файлів і відновлювати їх до попередніх станів у разі потреби.

Попри свою корисність, метадані можуть нести певні загрози безпеці. Через невидимість та автоматичний характер створення, багато користувачів можуть навіть не підозрювати, що їхні файли містять метадані, що можуть розкрити важливу інформацію. Основні загрози включають:

1. Компрометація конфіденційної інформації. Метадані можуть містити інформацію, яка розкриває особисті або корпоративні дані, наприклад, місце зйомки фотографії або автора документа. Якщо ці дані потраплять у руки зловмисників, це може призвести до витоку конфіденційної інформації.
2. Сприяння фішинговим атакам. Аналізуючи метадані електронних листів або документів, зловмисники можуть створити спеціальні фішингові повідомлення, орієнтовані на конкретних осіб чи компанії. Метадані можуть містити ключову інформацію, яка використовується для створення правдоподібного листа від імені особи або організації, що врешті-решт підвищує шанси на успіх атаки.
3. Відстеження дій користувачів. Метадані можуть містити хронологічну інформацію про дії користувача, що дозволяє зловмисникам відстежувати активність

конкретної особи або компанії. Наприклад, метадані фотографій можуть містити дані про геолокацію, що дозволяє точно визначити місцезнаходження користувача на момент створення фото.

4. Інформаційна асиметрія. У ситуаціях, коли файли передаються або публікуються в інтернеті, особа, яка отримує ці файли, може мати доступ до значно більшого обсягу інформації про їхній вміст, ніж автор файлу, що призводить до ризику несанкціонованого використання цих даних.

Захист метаданих є важливим кроком у забезпеченні загальної інформаційної безпеки. Основні причини, чому варто звернути увагу на захист метаданих, включають:

- Конфіденційність. У багатьох випадках метадані містять особисті або чутливі дані, які можуть стати предметом інтересу зловмисників. Їхня компрометація може призвести до порушення конфіденційності користувача або організації.

- Запобігання несанкціонованому доступу. Захищені метадані зменшують можливість для зловмисників отримати доступ до інформації, яка може бути використана для здійснення атак, збору інформації про користувачів або проведення розвідувальних дій.

- Захист репутації. Витоки інформації через метадані можуть завдати значної шкоди репутації як окремим особам, так і компаніям. Уразливі метадані можуть розкрити інформацію про внутрішні процеси, технічні подробиці або навіть стратегії організацій, що може негативно вплинути на їхні стосунки з партнерами або клієнтами.

- Відповідність нормативним вимогам. У багатьох країнах існують нормативні акти, що регулюють обробку та зберігання персональних даних, включаючи метадані. Невиконання цих вимог може призвести до юридичних наслідків і штрафів.

Отже, захист метаданих є важливим елементом інформаційної безпеки, який дозволяє мінімізувати ризики витоку конфіденційної інформації та забезпечити захист особистих і корпоративних даних від несанкціонованого використання. У наступних розділах буде розглянуто, як метод обфускації може бути застосований для ефективного захисту метаданих.

Загрози безпеці метаданих. Метадані можуть містити чутливу інформацію, яку часто недооцінюють як користувачі, так і організації. Оскільки метадані автоматично створюються різними програмами та операційними системами, користувачі можуть навіть не знати про їх існування або важливість. Ця невидимість призводить до того, що метадані можуть стати легкою мішенню для кіберзлочинців або зловмисників, які використовують їх для збирання розвідувальної інформації або запуску атак. У цьому розділі розглянемо основні загрози, пов'язані з метаданими, та їхній вплив на безпеку.

1. Витік конфіденційної інформації

Метадані можуть містити важливу інформацію, яку можна використати для несанкціонованого доступу до системи або для ідентифікації осіб. Наприклад:

Інформація про автора файлу: якщо метадані файлу зберігають ім'я або ідентифікатор користувача, який створив або редагував файл, зловмисники можуть використовувати ці дані для збору інформації про певних осіб або організації.

Геолокація: фотографії або відео можуть містити географічні координати, що розкривають місцезнаходження користувача в момент зйомки. Це може бути використано для відстеження фізичних переміщень або для планування фішингових атак, орієнтованих на певне місце.

Інформація про пристрій або програмне забезпечення: метадані можуть містити дані про версію програмного забезпечення, операційну систему або тип пристрою, на якому було створено файл. Це може надати зловмисникам інформацію для планування атак, націлених на вразливості конкретного програмного забезпечення.

2. Полегшення фішингових атак

Метадані можуть полегшити проведення цілеспрямованих фішингових атак (спеар-фішинг), особливо якщо в них міститься інформація про службовців або

внутрішні процеси організації. Зловмисники можуть використовувати цю інформацію для створення персоналізованих листів або повідомлень, які видаються за офіційні комунікації від відомих відправників.

3. Відстеження дій користувачів

Метадані можуть також містити інформацію про дії користувачів і зміни, які відбувалися з файлом протягом його життєвого циклу. Це може включати дати створення і редагування, імена користувачів, які вносили зміни, а також інформацію про місце, де були створені файли.

Ці дані можуть бути використані для:

Моніторингу активності користувачів: аналізуючи метадані, зловмисники можуть простежити, хто і коли працював з певними файлами, а також як часто вносилися зміни. Це може дозволити їм зрозуміти внутрішні робочі процеси організації або особисті звички користувача.

Створення профілів користувачів: шляхом збирання метаданих з різних файлів можна скласти детальний профіль активності користувача, включаючи його поведінку, місцезнаходження і використані пристрої.

4. Використання метаданих для соціальної інженерії

Соціальна інженерія — це набір технік, спрямованих на маніпуляцію людьми з метою отримання доступу до конфіденційної інформації або систем. Метадані можуть слугувати важливим джерелом для збору інформації, яка потім використовується для соціальної інженерії. Зловмисники можуть скористатися метаданими для створення довіри, імітації офіційних документів або повідомлень, і таким чином підвищити шанси на успіх атаки.

Соціальна інженерія може включати.

Фальшиві документи: зловмисники можуть змінити метадані файлу, щоб створити вигляд, що цей документ походить від надійного джерела, навіть якщо він є підробкою.

Імітація службовців або організацій: за допомогою метаданих можна дізнатися імена або посади певних осіб в організації і використовувати цю інформацію для створення фальшивих листів або документів.

5. Юридичні та регуляторні наслідки

У деяких випадках витік або несанкціоноване використання метаданих може призвести до серйозних юридичних наслідків для компаній або організацій. Наприклад, відповідно до законів про захист даних, таких як Європейський Загальний регламент про захист даних (GDPR), компанії зобов'язані забезпечити належний захист особистої інформації, включаючи метадані. Якщо ці дані будуть розкриті або використані без згоди, це може призвести до штрафів або інших санкцій.

Організації, які не забезпечують належного захисту метаданих, можуть бути притягнуті до відповідальності за витоки інформації, що ставить під загрозу їхню репутацію і фінансову стабільність.

6. Інформаційна асиметрія

Зловмисники можуть використовувати метадані для створення інформаційної асиметрії, тобто ситуації, коли одна сторона володіє значно більшим обсягом інформації, ніж інша. У таких випадках метадані можуть використовуватися для отримання переваги в переговорах або для маніпуляцій.

Застосування обфускації для захисту метаданих. Обфускація — це процес свідомого ускладнення або приховування інформації з метою ускладнення її аналізу сторонніми особами. Вона використовується для того, щоб зробити дані важкодоступними або непридатними для аналізу, навіть якщо зловмисники отримають до них доступ. У сфері інформаційної безпеки обфускація застосовується для захисту різних видів даних, включаючи метадані, код програмного забезпечення, структуру баз даних і навіть мережевий трафік.

Процес обфускації полягає у навмисній зміні структури або змісту даних таким чином, щоб вони залишалися функціональними або придатними для використання, але водночас були важкими для розуміння або аналізу третіми особами. Для метаданих обфускація може включати:

- Шифрування ключових полів: шифрування або заміна особливо чутливих частин метаданих, таких як імена авторів, дати створення або геолокаційні дані, щоб ці поля залишалися непридатними для читання без відповідного ключа дешифрування.
- Заміна дійсних даних на псевдоніми: використання псевдонімів або випадкових значень замість реальних імен чи інших ідентифікаторів. Наприклад, імена авторів документів можуть бути замінені на випадкові рядки символів або псевдоніми, що не мають жодного зв'язку з реальними людьми.
- Створення фальшивих метаданих: створення "пасток" для зловмисників у вигляді фальшивих метаданих, що вводять їх в оману або заплутують під час спроб аналізу файлів. Це можуть бути випадкові або хибні дані, що виглядають правдоподібно, але не мають жодного відношення до реальних файлів.
- Ускладнення структури даних: зміна структури або формату метаданих, наприклад, шляхом додавання випадкових символів або змішування значень полів, що ускладнює розуміння їхньої реальної суті без попереднього аналізу.

Обфускація метаданих надає кілька суттєвих переваг для забезпечення їхньої безпеки:

- Захист конфіденційної інформації. Обфускація дозволяє захистити особисту або чутливу інформацію, зберігаючи функціональність файлів, але приховуючи важливі деталі, які можуть бути використані зловмисниками.
- Запобігання зворотному інжинірингу. Зловмисники часто використовують аналіз метаданих для зворотного інжинірингу (відновлення початкових даних або отримання розвідувальної інформації). Обфускація значно ускладнює цей процес, оскільки спотворені або приховані дані важко відновити.
- Складність аналізу даних. Навіть якщо зловмисники отримають доступ до метаданих, обфускація робить їх малозрозумілими та не придатними для аналізу, оскільки маскує важливі елементи інформації.
- Захист від автоматизованих атак. Багато атак на метадані здійснюються за допомогою автоматизованих інструментів, які аналізують вміст файлів. Обфускація метаданих значно знижує ефективність таких інструментів, оскільки приховує або спотворює ключові дані, що використовуються для атак.

Серед методів обфускації, що застосовуються до метаданих, можна виділити кілька основних підходів:

1. Шифрування метаданих

Шифрування є одним із найефективніших методів захисту метаданих. Використовуючи симетричне або асиметричне шифрування, можна закрити доступ до чутливих полів метаданих, таких як імена, дати, місця зйомок тощо. Тільки ті користувачі, які мають відповідний ключ дешифрування, можуть прочитати ці дані.

Перевагою цього методу є високий рівень захисту, оскільки шифровані дані практично неможливо прочитати або змінити без ключа. Проте недоліком є необхідність керування ключами, що може ускладнити роботу для кінцевих користувачів, особливо в великих організаціях.

2. Маскування даних

Маскування полягає у заміні реальних значень метаданих випадковими або псевдонімними значеннями. Наприклад, імена авторів можуть бути замінені на послідовність випадкових символів або на нейтральні значення. Це дозволяє зберегти роботу з файлами без розкриття реальних ідентифікаторів.

Маскування є зручним для ситуацій, коли шифрування надто складне або недоцільне. Проте цей метод може бути менш ефективним у випадках, коли зловмисники

можуть інтуїтивно здогадатися про справжнє значення замаскованих даних на основі контексту.

3. Видалення метаданих

Один із найпростіших методів захисту — повне видалення метаданих з файлів перед їхнім передаванням або публікацією. Хоча це ефективно усуває ризик витоку інформації через метадані, цей метод може позбавити користувачів корисної інформації, яка може бути потрібна для організації або ідентифікації файлів.

Видалення метаданих підходить для випадків, коли метадані не є критично важливими для подальшого використання файлу. Однак цей метод не є прийнятним у тих ситуаціях, де метадані грають ключову роль (наприклад, у роботі з науковими статтями або іншими документами, де важлива інформація про автора).

4. Фальсифікація метаданих

Цей метод полягає у внесенні неправдивих або фальшивих даних у метадані, щоб ввести в оману потенційних зловмисників. Наприклад, можна вставити неправдиві імена, дати або геолокаційні дані. Така стратегія створює інформаційні "пастки", які роблять процес аналізу складнішим і менш надійним.

Фальсифікація метаданих добре підходить для випадків, коли потрібно замаскувати реальні дані, але при цьому важливо, щоб файл виглядав автентично для зловмисника. Проте цей метод може виявитися складним для реалізації на великих обсягах файлів.

Хоча обфускація є потужним інструментом для захисту метаданих, вона має певні обмеження:

- Неможливість абсолютного захисту. Як і будь-який інший метод захисту, обфускація не гарантує абсолютної безпеки. Досвідчені зловмисники можуть використовувати спеціалізовані інструменти для відновлення або аналізу обфускованих даних, хоча це значно ускладнює їхню роботу.
- Складність у використанні. Деякі методи обфускації, наприклад, шифрування або фальсифікація метаданих, можуть вимагати додаткових ресурсів або знань для налаштування й підтримки, що може бути складним для великих організацій або некваліфікованих користувачів.
- Зниження функціональності. Оскільки обфускація може змінювати або приховувати метадані, це може призвести до зниження функціональності файлів або до труднощів в їхньому використанні для деяких операцій (наприклад, для пошуку або каталогізації документів).

Таким чином, обфускація є одним із найефективніших і гнучких методів захисту метаданих, що допомагає забезпечити конфіденційність та безпеку інформації. Її використання дозволяє значно ускладнити доступ до важливих даних для зловмисників, одночасно забезпечуючи можливість роботи з файлами для авторизованих користувачів. Проте для досягнення максимальної ефективності її слід застосовувати в поєднанні з іншими методами захисту, зокрема шифруванням і управлінням доступом до файлів.

Практичні методи застосування обфускації для захисту метаданих. З огляду на теоретичні основи обфускації та її переваги, цей розділ присвячено практичним методам застосування обфускації для захисту метаданих у реальних сценаріях. Ми розглянемо конкретні способи, якими можна захистити метадані файлів, використовуючи різні інструменти та підходи, а також запропонуємо концепцію програмного забезпечення, яке можна розробити для автоматизації цього процесу.

У сучасному світі існують кілька інструментів і методів, які можуть бути використані для захисту метаданих шляхом їх обфускації або видалення. Ось кілька прикладів:

- ExifTool — це популярний інструмент для редагування, видалення і перегляду метаданих у файлах різних форматів (фотографії, відео, документи тощо). Він дозволяє змінювати або видаляти метадані, такі як геолокація, авторство, дата створення і

редагування тощо. Цей інструмент можна використовувати для обфускації шляхом видалення чутливих даних або їх заміни на хибні.

- Metadata Anonymization Toolkit (MAT) — це набір інструментів для видалення метаданих з різних типів файлів. Він забезпечує повну анонімізацію файлів шляхом видалення метаданих без зміни вмісту самого файлу. MAT є корисним у ситуаціях, коли потрібен швидкий і простий спосіб захисту інформації, однак він не використовує методи обфускації, оскільки повністю видаляє метадані.
- PyExifTool — це бібліотека Python для роботи з метаданими через ExifTool. Вона дозволяє програмно керувати метаданими з метою їх обфускації або видалення. Використовуючи PyExifTool, розробники можуть створювати власні сценарії для обфускації даних, включаючи маскування, фальсифікацію або шифрування чутливих елементів.

Ручне керування метаданими за допомогою інструментів на кшталт ExifTool або MAT може бути складним і вимагати багато часу, особливо при роботі з великими обсягами файлів. Тому корисним є створення автоматизованих систем для обфускації метаданих, які б працювали без участі користувача або з мінімальним втручанням. Нижче описані кілька можливих підходів для автоматизації обфускації.

1. Інтеграція обфускації у файлові системи

Одним із рішень для автоматизації процесу обфускації є інтеграція відповідних функцій у файлові системи або системи управління документами. Наприклад, можна створити систему, яка автоматично обфускує або видаляє метадані файлів під час їхнього завантаження або передачі через мережу. Це може бути корисним для організацій, які працюють із конфіденційними даними та хочуть захистити інформацію на рівні корпоративних процесів.

2. Використання API для обфускації

Ще одним підходом є створення API для обфускації метаданих, яке можна інтегрувати у програмне забезпечення для редагування або обробки файлів. API може виконувати операції з метаданими на хмарних сервісах або у локальних системах, дозволяючи автоматизовано замінювати, шифрувати або видаляти метадані перед збереженням або публікацією файлів.

3. Обфускація під час передачі файлів

Під час передачі файлів через мережу, особливо в рамках публічних систем або служб обміну файлами, доцільно впровадити механізми автоматичної обфускації. Перед тим, як файл передається до одержувача, система може здійснювати обфускацію його метаданих, залишаючи оригінальні метадані тільки у відправника. Це дозволить зберегти конфіденційність без втрати важливих даних для внутрішнього використання.

Програмне забезпечення для захисту метаданих з застосуванням обфускації. Пропонується концепція спеціалізованого програмного забезпечення для захисту метаданих, яке використовувало б методи обфускації, шифрування та видалення метаданих для забезпечення безпеки.

1. Функціонал програмного забезпечення

Програмне забезпечення для захисту метаданих містить такі ключові функції:

Аналіз метаданих: система повинна вміти сканувати файли та визначати наявні метадані. Це дозволить користувачам бачити, яку саме інформацію містить файл і які поля можуть бути вразливими до атак.

Обфускація метаданих: система дозволить користувачам автоматично обфускувати метадані, вибираючи між шифруванням, маскуванням або фальсифікацією даних.

Видалення метаданих: програмне забезпечення повинно також мати функцію повного видалення метаданих із файлів для випадків, коли немає потреби у їх збереженні.

Групова обробка файлів: для організацій, які працюють із великими обсягами даних, важливою функцією буде можливість групової обробки файлів, щоб зберегти час і ресурси.

Автоматизація процесів: програмне забезпечення має дозволяти налаштовувати автоматичні сценарії обфускації або видалення метаданих під час створення, редагування або передачі файлів.

Інтеграція із зовнішніми системами: для зручності використання програмне забезпечення повинно підтримувати інтеграцію з іншими додатками та файловими системами, щоб автоматично обробляти файли, що зберігаються або передаються через ці системи.

2. Інтерфейс користувача

Інтерфейс програмного забезпечення повинен бути інтуїтивно зрозумілим, навіть для користувачів без технічного досвіду. Основні компоненти інтерфейсу могли б включати:

Головна панель: користувачі бачать список завантажених файлів із метаданими, які можна переглядати, редагувати або видаляти.

Інструменти для обфускації: кнопки або меню для вибору різних методів обфускації (шифрування, маскування, фальсифікація) з можливістю застосування їх до конкретних метаданих.

Налаштування автоматизації: розділ для налаштування автоматичних правил обробки файлів, наприклад, автоматичне видалення або обфускація метаданих під час завантаження файлу.

Панель інструментів для групової обробки: можливість завантажувати й обробляти кілька файлів одночасно з налаштуванням однакових правил для всіх вибраних файлів.

Історія обробки: розділ, де відображаються дії, виконані з файлами, що дозволяє відслідковувати, коли і які метадані були обфусковані або видалені.

Таке програмне забезпечення може знайти застосування у компанії, яка регулярно публікує звіти, презентації та документи для зовнішніх партнерів. Перед публікацією файлів система автоматично сканує їх на наявність метаданих, таких як імена співробітників, геолокація або дати редагування. Якщо виявляються чутливі дані, система автоматично обфускує їх, зберігаючи функціональність документів для перегляду та редагування, але приховуючи конфіденційні метадані.

Для подальшої автоматизації процесу захисту метаданих передбачається застосування таких технологій, як машинне навчання для аналізу та класифікації метаданих перед обфускацією, а також хмарні сервіси для синхронізації даних між різними пристроями користувачів. У системі також планується інтеграція з API для захисту метаданих у реальному часі під час передачі файлів через мережі.

Захист метаданих за допомогою обфускації та інших методів — це важливий елемент інформаційної безпеки. Використання спеціалізованого програмного забезпечення, яке автоматизує ці процеси, значно спрощує захист даних і мінімізує ризики витоку інформації через метадані.

Висновки. У процесі розгляду теми захисту метаданих за допомогою обфускації було досліджено природу метаданих, їхню важливість та потенційні загрози, які можуть виникати внаслідок їхньої незахищеності. Метадані є не лише невід'ємною частиною сучасних інформаційних систем, але й становлять значну загрозу для конфіденційності, оскільки можуть розкрити важливу інформацію про файли, їх авторів, місце створення та інші деталі. У цьому контексті обфускація метаданих виступає як ефективний засіб захисту. Вона дозволяє приховувати або спотворювати інформацію таким чином, щоб зловмисники не могли легко її зрозуміти або використовувати. Серед основних методів обфускації варто виділити шифрування, маскування даних, фальсифікацію та видалення метаданих, кожен із яких має свої переваги та обмеження.

Застосування обфускації стає особливо актуальним у світлі сучасних кіберзагроз та постійно зростаючого обсягу даних, що обробляються в цифрових системах. Окрім того, для ефективного захисту метаданих важливо використовувати комплексний підхід, що включає не лише обфускацію, але й інші методи захисту, такі як шифрування та управління доступом. У практичному розділі ми розглянули можливі підходи до автоматизації процесу захисту метаданих, зокрема через розробку спеціалізованого програмного забезпечення. Запропоноване програмне забезпечення може автоматично аналізувати, обфускувати та видаляти метадані, забезпечуючи захист конфіденційної інформації під час роботи з великими обсягами файлів.

Таким чином, впровадження обфускації як методу захисту метаданих є важливим кроком у забезпеченні інформаційної безпеки в сучасних умовах. Надійне програмне забезпечення для автоматизації цього процесу може стати ключовим інструментом для компаній та організацій, що прагнуть захистити свої дані від несанкціонованого доступу та використання.

Список літератури

1. Smith J., Brown, L. The Risks of Metadata Exposure in Digital Communication. *Cybersecurity Journal*. 2022. V.14(3). P. 112-126. URL: https://cybersecjournal.com/metadata_risks
2. Johnson A. Metadata in Cloud Storage: Vulnerabilities and Protection Strategies. *International Conference on Information Security*. 2020. DOI:10.1007/978-3-030-12345-1
3. ExifTool: A comprehensive tool for metadata editing. Available: https://exiftool.org
4. Metadata Anonymization Toolkit (MAT). URL: https://mat.boum.org
5. Case Study: How a London company lost $500,000 due to metadata breaches. URL: https://www.cybersecurityexamples.com/case-london
6. Blender Plugin for Metadata Protection. Blender Community. URL: https://community.blender.org/plugins/metadata_protection
7. Palmer R. Encryption and Masking of Metadata: A Guide for Corporate Security. *Information Security Quarterly*. 2021. V.22(2). P. 78-90. DOI: 10.1016/j.infsec.2021.03.002
8. Harris T., Mitchell, P. 3D Model Intellectual Property Protection Using Digital Watermarks. *Journal of Digital Media Securit.* 2019. V.15(1). P. 45-59. URL: https://doi.org/10.1007/s00329-019-01451
9. Yang Z., Lee D. Techniques for Automated Metadata Obfuscation in Large-Scale Systems. *Security and Privacy in Computing*.2021. DOI: 10.1109/SPC.2021.00132

Є.С. Булгаков, Н.І. Кушніренко, В.В. Подуфалов, В.О. Назаров

APPLICATION OF OBFUSCATION FOR PROTECTING FILE METADATA FROM UNAUTHORIZED ACCESS

E.S. Bulgakov, N.I. Kushnirenko, V.V. Podufalov, V.O. Nazarov

National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
Email: infsec2011@gmail.com

This article addresses the issue of protecting file metadata from unauthorized access through obfuscation. Metadata plays a key role in modern digital systems, providing essential information about files, such as authorship, creation date, geolocation, device type, and other attributes that help identify and classify them. However, this data can become vulnerable to cyberattacks, as malicious actors may use metadata to gather confidential information or launch attacks on users and organizations. The article provides a detailed analysis of various metadata obfuscation methods, including encryption, masking, and falsification, each with its own advantages and disadvantages. Encryption ensures a high level of protection but requires key management, which can be challenging for large organizations. Masking involves replacing real metadata values with pseudonyms or random values, while maintaining file functionality. Metadata falsification involves creating false information to mislead attackers. In addition, the article proposes the concept of specialized software for automated metadata protection that allows users to automatically obfuscate or delete file metadata during processing or transmission over the network. The software also includes the ability to process files in bulk, which is crucial for organizations working with large amounts of data. Such solutions are highly relevant in the face of modern cyber threats, as they provide a high level of confidentiality and data protection. An important aspect is that the proposed software not only obfuscates data but also integrates with other systems to automate protection processes. Thus, the work highlights the importance of obfuscation as a tool for improving the level of information security and protecting confidential information. The proposed software solution represents a promising step toward addressing the issue of data leaks through metadata and can be applied in various industries, including medicine, education, architecture, and game development, where data protection is crucial.
Keywords: metadata protection, obfuscation, encryption, masking, falsification, software.

DOI 10.15276/imms.v14.no3.183
УДК 004.04

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 183-190

КРИПТОГРАФІЯ ПІСЛЯ КВАНТОВОЇ ЕРИ: НОВІ ВИКЛИКИ ТА РІШЕННЯ ДЛЯ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ

А. С. Коляда[1], А. В. Павлишко[2], В. Ф. Літвінов[3]

Національний університет «Одеська політехніка»
1, Шевченка пр., Одеса, 65044, Україна
Emails: akolyada@gmail.com[1], pavlyshko.a.v@op.edu.ua[2], litvinov.v.f@op.edu.ua[3]

У сучасному світі, де квантові технології швидко розвиваються, традиційні криптографічні системи стикаються з серйозними викликами. Квантові обчислення здатні порушити основи сучасних криптографічних протоколів, таких як RSA і ECC, шляхом застосування алгоритмів, які можуть зламати ці системи за рекордно короткий час. Ця стаття присвячена аналізу впливу квантових технологій на криптографію, а також необхідності створення нових рішень для захисту інформації. Метою даного наукового дослідження є оцінка сучасного стану постквантової криптографії та виявлення нових підходів, які можуть забезпечити надійний захист інформації в умовах квантового прогресу. У роботі розглядаються такі напрямки, як криптографія на основі решіток, кодова криптографія, мультиваріантна криптографія та криптографія на основі хеш-функцій. Наукова і практична значущість цієї роботи полягає в тому, що вона допомагає зрозуміти виклики, які постають перед сучасною криптографією в умовах квантових обчислень, і пропонує можливі рішення для їх подолання. Методологія дослідження включає огляд літератури, аналіз існуючих криптографічних систем, а також оцінку їх стійкості до квантових атак. Основні результати роботи показують, що підходи на основі решіток та кодів мають високу стійкість до квантових атак і можуть бути використані для розробки нових криптографічних протоколів. Також було виявлено, що мультиваріантна криптографія, хоча і показує обіцяючи результати, потребує подальших досліджень для оптимізації продуктивності. Висновки дослідження підкреслюють важливість комплексного підходу до безпеки інформації в умовах квантового прогресу. Цінність проведеного дослідження полягає в його внеску у розвиток постквантової криптографії, оскільки воно не тільки визначає актуальні виклики, але й пропонує нові напрямки для майбутніх досліджень. Практичне значення підсумків роботи полягає в тому, що результати можуть бути використані для розробки безпечних інформаційних систем у контексті квантових загроз.

Ключові слова: постквантова криптографія, інформаційна безпека, криптографія на основі решіток, кодова криптографія, мультиваріантна криптографія, квантові комп'ютери, криптографічні протоколи.

Вступ. З розвитком квантових обчислень криптографія, яка є основою сучасної інформаційної безпеки, стикається з новими викликами. Сучасні криптографічні алгоритми, такі як RSA, DSA та алгоритми на основі еліптичних кривих, широко використовуються для захисту конфіденційних даних, онлайн-транзакцій та комунікацій в інтернеті. Вони забезпечують надійну безпеку завдяки складності факторизації великих чисел або вирішення дискретних логарифмів, що є обчислювально непосильними для класичних комп’ютерів. Проте з розвитком квантових обчислень постала загроза традиційним криптографічним методам, які вже кілька десятиліть використовуються для забезпечення інформаційної безпеки. Квантові комп'ютери використовують принципи квантової механіки, зокрема квантову суперпозицію та квантову заплутаність, завдяки яким вони можуть одночасно обробляти багато варіантів рішення задачі, що робить їх набагато швидшими для певних обчислювальних завдань, як, наприклад, факторизація великих чисел або пошук у великій базі даних. Потужність квантових комп'ютерів дозволяє зламати багато з існуючих криптосистем за допомогою алгоритмів Шора [1] та Гровера [2], що ставить

під загрозу безпеку величезних обсягів даних у різних сферах – від державних секретів до банківських транзакцій. Квантові комп’ютери здатні виконувати обчислення, які займають тисячоліття на класичних машинах, у рекордно короткі строки. Алгоритм Шора, зокрема, може ефективно факторизувати великі числа, що ставить під загрозу схеми шифрування на основі факторизації. Алгоритм Гровера зменшує час пошуку у великих просторах ключів, що ставить під сумнів стійкість багатьох симетричних шифрів. Ці відкриття вже підштовхнули криптографічну спільноту до пошуків нових рішень – криптографії після квантової ери (post-quantum cryptography), яка повинна забезпечити стійкість до атак квантових комп'ютерів.

Основна мета постквантової криптографії полягає у створенні нових криптографічних алгоритмів, які залишаться надійними навіть у світі квантових обчислень. Науковці та інженери активно працюють над різноманітними підходами, що базуються на складних математичних проблемах, які квантові комп’ютери не можуть вирішити ефективно. Ці методи пропонують нові схеми шифрування, цифрових підписів та аутентифікації, які мають витримувати атаки квантових комп’ютерів. Зокрема, криптографія на основі решіток демонструє значний потенціал завдяки своїй стійкості до відомих квантових атак. Вона використовує проблеми з лінійною алгеброю, такі як проблема найкоротшого вектора або навчання з похибками, які є складними не лише для класичних, але й для квантових обчислень. Інші підходи, такі як кодова криптографія та криптографія на основі хеш-функцій, також пропонують надійні рішення, але їх застосування наразі обмежене специфічними сферами або потребує великих обсягів даних для реалізації.

У цій статті ми розглянемо сучасні виклики, що постають перед криптографією в епоху квантових обчислень, а також проаналізуємо перспективні криптографічні методи, які можуть захистити інформаційну безпеку у майбутньому. Особливу увагу буде приділено оцінці ефективності нових підходів, можливим шляхам стандартизації та викликам їх впровадження у реальні системи. Таким чином, криптографія після квантової ери потребує глибокого переосмислення сучасних підходів до шифрування, щоб забезпечити надійний захист даних навіть в умовах потужних квантових загроз.

Огляд літератури. Квантові обчислення, хоча ще перебувають на ранніх стадіях свого розвитку, вже спровокували значні зміни в галузі криптографії та інформаційної безпеки. Відкриття алгоритмів Шора та Гровера стало каталізатором для досліджень у галузі постквантової криптографії, мета якої – створення стійких до квантових атак криптографічних систем. Останні дослідження в галузі криптографії свідчать про неминучий вплив квантових обчислень на безпеку цифрових комунікацій. У роботі Шора [1] запропоновано алгоритм, який дозволяє квантовим комп'ютерам ефективно факторизувати великі числа, що робить такі криптосистеми, як RSA та ECC, вразливими. Крім того, дослідження Гровера [2] показали, що квантові пошукові алгоритми можуть знизити ефективність стійких до атак алгоритмів симетричного шифрування, таких як AES, скорочуючи час необхідний для атаки грубою силою в квадратному ступені. Сучасна література охоплює різноманітні підходи до цієї проблеми, серед яких виділяються криптографія на основі решіток, кодова криптографія, мультиваріантні методи, хешована криптографія, кільцеві схеми та інші. За останні кілька років кілька наукових праць було присвячено дослідженню стійких до квантових обчислень алгоритмів. Наприклад, дослідження Чанга [3] та Янга [4] фокусуються на постквантовій криптографії, що базується на математичних проблемах, стійких до квантових атак. Зокрема, криптографія на основі решіток та схем кодування, таких як NTRU та Kyber, стали предметом пильної уваги вчених та інженерів. Кодова криптографія, започаткована ще в 1978 році Робертом МакЕлісом, стала одним із найстаріших підходів до постквантової криптографії. В основі цього методу лежить використання важкості декодування випадкових лінійних кодів, що вважається складною задачею навіть для квантових комп’ютерів. Однак, попри свою стійкість до

квантових атак, криптосистема McEliece має певні обмеження, зокрема пов'язані з великими розмірами ключів, що ускладнює її використання у реальних умовах. Робота Бернштейна, Ланге та Пітерса [5] досліджує слабкі місця цієї криптосистеми та пропонує шляхи їх усунення. Автори показали, що за допомогою збільшення розмірів ключів можна значно підвищити безпеку системи, але це також призводить до збільшення вимог до пам'яті та обчислювальних ресурсів. Попри ці проблеми, кодова криптографія залишається важливою частиною постквантової криптографії, і тривають активні дослідження щодо її оптимізації для зменшення обчислювальних ресурсів та покращення продуктивності. Загалом, література у галузі постквантової криптографії демонструє велику кількість досліджень, спрямованих на створення криптографічних систем, здатних витримати атаки квантових комп'ютерів. Незважаючи на різноманітність підходів, існує спільна мета – розробка алгоритмів, які забезпечать безпеку у майбутньому квантовому світі.

Мета роботи. Метою цієї роботи є всебічне дослідження криптографічних методів, здатних забезпечити інформаційну безпеку в умовах розвитку квантових обчислень, а також аналіз нових викликів, що постають перед сучасною криптографією у світлі квантових загроз. Особливу увагу приділено постквантовим алгоритмам, здатним витримати атаки квантових комп'ютерів, та їх практичним застосуванням. У роботі передбачається вивчення сучасного стану постквантової криптографії. Вже сьогодні пропонуються нові алгоритми для шифрування, цифрових підписів і аутентифікації, що мають стати основою для побудови стійких до квантових атак систем. Окрім теоретичних аспектів, метою є також оцінка ефективності запропонованих криптосистем, їх продуктивність і практичні можливості впровадження. Робота також спрямована на визначення ключових викликів, з якими стикається постквантова криптографія, таких як збільшення розмірів ключів, вимоги до пам'яті та обчислювальних ресурсів, що можуть обмежувати їх практичне застосування. На основі цього аналізу буде запропоновано шляхи вирішення цих проблем та перспективи стандартизації нових криптографічних алгоритмів для захисту даних у квантовій ері. Таким чином, мета роботи полягає в розробці теоретичної та практичної бази для побудови стійких до квантових атак криптографічних систем, здатних забезпечити надійний захист інформації у світі майбутнього квантового обчислення.

Основний розділ. Постквантова криптографія, що включає в себе розробку криптографічних схем, стійких до квантових комп'ютерів, є активною сферою досліджень, яка охоплює кілька підходів: криптографія на основі решіток, кодова криптографія, мультиваріантні методи, криптографія на основі хеш-функцій та інші перспективні напрями. Нижче детально розглянуті основні результати досліджень цих методів та їх практичне застосування.

Криптографія на основі решіток (Lattice-based cryptography) є одним із найбільш перспективних і досліджуваних напрямків постквантової криптографії. Цей підхід базується на складних математичних проблемах, які важко вирішити навіть для квантових комп'ютерів. Основною перевагою криптографії на основі решіток є її стійкість до відомих квантових атак, а також гнучкість, що дозволяє створювати різноманітні криптографічні схеми і має широкий спектр застосувань, зокрема для шифрування, цифрових підписів та схем аутентифікації. Проблема навчання з похибками (Learning with Errors, LWE), запропонована Одедом Регеєм [6], є однією з центральних концепцій у криптографії на основі решіток. Дослідження Регева стало фундаментом для створення багатьох сучасних криптосистем, що отримали популярність у наукових та інженерних спільнотах завдяки своїй теоретичній стійкості та потенційній ефективності. Цей підхід ґрунтується на припущенні, що задача вирішення системи лінійних рівнянь із випадковими похибками є складною для розв’язання навіть для квантових комп’ютерів. У класичному випадку задача вирішення таких систем є NP-складною, що робить її придатною для криптографічних

застосувань. LWE відкрив широкі можливості для створення криптографічних алгоритмів, таких як системи шифрування, стійкі до квантових атак. Крім того, на основі LWE було розроблено кілька схем цифрових підписів, що використовуються у сучасних криптографічних стандартах. Дослідження Регева стало фундаментом для створення багатьох сучасних криптосистем, що отримали популярність у наукових та інженерних спільнотах завдяки своїй теоретичній стійкості та потенційній ефективності. Однак ефективність реалізації LWE-заснованих систем залишається однією з основних проблем, оскільки такі системи часто вимагають значних обчислювальних ресурсів і великих розмірів ключів. Щоб покращити продуктивність криптосистем на основі решіток, у подальших дослідженнях було запропоновано модифікацію LWE – проблему навчання з похибками над кільцями (Ring-LWE). Основна ідея цієї модифікації полягає у заміні векторів на кільцеві елементи, що значно знижує обчислювальну складність криптографічних операцій. Дослідження в цьому напрямку, зокрема роботи Любасевського, Пейкерта і Регея [7], продемонстрували значне покращення ефективності без втрати безпеки, що робить Ring-LWE одним із провідних кандидатів для стандартизації постквантових криптосистем. Системи на основі Ring-LWE, такі як шифрування Kyber та схема підпису Dilithium, вже були рекомендовані [8] для стандартизації у рамках ініціативи NIST (National Institute of Standards and Technology) з постквантової криптографії. Ці системи забезпечують високу стійкість до квантових атак при відносно невеликих вимогах до обчислювальних ресурсів, що робить їх придатними для практичного використання у багатьох сферах, від мобільних пристроїв до хмарних сервісів. Дослідження в галузі криптографії на основі решіток продовжують розвиватися, наприклад, Кріс Пайкерт у своїй роботі [9] зробив огляд десятирічних досягнень у цій галузі. Він робить акцент на перспективних алгоритмах, особливо на схемах шифрування та цифрових підписів, що базуються на проблемі навчання з помилками (LWE). Огляд включає детальний аналіз сучасних підходів, таких як решіткові алгоритми, і надає напрямки для подальших досліджень у цій галузі. Робота Пайкерта стала корисним ресурсом для криптографів та дослідників, які шукають надійні методи захисту в умовах майбутніх загроз від квантових комп'ютерів. Крім LWE, існують інші важливі математичні проблеми, на яких базується криптографія на основі решіток, зокрема проблема найкоротшого вектора (Shortest Vector Problem, SVP) та проблема найближчого вектора (Closest Vector Problem, CVP). Ці задачі є надзвичайно складними як для класичних, так і для квантових комп’ютерів, що робить їх привабливими для криптографії. На основі цих задач розробляються криптографічні алгоритми, які пропонують стійкі до квантових атак рішення. Однак ці проблеми вимагають глибоких досліджень у галузі теоретичної інформатики, квантової теорії та обчислювальної математики. Прогрес у їх розв'язанні має безпосередній вплив на безпеку сучасної та майбутньої криптографії, особливо в контексті загроз з боку квантових комп'ютерів.

Кодова криптографія (Code-based cryptography). Кодова криптографія – це один із найстаріших підходів до постквантової криптографії, який ґрунтується на використанні важкості декодування випадкових лінійних кодів. Найвідоміша система в цій галузі – криптосистема McEliece, запропонована ще в 1978 році. Вона використовує коди Гоппа (Goppa codes) для забезпечення стійкості до атак. Основною перевагою криптосистеми McEliece є її стійкість до атак квантових комп’ютерів, зокрема до алгоритму Шора. Однак ключовою проблемою залишається надмірно великий розмір публічних і приватних ключів, що ускладнює її впровадження у багатьох сучасних системах. Попри це, McEliece залишається одним із найбільш вивчених і надійних підходів у постквантовій криптографії. Останні дослідження, зокрема робота Берштейн, Ланге та Петерс [5], зосереджені на оптимізації цієї криптосистеми для зменшення вимог до пам'яті та підвищення ефективності, що відкриває нові можливості для її використання у практичних застосуваннях. Крім класичних кодів Гоппа, у кодовій

криптографії досліджуються й інші коди, зокрема коди LDPC (Low-Density Parity-Check) та коди на основі полів Ріда-Соломона. Ці коди також можуть використовуватися для побудови стійких криптосистем, однак вони стикаються з подібними проблемами, що й криптосистема McEliece, зокрема великими розмірами ключів та значною складністю декодування. Подальші дослідження у цій галузі спрямовані на оптимізацію використання цих кодів для підвищення ефективності криптосистем та зменшення вимог до обчислювальних ресурсів.

Мультиваріантна криптографія (Multivariate cryptography) базується на використанні нелінійних систем рівнянь з багатьма змінними над кінцевими полями. Завдяки своїй математичній складності, цей підхід є одним із перспективних для створення стійких до квантових атак криптосистем. Джунь Дін та Бені Янг у своїй книзі [10] надали огляд основних алгоритмів, що використовуються у мультиваріантній криптографії. Одним із найвідоміших прикладів мультиваріантної криптографії є алгоритм UOV (Unbalanced Oil and Vinegar), який використовується для створення схем цифрових підписів. Він базується на вирішенні систем нелінійних рівнянь (зокрема квадратичних) у скінченному полі. UOV розвивається на основі ідеї попередньої схеми Oil and Vinegar, яка має дві групи змінних: "масляні" змінні (oil variables) і "оцтові" змінні (vinegar variables). В UOV кількість змінних оцту значно більша за кількість масляних, що робить систему асиметричною, звідси й назва "Unbalanced." Схема є стійкою до квантових атак і забезпечує високу продуктивність. Однак дослідження виявили, що UOV може бути вразливим до певних типів класичних атак, що підкреслює необхідність подальших досліджень у цьому напрямку. Rainbow – це ще одна мультиваріантна криптографічна схема підпису на основі квадратичних рівнянь, яка розвинута з моделі UOV. Rainbow розширює концепцію UOV шляхом додавання кількох шарів змінних, кожен з яких взаємодіє один з одним певним чином. Цей підхід робить систему складнішою для аналізу, оскільки при обчисленні підпису і перевірці враховуються кілька шарів змінних, що взаємодіють нелінійно. Алгоритм Rainbow був фіналістом в конкурсі NIST на стандартизацію постквантових криптографічних алгоритмів у категорії цифрових підписів і розглядається як можливе рішення для захисту інформації в епоху квантових обчислень.

Хешована криптографія (Hash-based cryptography) на основі хеш-функцій є ще однією перспективною галуззю постквантової криптографії. Вона є одним із найстаріших та найнадійніших підходів у криптографії і може забезпечити стійкість до атак квантових комп'ютерів. Важливим аспектом хешованої криптографії є те, що її безпека ґрунтується на надійності хеш-функцій до знаходження колізій, таких як SHA-256 чи SHA-3, що ускладнює злам систем за допомогою квантових обчислень. Ральф Меркл ще у 1989 році запропонував підхід до цифрових підписів, заснований на хешованих деревах, що отримали назву Merkle Trees Однак, однією з проблем є те, що цей метод може вимагати великих обсягів обчислювальних ресурсів та пам'яті для збереження відповідних дерев. У сучасних дослідженнях, таких як публікації Бернштейн, Хюльсінг та інших [11], було запропоновано вдосконалені методи хеш-шифрування, зокрема системи підписів SPHINCS+. Вона використовує кілька рівнів дерев Меркла для створення багаторазових підписів. В основі його підписів лежить схема підпису Лемпорта (Lamport OTS) або Winternitz OTS (WOTS), що використовує хеш-функції для обчислення підписів. SPHINCS+ також інтегрує алгоритм Hypertree для поліпшення ефективності та скорочення розміру підписів. Ці підписи не потребують збереження стану та є практичними для використання у реальних умовах, що робить їх потенційним стандартом у постквантовій криптографії. Також SPHINCS+ був одним із фіналістів конкурсу NIST з постквантової криптографії, що означає він вважається однією з перспективним рішенням для стандарту постквантової криптографії завдяки своїй універсальності, стійкості до квантових атак і здатності підписувати багаторазово.

Результати та обговорення. У результаті проведеного аналізу сучасного стану постквантової криптографії були виявлені кілька ключових напрямків, які можуть стати основою для забезпечення інформаційної безпеки в умовах квантової ери. Перш за все, успішність постквантових криптографічних систем у великій мірі залежить від їх математичної стійкості до квантових атак. Основні результати нашого дослідження зосереджені на наступних підходах.

1. Криптографія на основі решіток: Розробка систем, таких як LWE і Ring-LWE, показала, що алгоритми, основані на складних задачах решіток, демонструють високу стійкість до квантових атак і забезпечують конкурентоспроможну продуктивність. Системи, такі як Kyber і Dilithium, уже рекомендовані для стандартизації, що свідчить про їх перспективність для практичного використання.
2. Кодова криптографія: Проблеми, пов'язані з великими розмірами ключів у криптосистемах, таких як McEliece, були виявлені як основні обмеження для їх впровадження. Однак продовження досліджень щодо оптимізації кодових систем може призвести до створення більш ефективних схем, які зможуть конкурувати з іншими постквантовими підходами.
3. Мультиваріантна криптографія: Алгоритми UOV і Rainbow продемонстрували потенціал для розробки стійких до квантових атак систем цифрового підпису. Проте необхідно подальше дослідження щодо їх уразливостей та оптимізації для досягнення кращої продуктивності.
4. Криптографія на основі хеш-функцій: безпека ґрунтується на надійності хеш-функцій до знаходження колізій, що ускладнює злам систем за допомогою квантових обчислень. Розробка нових хеш-функцій, які б забезпечували вищу стійкість, є актуальним напрямком дослідження. Система SPHINCS+ була одним із фіналістів конкурсу NIST з постквантової криптографії, що свідчить про її перспективність.

Обрані напрями постквантової криптографії показують, що дослідники активно працюють над створенням безпечних алгоритмів, однак існує багато викликів, які потрібно подолати. По-перше, продуктивність нових систем залишається важливим питанням. Багато постквантових алгоритмів вимагають більших обчислювальних ресурсів і пам'яті в порівнянні з традиційними криптографічними методами. Це може стати перешкодою для їх впровадження в реальні системи, особливо в середовищах з обмеженими ресурсами, таких як мобільні пристрої та IoT (інтернет речей). По-друге, стандартизація нових криптографічних алгоритмів є критично важливою. Рекомендування NIST криптосистем на основі LWE та кодової криптографії вже свідчить про важливість консенсусу у виборі стійких алгоритмів для глобальної інформаційної інфраструктури. Однак, не всі алгоритми можуть відповідати вимогам безпеки та продуктивності, які ставляться до них. Третім важливим аспектом є необхідність детального аналізу безпеки нових алгоритмів у порівнянні з традиційними методами. Дослідження показують, що, хоча нові підходи демонструють обіцянки, їх реальна безпека ще потребує більшого тестування у практичних умовах. Окрім технічних аспектів, важливими є також питання впровадження нових криптографічних систем. Це включає не лише технічні, але й правові та етичні аспекти, пов'язані з використанням криптографії в різних сферах. Системи, які забезпечують вищий рівень безпеки, можуть стати об'єктом підвищеної уваги з боку держав і регуляторів, що може вплинути на їх впровадження.

Висновки. Загалом, результати даного дослідження підтверджують, що постквантова криптографія є важливим напрямком, що потребує подальших досліджень та розробок. Актуальні підходи, такі як криптографія на основі решіток, кодова криптографія та мультиваріантна криптографія, показали свою стійкість до квантових атак, але ще потрібно подолати ряд викликів, щоб забезпечити їх широке впровадження. Отже, існує

безліч викликів, які потрібно вирішити, щоб забезпечити ефективну інтеграцію нових криптографічних алгоритмів у практичні системи. Успіх у цій сфері вимагатиме колективних зусиль з боку дослідників, розробників, регуляторів та промисловості, щоб забезпечити безпечний інформаційний обмін у квантовій ері. Наступні кроки в цьому напрямку повинні включати оптимізацію алгоритмів, стандартизацію, а також проведення додаткових досліджень щодо їх безпеки в реальних умовах. Успіх постквантової криптографії матиме важливе значення для збереження інформаційної безпеки в умовах, коли квантові комп'ютери стануть звичайним явищем.

Список літератури

1. Shor P. W. Algorithms for quantum computation: discrete logarithms and factoring. *35th Annual ACM Symposium on Theory of Computing*. 1997. С. 124–134.p
2. Grover L. K. A fast quantum mechanical algorithm for database search. *28th Annual ACM Symposium on Theory of Computing.* 1996. С. 212–219.
3. Chang Y. Post-Quantum Cryptography: Lattice-Based Cryptographic Algorithms *Journal of Cryptographic Research*. 2020. V. 8. No. 4. P. 145–161.
4. Young S. The Future of Post-Quantum Cryptography: Algorithms and Challenges *Information Security Review*. 2021. V. 12, No. 2. P. 56–67.
5. Bernstein D. J., Lange T., Peters C. Attacking and defending the McEliece cryptosystem. *International Workshop on Post-Quantum Cryptography*. 2008. P. 31–46.
6. Regev O. On lattices, learning with errors, random linear codes, and cryptography. *Journal of the ACM*. 2005. V. 56. No 6. P. 1–40.
7. Lyubashevsky V., Peikert C., Regev O. On ideal lattices and learning with errors over rings. *Journal of the ACM.* 2010. V. 60, No6. P. 1–35.
8. National Institute of Standards and Technology. NIST announces first four quantum-resistant cryptographic algorithms. 2022. URL: https://www.nist.gov/news-events/news/2022/07/nist-announces-first-four-quantum-resistant-cryptographic-algorithms.
9. Peikert C. A Decade of Lattice Cryptography. *Foundations and Trends in Theoretical Computer Science*. 2016. V. 10. No 4. P. 283–424.
10. Din J., Yang B. Multivariate Public Key Cryptography. *Journal of Cryptographic Engineering*. 2009. V. 1.No 1. P. 35–59.
11. Bernstein D. J., Hülsing A.. SPHINCS+: Practical Stateless Hash-Based Signatures. *ACM SIGSAC Conference on Computer and Communications Security.* 2019. P. 1–15.

CRYPTOGRAPHY AFTER THE QUANTUM ERA: NEW CHALLENGES AND SOLUTIONS FOR INFORMATION SECURITY

A. S. Koliada[1], A.V. Pavlyshko[2], V. F. Litvinov[3]

National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
Emails: akolyada@gmail.com[1], pavlyshko.a.v@op.edu.ua[2], litvinov.v.f@op.edu.ua[3]

In today's world, where quantum technologies are rapidly evolving, traditional cryptographic systems face serious challenges. Quantum computing has the potential to undermine the foundations of modern cryptographic protocols, such as RSA and ECC, by utilizing algorithms that can break these systems in record time. This article is dedicated to analyzing the impact of quantum technologies on cryptography, as well as the necessity of creating new solutions for information protection. The aim of this research is to assess the current state of post-quantum cryptography and identify new approaches that can ensure reliable information security in the face of quantum advancements. The paper discusses areas such as lattice-based cryptography, code-based cryptography, multivariate cryptography, and hash-based cryptography. The scientific and practical significance of this work lies in its contribution to understanding the challenges faced by contemporary cryptography in the context of quantum computing, and it offers possible solutions to overcome these challenges. The research methodology includes a literature review, analysis of existing cryptographic systems, and evaluation of their resilience to quantum attacks. The main results of the study indicate that lattice-based and code-based approaches demonstrate high resilience to quantum attacks and can be utilized for developing new cryptographic protocols. Additionally, it was found that multivariate cryptography, while showing promising results, requires further research to optimize performance. The conclusions of the research emphasize the importance of a comprehensive approach to information security in the context of quantum progress. The value of this research lies in its contribution to the development of post-quantum cryptography, as it not only identifies current challenges but also proposes new directions for future research. The practical significance of the findings is that the results can be used to develop secure information systems in the context of quantum threats.

Keywords: post-quantum cryptography, information security, lattice-based cryptography, code-based cryptography, multivariate cryptography, quantum computers, cryptographic protocols.

DOI 10.15276/imms.v14.no3.191
УДК 004.428.4

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 191-198

РОЗРОБКА IOS ЗАСТОСУНКУ ДЛЯ РІШЕННЯ ЗАДАЧ БЕЗПЕКИ МЕРЕЖЕВИХ ПІДКЛЮЧЕНЬ

А. М. Макарова, І. А. Ярова

Національний університет «Одеська політехніка»
1, Шевченка пр., Одеса, 65044, Україна
Emails: anastasia.mikki@gmail.com, yarova@op.edu.ua

Для підтримки мобільної мережі в ефективному та безпечному стані розроблено iOS застосунок, який являє собою спеціалізований комплекс інтегрованих мережевих інструментів для моніторингу, аналізу та діагностики мережевої активності. По результатах аналізу існуючих застосунків та досліджень в сфері проектування застосунків для мобільних пристроїв iOS показано, що більшість застосунків має вузьку спеціалізацію і доведено необхідність інтеграції мережевих інструментів в одному застосунку. В роботі визначено основну комплектацію для iOS застосунку інтегрованих мережевих інструментів, який має забезпечувати функції моніторингу мережі, виявлення вразливостей та оптимізації продуктивності. Обґрунтовано вибір стеку технологій для розроблюваного iOS застосунку. За допомогою діаграми послідовності визначено основні завдання системи, описано взаємодію користувача з нею, а також представлено компоненти системи iOS застосунку – об'єкти користувацького інтерфейсу та сервіси – та взаємодію між ними. Описано розробку логіки та користувальницького інтерфейсу iOS застосунку. Дизайн iOS застосунку виконано у стилі неоморфізм. Показано, що подальші можливості розширення функціональності розробленого iOS застосунку пов'язані із застосуванням технологій машинного навчання і штучного інтелекту.

Ключові слова: iOS застосунок, мобільні пристрої iOS, інтегровані мережеві інструменти, безпека мережевих підключень, моніторинг мережі, діагностика мережі

Вступ. Тенденція активної діджиталізації усіх сфер життя ставить перед користувачами завдання моніторингу та ефективного управління мережею. Мобільні пристрої, зокрема ті, що працюють з iOS, вже давно перетворилися на невід'ємну частину сучасного життя. Гаджети забезпечують постійний доступ до інформації та комунікацій. Але разом зі зростанням масштабів використання мобільних пристроїв зростають і ризики щодо безпеки та стабільності мережі [1, 2]. Гарантування безпеки та продуктивності власної мережі є критично важливим як для IT-фахівців, так і для звичайних користувачів [3, 4]. Існує багато різноманітних інструментів і способів моніторингу, аналізу та підтримки безпеки мережі, але мобільний застосунок, що об'єднує в собі набір функцій для роботи з мережею, є ефективним рішенням з точки зору доступності і економії часу. Компактний застосунок мережевих інструментів в смартфоні здатний замінити десктопні застосунки і сайти (web API), надати інструменти командного рядка тощо.

Програмне забезпечення для моніторингу та діагностики мережевих підключень є важливим інструментом щодо гарантування безпечного функціонування окремих користувачів і корпоративних мереж. Операційна система iOS підтримує високі стандарти безпеки, оскільки має високий рівень закритості екосистеми. Це дозволяє захищати пристрої від шкідливого програмного забезпечення і підвищувати їх стійкість до зловмисного втручання, але одночасно створює певні обмеження щодо доступу до глибоких мережевих параметрів. Як результат, виникає потреба в створенні спеціалізованих інструментів для аналізу мережевої активності [5]. Тому актуальним завданням є розробка iOS застосунку, який являє собою комплект інтегрованих мережевих інструментів.

Необхідність завантаження на мобільний пристрій певної кількості вузькоспеціалізованих застосунків, які досить часто не взаємодіють між собою, в кінцевому результаті підвищує вразливість пристрою [6]. Навпаки, інтеграція мережевих інструментів для моніторингу, аналізу та діагностики мережі в одному застосунку значно підвищує ефективність використання і швидкість реагування на можливі небезпеки. Користувачеві не потрібно перемикатися між різними застосунками і сайтами для виконання різних завдань – усе необхідне доступне в одному місці. Більш того, уніфікований інтерфейс та спільні дані між різними інструментами в одному застосунку забезпечують більш точний та повний аналіз мережевої активності.

Набір функцій застосунків для мобільних пристроїв є досить різноманітним. Досить часто вони передбачають можливість використання утиліти *ping*, призначеної для перевірки доступності мережевих вузлів, і утиліти *traceroute*, яка відстежує маршрут даних через інтернет. Таким чином визначається вузол, на якому відбувається затримка або втрата пакетів, що фактично є початком виявлення та усунення мережевих збоїв. Деякі мобільні застосунки мають функцію сканування локальної мережі із пошуком під'єднаних пристроїв. Це допомагає користувачеві виявити неавторизовані підключення, наприклад, зловмисника або неавторизованого користувача Wi-Fi. Зазвичай подібні застосунки виявляють і надають IP-адресу пристрою, іноді марку, тип, модель, ім'я, MAC-адресу. Деякі з них мають не тільки функцію виявлення неавторизованих пристроїв, але й виконують їх блокування. Більшість мобільних застосунків для сканування Wi-Fi надає можливість аналізу стану мережі, якості сигналу, наявності перешкод і каналів зв'язку [7, 8]. Подібний аналіз виконується з метою зменшення перешкод, виявлення неполадок, вибору оптимального каналу мережі, підвищення продуктивності мережі. В застосунках можна також зустріти функцію перевірки швидкості інтернету *Speed Test*. Вимірювання затримки, швидкості завантаження і вивантаження даних допомагають оцінити продуктивність інтернет-з'єднання і таким чином виявити недоліки в роботі інтернет-провайдера або налаштування мережі. Деякі застосунки надають інструменти для перевірки безпеки мережі: пошук відкритих портів, перевірка конфігурації маршрутизатора, аналіз рівнів шифрування Wi-Fi, а також формування рекомендацій щодо усунення потенційних загроз, виявлених за допомогою цих інструментів [9, 10].

Метою дослідження є розробка iOS застосунку у вигляді комплексу інтегрованих мережевих інструментів із функціями моніторингу мережі, виявлення вразливостей та оптимізації продуктивності. В якості функцій моніторингу мережі розглядається можливість отримання актуальної інформації про стан мережі: визначення переліку підключених пристроїв, маршрутизація та якість інтернет-з'єднання. Обрані функції діагностики призначені для виявлення потенційних загроз для мережі, перш за все, несанкціонованого доступу та недостатньо ефективного налаштування мережевих елементів. Функції аналізу мережевої продуктивності – перевірка швидкості інтернету та пінг – призначені для ідентифікації та усунення перешкод під час підключення до мережі.

Визначення основних функцій застосунку. Аналіз існуючих застосунків для мобільних пристроїв показує, що iOS застосунок мережевих інструментів повинен виконувати наступні функції: відображення загальної інформації про мережу, сканування мережевих інтерфейсів, відображення таблиці маршрутів, сканування локальної мережі для пошуку несанкціонованих підключень, утиліти *ping*, *traceroute*, *internet speed test*. Також мобільний застосунок має містити у собі функції надання загальної інформації щодо поточної мережі, відображення мережевих інтерфейсів, виведення таблиці маршрутизації, сканування локальної мережі для пошуку неавторизованих підключень, утиліти *ping*, *traceroute*, і тестування швидкості інтернету. Застосунок повинен мати зручний користувацький інтерфейс і локалізацію – застосовувати прийнятні для користувача мови. На кожному етапі розробки продукту

слід перевіряти відповідність користувацького сценарію реальній користувацькій взаємодії. Усі процеси створення продукту мають бути безперервними і циклічними [11].

Стек технологій. Оптимальними мовами програмування для розробки застосунка iOS, на наш погляд, є мови Swift, C, Objective C. В якості середовища розробки використовувалось інтегроване середовище розробки Xcode. Для розробки користувацького інтерфейсу був обраний фреймворк UIKit, який в даному випадку є більш придатним, ніж SwiftUI. Незважаючи на те, що фреймворк SwiftUI є більш декларативним і простішим для написання коду, він має певні обмеження для деяких задач [12]. Перевага UIKit над альтернативними фреймворками полягає в тому, що він є основним фреймворком для розробки користувацького інтерфейсу на iOS, має значну кількість інструментів і ресурсів, а також широку спільноту користувачів.

Діаграма послідовності. Для наочного відтворення процесів взаємодії між клієнтом і об'єктами та компонентами системи було побудовано діаграму послідовності (рис. 1). Аналіз діаграми послідовності дозволяє виявити потенційні неефективності у взаємодії об'єктів, зайві ускладнення, не оптимальні сценарії виконання [13].

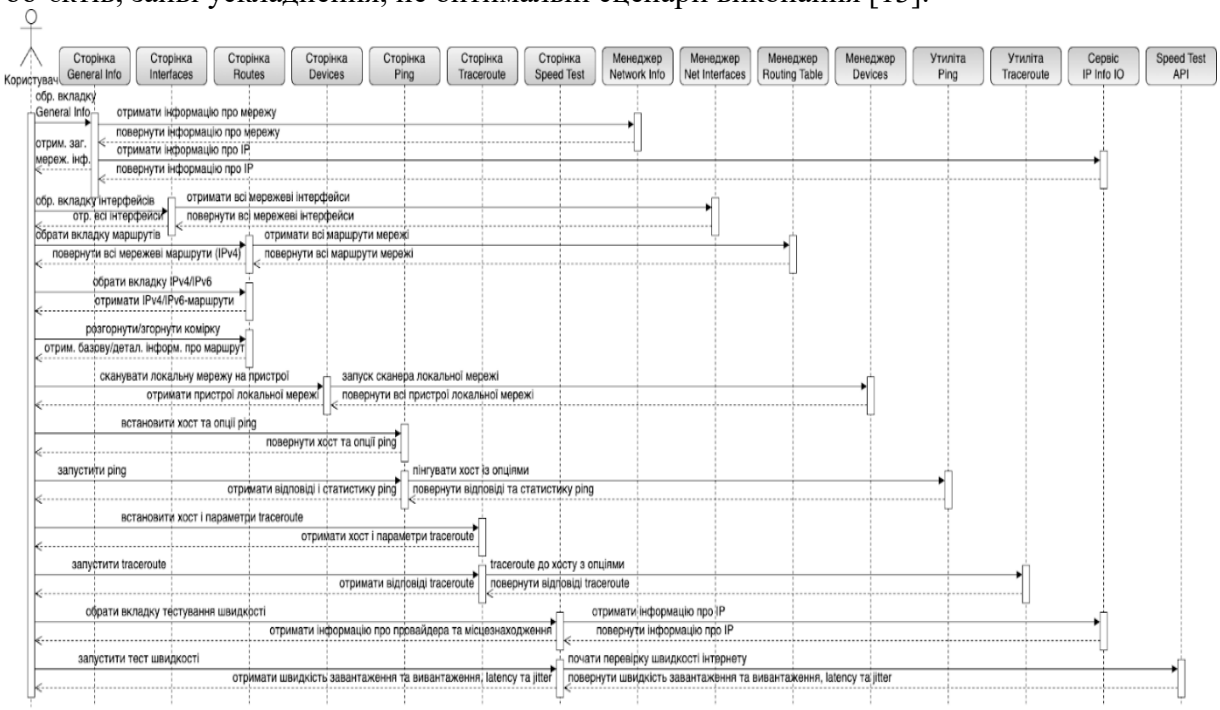


Рис. 1. Діаграма послідовності для розроблюваного iOS застосунку

Діаграма послідовності відображає об'єкти користувацького інтерфейсу та сервіси розроблюваного iOS застосунку. Користувацький інтерфейс – це сторінки програми і контролери. Через контролери відбувається звернення до сервісів для відображення інформації в користувацькому інтерфейсі і для відгуку на дії користувача. При переході користувача на сторінку *General Info* контролер запитує інформацію про мережу у сервісу. Сервіс повертає певні мережеві дані, наприклад IP-адресу маршрутизатора або мережевий шлюз. Таким чином у контролері створюються об'єкти мережевих даних, які впорядковуються і відображаються в користувацькому інтерфейсі. Під час переходу на сторінку мережевих інтерфейсів контролер запитує всі мережеві інтерфейси у менеджера. У контролері отримані інтерфейси групуються за протоколами IPv4 та IPv6 і відображаються в користувацькому інтерфейсі. При переході на сторінку мережевих маршрутів контролер запитує таблицю маршрутів у менеджера. У контролері отримані маршрути групуються за протоколами IPv4 та IPv6 і відображаються в різних вкладках. Під час зміни вкладок IPv4/IPv6 повторного запиту даних не відбувається. Об'єкт маршруту представлений у вигляді комірки, яка може бути розгорнути і згорнути.

Користувач переходить на сторінку *Devices* і натисканням на кнопку починає сканування мережі для отримання списку пристроїв у локальній мережі. Після цього відбувається звернення до сервісу щодо отримання списку пристроїв, контролер отримує об'єкти пристроїв і відображає їх у користувацькому інтерфейсі.

На сторінках *Ping* та *Traceroute* користувач має можливість встановити хост, на який буде виконуватися перевірка якості з'єднання, і опції, з якими він буде виконуватися. Операція (звернення до утиліти) починається після натискання на кнопку користувачем. Після завершення операції контролер отримує і відображає в користувацькому інтерфейсі ICMP-відповіді і статистику.

На сторінці *Speed Test* користувач має можливість отримати інформацію про швидкісні характеристики мережі. Під час переходу на сторінку контролер запрошує у сервісу інформацію про IP-адресу користувача. Таким чином користувач отримує інформацію щодо інтернет-провайдера і геолокації. Після натискання на кнопку запуску *Run Speed Test* відбувається вимірювання швидкості інтернету, в користувацькому інтерфейсі відображається швидкість завантаження і вивантаження даних, затримка і джитер – фазові спотворення сигналу, що передається.

Розробка дизайну застосунку. Перед початком розробки дизайну мобільного iOS застосунку результати проєктування були структуровані за компонентами проєкту і перенесені в платформу корпоративного програмного забезпечення *Atlassian Confluence* як документація проєкту. Відповідно до проєктної документації було реалізовано дизайн у веб-застосунку *Figma*, за яким розроблено iOS застосунок. Для дизайну iOS застосунку обрано стиль неоморфізм [14]. Цей стиль створює візуально-психологічний комфорт завдяки пастельним тонам кольорів, м'якості тіней і згладженості ліній. У дизайні iOS застосунку були використані векторні іконки, внутрішні та зовнішні тіні, градієнти.

Навігація по застосунку. Для навігації по iOS застосунку у ньому передбачене бічне меню з двома секціями: *Network Info* та *Utilities*, яке на екрані пристрою відображається ліворуч. Кожна секція містить панель вкладок, яка забезпечує навігацію між сторінками застосунка. У секції *Network Info* розміщені сторінки *General Info*, *Interfaces*, *Routes*, *Devices*. У секції *Utilities* розміщені сторінки *Ping*, *Traceroute*, *Speed Test*. Управління відображенням контенту виконує кореневий контролер. У кореневому контролері налаштовується кастомізований заголовок застосунка, який містить кнопки «Меню» та «Оновлення», а також відображає назву сторінки. Крім кнопки в заголовку iOS застосунка передбачено відкривання меню за допомогою свайп-технології. Контролер вкладок, який реалізує кастомізовану панель вкладок, призначений для відображення сторінок секцій через вкладки.

Реалізація логіки сторінок застосунку. Логіка сторінки загальної мережевої інформації реалізована в мережевому менеджері. Мережевий менеджер надає методи для отримання інформації про мережеві інтерфейси, IP-адреси, MAC-адреси, маску підмережі, адреси DNS-серверів, стан VPN-інтерфейсу та інші мережеві дані. Клас мережевого менеджера в основному використовує системні функції, бібліотеки і функції мовою C (*getifaddrs*, *resolv.h*, *inet_ntop*, *CFNetwork*).

На цій сторінці також відображаються дані про публічну IP-адресу, місцезнаходження, провайдера і ім'я хосту користувача. Для цього було розроблено сервіс, який реалізує взаємодію з API IPInfo.io. Структура даних API про публічну IP-адресу має властивості для IP, ім'я хосту, міста, регіону, країни, місця розташування, назви провайдера, поштового коду, часового поясу.

Логіка для сторінки *Interfaces* реалізована за допомогою сервісу мережевих інтерфейсів. Цей клас дає можливість ітерування за всіма доступними інтерфейсами і надає методи для їхньої фільтрації та отримання докладних характеристик. Серед властивостей об'єкту інтерфейсу: ім'я, IP-адреса, маска мережі, адреса призначення, перевірка належності до VPN. В структурі передбачені властивості для прапорів IFF_UP,

IFF_BROADCAST, IFF_LOOPBACK, IFF_POINTOPOINT, IFF_RUNNING, IFF_NOARP, IFF_MULTICAST, а також тип IP-адреси, тип адреси призначення, тип маски мережі.

Для реалізації логіки сторінки *Routes* було реалізовано класи менеджера маршрутів та об'єкту маршруту. Менеджер маршрутів надає методи для сканування маршрутів, отримання нещодавніх маршрутів і формування таблиці маршрутів. Серед властивостей моделі маршруту: адреса призначення маршруту, ім'я призначення маршруту, адреса шлюзу, ім'я шлюзу, прапори маршруту, посилання на маршрут, використання маршруту, ім'я мережевого інтерфейсу, сімейство маршруту, час закінчення маршруту. Інтерфейс користувача для сторінки *Routes* було реалізовано в контролері маршрутів. Контролер відображає маршрути мережі за вкладками IPv4/IPv6 у вигляді таблиці і реалізує перемикання між цими вкладками.

Сторінка *Devices* призначена для виявлення пристроїв, що підключені до локальної мережі. Для сканування мережі використовується бібліотека *MMLanScan*. Класичний мережевий сканер виконує пінг кожного хосту у мережі для побудови ARP-таблиці, після чого відбувається спроба отримання MAC-адреси кожного хосту. Якщо MAC-адресу знайдено, то вважається, що хост існує у мережі. Для пінгу використано бібліотеку *Apple SimplePing*. У контролері девайсів відбувається сканування та відображення пристроїв, підключених до локальної мережі. Якщо мережа стільникова або відсутнє підключення до мережі, користувачеві не надається можливість сканування. Для цієї функції також реалізовано ідентифікацію поточного пристрою.

Для перевірки цілісності і якості з'єднань в мережах використовується *ping.c* – реалізація стандартної утиліти *ping* на рівні коду C, що забезпечує повний контроль над ICMP-пакетами і дозволяє змінювати будь-які параметри, наприклад, розмір пакетів, тривалість існування пакетів в мережі (TTL), час між відправками. Для інтеграції *ping.c* у проєкт був створений клас, який включає необхідні властивості та методи для ініціалізації та виконання ICMP-запитів, таких як ім'я хосту, кількість пакетів, інтервал часу та обробка помилок. Була визначена структура для зберігання результатів, а також перелік можливих помилок при пінгуванні. Контролер реалізує інтерфейс користувача та логіку для виконання операцій *ping* у розробленому iOS застосунку. Він надає користувачеві можливість задавати наступні параметри для тестування підключення до мережі: хост, інтервал між пінгами, таймаут, розмір пакету та кількість пакетів.

Для реалізації операції визначення маршруту проходження даних від мобільного пристрою до сервера для зазначеного хосту було реалізовано окремий клас для утиліти *traceroute*. В розробленому класі доменне ім'я перетворюється на IP-адресу, після чого створюються сокети для відправки та отримання пакетів, ініціалізується значення TTL та інші змінні для вимірювання часу. Цикл *while* збільшує TTL на кожному кроці, до досягнення цільового хосту, або до перевищення максимального значення TTL. На кожному кроці на цільовий хост відправляються UDP-пакети зі збільшуваним TTL і очікується ICMP-відповідь від проміжних маршрутизаторів. Пакети надсилаються задану кількість разів для кожного TTL. При отриманні відповіді ICMP фіксується час проходження пакета і записується лог з інформацією про маршрутизатор. Якщо час очікування закінчився, або не отримано відповіді, у лог записується символ зірочки. При досягненні цільового хосту виконання припиняється. Через опції контролера встановлюються максимальна кількість переходів (*Bad Hops Limit*) та порт для визначення маршруту проходження даних.

Для тестування швидкості інтернету використовується *Speedchecker* SDK iOS, який дозволяє визначити такі параметри як затримка, швидкість завантаження та вивантаження з'єднання. SDK використовується для стільникових, бездротових та локальних мереж, надаючи деталі тестування, такі як поточна швидкість та прогрес. Методологія вимірювання швидкості інтернет-з'єднання ґрунтується на активних тестах з використанням локальних серверів та комерційних CDN, які передають великий обсяг інтернет-трафіку. Тест починається з відправки десяти ICMP-пакетів на сервер для

вимірювання затримки, після чого починається процес тестування швидкості завантаження та вивантаження, використовуючи два або десять потоків передачі HTTP-запитів, в залежності від можливостей з'єднання. Пакети даних передаються з високою швидкістю для повного залучення пропускної спроможності мережі, результати семплюються кожні 100 мс. Наприкінці тесту середня швидкість обчислюється як із семплів, так і з необроблених даних, після чого як остаточний результат обирається максимальне з двох значень.

Контролер для вимірювання швидкості передачі даних відображає процес тестування за допомогою кастомізованого елемента спідометру. Він візуально відображає поточну швидкість інтернет-з'єднання. Основними елементами є текстова мітка для швидкості передачі даних, стрілка та кругова шкала прогресу.

Висновки і подальші перспективи. За результатами аналізу спектру можливостей існуючих мобільних застосунків спроектовано iOS застосунок для мобільних пристроїв, який являє собою комплекс інтегрованих iOS засобів моніторингу та виявлення неполадок мобільної мережі. До набору функцій спроектованого iOS застосунку включено: сканування мережі, аналіз Wi-Fi, перевірку конфігурації маршрутизатора, аналіз рівнів шифрування Wi-Fi, перевірку швидкості інтернету, перевірку цілісності і якості з'єднань в мережах, визначення маршруту проходження даних тощо.

Розроблений iOS застосунок повністю відповідає поставленим функціональним вимогам. Усі функції розглянутого застосунку мають високу точність і швидкість виконання порівняно з еталонними інструментами. Завдяки широкому набору функцій та зручному інтерфейсу розроблений iOS застосунок може використовуватися як професіоналами, так і звичайними користувачами. Розроблений iOS застосунок з компактним набором мережевих інструментів підвищує швидкість реагування на мережеві події, що своєю чергою сприяє підвищенню рівня безпеки мережевого підключення з пристроїв iOS.

В цілому розроблений iOS застосунок має широкі можливості для розширення у вигляді застосункових мережевих інструментів для перевірки вразливостей мережі, технології VPN, брандмауера, сніфера трафіку. Серед застосункових функцій безпеки слід звернути увагу на функції виявлення компрометації даних користувача.

Подальші перспективи удосконалення розробленого iOS застосунку лежать в сфері застосування технологій машинного навчання і штучного інтелекту, які пропонують інноваційні рішення для усунення неполадок у мережах [15, 16]. Перспективним напрямком є автоматичне виявлення та діагностика помилок мережних підключень на основі моделей аномальної поведінки або систем попереджень, що використовують шаблони. Впровадження методів машинного навчання для прогнозування вразливостей і аналізу інцидентів в режимі реального часу є передумовою для підвищення ефективності управління мережевою безпекою. Для аналізу вразливостей можливе застосування методів обробки даних і методів NLP (обробка природної мови), які дають змогу збирати дані про загрози з відкритих джерел, оцінювати їх і класифікувати. Методи контрольованого навчання можуть бути впроваджені для пошуку шаблонів вразливостей в історичних даних.

Список літератури

1. Zihan Zhou. Distributed WSN Vulnerability Remediation System Based on Mobile-N Policy. 2023. URL: https://doi.org/10.21203/rs.3.rs-3740423/v1
2. Abdellaoui A., Elmhamdi J. Network Stability Based Multicriteria Weighted MPRs Selection Algorithm for Mobile Ad Hoc Networks. *International Journal of Communication Networks and Information Security*. 2024. vol. 16, № 2. 17 p. URL: https://doi.org/10.17762/ijcnis.v16i2.6668.
3. Goggin G. Apps: From mobile phones to digital lives. John Wiley & Sons, 2021. 154 p.
4. Rahul Ranjan, Ram Keshwar Prasad Yadav. A Decision Framework for Enhancing Adhoc Network Stability and Security. *International Journal of Innovative Science and Research*

Technology. 2024. V. 9, Iss. 10. P. 55–61. URL: https://doi.org/10.38124/ijisrt/IJISRT24OCT246

5. Faria Nawshin, Radwa Gad, Devrim Unal, Abdulla Khalid Al-Ali, Ponnuthurai N. Suganthan. Malware detection for mobile computing using secure and privacy-preserving machine learning approaches: A comprehensive survey. *Computers and Electrical Engineering*. 2024. Vol. 117. 24 p. URL: https://doi.org/10.1016/j.compeleceng.2024.109233.
6. Hohenegger V. Developing a Vulnerability Assessment Concept for eHealth iOS Applications. Diss. Wien, 2021. 157 p. URL: https://web.archive.org/web/20220115233717id_/https://repositum.tuwien.at/bitstream/20.500.12708/18842/1/Hohenegger%20Vanessa%20-%202021%20-%20Developing%20a%20Vulnerability%20Assessment%20Concept%20for...pdf
7. Jivthesh M. R., Gaushik M. R., Adarsh P., Niranga G. H., Rao N. S. A Comprehensive survey of WiFi Analyzer Tools. *2022 IEEE 3rd Global Conference for Advancement in Technology, Bangalore, India*. 2022. P. 1–8. URL: 10.1109/GCAT55367.2022.9972040 https://ieeexplore.ieee.org/abstract/document/9972040/metrics#metrics
8. 10 Best Wi-Fi analyzer apps for iPhone and iPad. URL: https://www.igeeksblog.com/best-iphone-ipad-wifi-analyzer-apps/
9. Masum M. R. iOS App Development Training. 2018. URL: 10.13140/RG.2.2.10443.23847
10. Iversen J., Eierman M. Learning Mobile App Development: A Hands-on Guide to Building Apps with iOS and Android. Prospect Press, 2017. 352 p.
11. Резнік Р. Ю., Антонов Ю. С. Розробка застосунків під платформи Android та iOS. *Прикладні інформаційні технології*: мат. всеукр. науково-практ. конф. Вінниця : Донецький національний університет імені Василя Стуса. 2020. С. 132–135. URL: https://www.researchgate.net/publication/355370862_ROZROBKA_DODATKIV_PID_PLATFORMI_ANDROID_TA_IOS
12. Mohamed Ahmed Eltaher. SwiftUI vs UIKit: A Comprehensive Comparison. URL: https://medium.com/@mohamed.ahmedeltaher/swiftui-vs-uikit-a-comprehensive-comparison-92f58507495f#:~:text=SwiftUI%20and%20UIKit%20are%20both,framework%20with%20extensive%20customization%20options
13. System Sequence Diagram Used in Software Development. URL: https://www.researchgate.net/publication/371904764_System_Sequence_Diagram_Used_in_Software_Development
14. Bjork S. Flat and neumorphic design: aesthetic preferences compared between age groups. *21st Student Conference in Interaction Technology and Design*. Umea University, 2021. P. 71–78. URL: https://www.diva-portal.org/smash/get/diva2:1574853/FULLTEXT01.pdf#page=75
15. Cimitile A., Martinelli F., Mercaldo F. Machine Learning Meets iOS Malware: Identifying Malicious Applications on Apple Environment. In *Proceedings of the 3rd International Conference on Information Systems Security and Privacy*. Porto, Portugal, 2017. P. 487–492. URL: 10.5220/0006217304870492
16. Amster A. Automating Vulnerability Detection in Networks with AI. URL: https://www.allstarsit.com/blog/automating-vulnerability-detection-in-networks-with-ai

А. М. Макарова, І. А. Ярова

THE DEVELOPMENT OF IOS MOBILE APP FOR SOLVING THE NETWORK CONNECTION SECURITY PROBLEM

A. M. Makarova, I. A. Yarova

Odesa National Polytechnic University
1, Shevchenko Ave., 65044, Odesa, Ukraine
Emails: anastasia.mikki@gmail.com, yarova@op.edu.ua

To maintain mobile network in an efficient and secure state, an iOS mobile app has been developed, which is a specialized set of integrated network tools for monitoring, analyzing, and diagnosing network activity. The functions of up-to-date mobile apps were analyzed and the researches in the field of designing of applications for iOS mobile devices. It is shown, that most apps are of narrow specialization. The necessity to integrate network tools in one mobile app is proven. The basic set of integrated network tools for the developed iOS mobile app has been determined, which should provide network monitoring, vulnerability detection, and efficiency optimization functions. The technology stack for the developed iOS mobile app has been selected. Using the sequence diagram, the main functions of iOS app system have been determined, the interaction between the user and the system is described, and interaction between the components of iOS app system, which are user interface objects and services, is presented. The development of iOS mobile app logic and user interface of iOS mobile app is described. The design of developed iOS mobile app is inspired by neomorphism style. It is shown that further possibilities for expanding the functionality of the developed iOS app are associated with the use of machine learning and artificial intelligence technologies.

Keywords: iOS mobile app, iOS mobile devices, integrated network tools, network security, network monitoring, network diagnostics

DOI 10.15276/imms.v14.no3.199
УДК 519.6, 539.3

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 199-204

ДИФРАКЦІЯ ПЛОСКИХ ГАРМОНІЧНИХ ХВИЛЬ НА ЖОРСТКОМУ ЦИЛІНДРИЧНОМУ ВКЛЮЧЕННІ ДОВІЛЬНОГО ПОПЕРЕЧНОГО ПЕРЕРІЗУ

Б.Є. Панченко[1], Ю.О. Гунченко[2], Л.М. Тимошенко[3],
Л.Я. Мартинович[4], М.В. Северін[5]

[1,5]Державний університет інтелектуальних технологій та зв'язку
1, Кузнечна вул., Одеса, 65000, Україна
[2,4] Одеський національний університет ім. Мечникова
2, Всеволода Змієнка, Одеса, 65082, Україна
[3] Національний університет «Одеська політехніка»
1, Шевченка пр., 65044, Одеса, Україна
Emails: pr-bob@ukr.net[1], gunchenko@onu.edu.ua[2], l.m.timoshenko@op.edu.ua[3],
larysa.yaroslavna@onu.edu.ua[4], n_severin@ukr.net[5]

Високоточним чисельним дослідженням апробовано математичну модель, засновану на методі сингулярних інтегральних рівнянь. Досліджено крайову задачу про дифракцію плоских пружних стаціонарних хвиль на жорсткому циліндричному включенні довільного поперечного перерізу, розташованому в нескінченній ізотропному середовищі. Система сингулярних інтегральних рівнянь розв'язується обчислювальним методом механічних квадратур. Використано додаткові умови у інтегральній формі. Для підвищення точності результатів застосовано розпаралелювання алгоритму. Наводиться аналіз напружено-деформованого стану границі включення. Досліджено параметри механічних полів на неоднорідності ромбічної форми. Наприклад, отримано те, що при порівнянні результатів для різних типів хвиль, що взаємодіють з включенням, можна зробити висновок, що є принципова відмінність у розподілі контурних напружень при набіганні на жорстке включення хвилі розширення-стиснення (Р-випадок) чи хвилі зсуву (SV-випадок). У Р-випадку нормальне контурне напруження досягає локальних максимумів в лобовій та тіньовий точках відповідно. У SV-випадку нормальне напруження у цих точках дорівнюють нулю і досягають максимуму в околі точки зісковзування. Дотичне напруження в Р-випадку в лобовій і тіньовій точках дорівнюють нулю, а його максимум досягається поблизу точки зісковзування. У SV-випадку нормальне напруження приймає максимальне значення в лобовій точці і має локальний максимум поблизу точки зісковзування При збільшенні параметра відносної щільності включення спостерігається збільшення нормального напруження у Р-випадку та дотичного у SV-випадку поблизу лобової точки та їх зменшення в околі тіньової точки. А максимум дотичного напруження у Р-випадку та нормального у SV-випадку зміщується з тіньової області в освітлену. Крім того, у Р-випадку переважають нормальне напруження, а значення параметра відносної щільності практично не впливає на значення цього напруження в точці зісковзування.

Ключові слова: плоскі гармонічні хвилі, дифракція, сингулярні інтегральні рівняння, чисельний експеримент, жорстке включення.

Вступ. Дифракцію плоских гармонічних хвиль на відбивачах складної геометричної форми ефективно моделює метод інтегральних рівнянь [1,2]. У [2] цим методом вперше досліджено плоску динамічну задачу про коливання ізотропного середовища з пружним включенням довільного поперечного перерізу. У представленій тут роботі метод інтегральних рівнянь, запропонований в [2,3], використовується для чисельного аналізу механічних параметрів хвильового поля у випадку дифракції плоских хвиль на жорсткому («рухомому», проте абсолютно жорсткому) циліндричному включенні довільного поперечного перерізу.

В [2] зазначено, що запропонований там алгоритм моделювання дифракції плоских пружних хвиль має властивість не лише теоретичного, а й чисельного розв'язання окремих випадків, коли відбивачами є отвори [4], нерухомі [5] або рухомі абсолютно жорсткі включення [3] – завдяки варіантності вихідних параметрів.

Постановка задачі. Розглянемо у необмеженому ізотропному середовищі з коефіцієнтами Ламе λ, μ та щільністю ρ нескінченно довгий вздовж осі Ox_3 циліндр, поперечний переріз якого обмежений замкненим контуром L типу Ляпунова. Передбачається, що циліндр є абсолютно жорстким тілом зі щільністю ρ_0. У середовищі, перпендикулярно осі циліндра, поширюється гармонійна (залежність від часу виражається множником $e^{-i\omega t}$) хвиля розширення-стиснення (P -випадок):

$$U_1^{(0)} = 0,\ U_2^{(0)} = \tau_1 e^{-i\gamma_1 x_2},\ \gamma_1 = \frac{\omega}{c_1},\ c_1 = \sqrt{\frac{\lambda + 2\mu}{\rho}},\ \tau_1 = const \tag{1}$$

або хвиля зсуву (SV -випадок):

$$U_1^{(0)} = \tau_2 e^{-i\gamma_2 x_2},\ U_2^{(0)} = 0,\ \gamma_2 = \frac{\omega}{c_2},\ c_2 = \sqrt{\frac{\mu}{\rho}},\ \tau_2 = const \tag{2}$$

де де c_1 и c_2 - швидкості поздовжньої та поперечної хвиль у матриці, ω - частота коливань, $i^2 = -1$

Взаємодіючі зі включенням, хвиля, що набігає, породжує відбиті поздовжні та поперечні хвилі. Їхня сукупність визначає напружено-деформований стан середовища, який потрібно визначити. Наслідуючи принцип суперпозиції [1,2], загальне поле амплітуд переміщень і компонент тензора напружень будемо шукати у вигляді:

$$U_n = U_n^{(0)} + U_n^{(1)},\ \tau_{mn} = \tau_{mn}^{(0)} + \tau_{mn}^{(1)},$$
$$\tau_{mn} = \lambda\delta_{mn}\left(U_{1,1} + U_{2,2}\right) + \mu\left(U_{m,n} + U_{n,m}\right),\ m,\ n = 1,\ 2. \tag{3}$$

Тут $U_n^{(0)}$, $\tau_{mn}^{(0)}$ и $U_n^{(1)}$, $\tau_{mn}^{(1)}$ - амплітуди компонент вектора переміщень та тензора напружень падаючого і відбитого хвильових полів відповідно, δ_{mn} - символ Кронекера.

Відбите поле переміщень повинно задовольняти умові випромінювання на нескінченності, а також рівнянню руху [1]. Крім того, на поверхні жорсткого включення повинні виконуватись граничні умови:

$$U_1 = B_1 - \omega_0\eta,\ U_2 = B_2 + \omega_0\xi,\ \ \zeta = \xi + i\eta \in L \tag{4}$$

де B_1, B_2 и ω_0 - амплітуди поступального руху та жорсткого повороту включення.

Метод дослідження. Представлення амплітуд переміщень відбитого хвильового поля будемо шукати у вигляді потенціалів типу простого шару [1,2] (підсумовування $n = 1,2$):

$$U_k^{(1)}(M) = \int_L V_n^{(k)}(M,\ P)\,p_n(s)\,ds,\ k = 1,\ 2, \tag{5}$$

де $p_n(s)$ - невідомі функції; $V_n^{(k)}$ - компоненти матриці Гріна, що являють собою амплітуди переміщень у точці M при дії гармонічної сили, прикладеної у точці $P \in L$ та спрямованої вздовж осі Ox_1 ($k = 1$) або вздовж осі Ox_2 ($k = 2$).

Амплітуди переміщень $V_n^{(k)}$ та відповідних компонентів тензора напружень $\sigma_{mn}^{(k)}$ визначаються із співвідношень ($k,\ m,\ n = 1,\ 2$):

$$V_n^{(k)} = (-1)^{n+k} L_{nk} G,\quad \sigma_{mn}^{(k)} = \lambda\delta_{mn}(V_{1,1}^{(k)} + V_{2,2}^{(k)}) + \mu(V_{m,n}^{(k)} + V_{n,m}^{(k)}),$$

$$(\Delta + \gamma_1^2)(\Delta + \gamma_2^2)G = c\,\delta(x_1 - \xi,\ x_2 - \eta),\ \ c = -\frac{1}{\mu(\lambda + 2\mu)},$$

$$G(M,\,P)=\frac{c}{4i}\frac{H_0^{(1)}(\gamma_1 r)-H_0^{(1)}(\gamma_2 r)}{\gamma_2^2-\gamma_1^2}. \tag{6}$$

Тут $H_j^{(1)}(x)$ функція Ханкеля першого роду j-го порядку, Δ - оператор Лапласа.

Використовуючи фундаментальне рішення $G(M,\,P)$, для комбінацій переміщень $V_n^{(k)}$ отримуємо такі вирази:

$$V_1^{(1)}+iV_2^{(2)}=d\left(\frac{\chi}{4}\Phi_{20}-(0{,}5-\nu)\gamma_2^2\Phi_{00}\right),\quad V_1^{(2)}=V_2^{(1)},$$

$$e^{2i\alpha}\left(V_1^{(1)}-2iV_1^{(2)}-V_2^{(2)}\right)=e^{-2i\alpha}\left(V_1^{(1)}+2iV_1^{(2)}-V_2^{(2)}\right)=\frac{d}{4}\Phi_{22},\ d=\frac{i}{4\mu(1-\nu)},$$

$$\Phi_{l\,j}=\frac{\gamma_1^l H_j^{(1)}\left(\gamma_1 r\right)-\gamma_2^l H_j^{(1)}\left(\gamma_2 r\right)}{\gamma_1^2-\gamma_2^2},\ z-\zeta=re^{i\alpha},\ z=x_1+ix_2,\ \chi=3-4\nu,\ \nu=\frac{\lambda}{2(\lambda+\mu)}. \tag{7}$$

Аналіз формул (7) показує, що функції $V_1^{(1)}-2iV_1^{(2)}-V_2^{(2)}$ та $V_1^{(1)}+2iV_1^{(2)}-V_2^{(2)}$ безперервні в нулі, а функція $V_1^{(1)}+iV_2^{(2)}$ має логарифмічну особливість. Це означає, що задоволення граничних умов (4) зводить крайову задачу до системи інтегральних рівнянь із логарифмічними ядрами, чисельна реалізація яких є ускладненою. Тому, з метою отримання сингулярних інтегральних рівнянь з ядрами типу Коші, граничні умови (4) диференціювалися за дуговою координатою s_0 і записувалися у вигляді:

$$\left.\frac{d(U_1+iU_2)}{ds_0}\right|_L=i\omega_0 e^{i\varphi_0},\quad \left.\frac{d(U_1-iU_2)}{ds_0}\right|_L=-i\omega_0 e^{-i\varphi_0},\quad \left.\frac{dW}{ds_0}\right|_L=\left(\frac{\partial W}{\partial z}e^{i\varphi_0}+\frac{\partial W}{\partial \bar z}e^{-i\varphi_0}\right)_{z\to\zeta_0}, \tag{8}$$

де φ_0 - кут позитивної дотичної до L в точці $\zeta_0=\xi_0+i\eta_0\in L$ з віссю Ox_1, $\bar z=x_1-ix_2$.

Система сингулярних інтегральних рівнянь. В роботах [2,3] надано методику розв'язання цієї крайової задачі. Отже, задовольняючи граничні умови (8), приходимо до системи сингулярних інтегральних рівнянь першого роду з ядрами типу Коші (підсумовування для $n=1,\,2$):

$$\int_L B_{mn}(s_0,\,s)f_n(s)ds-M_m(s_0)\omega_0=-N_m(s_0),\ m=1,\,2; \tag{9}$$

$$B_{12}=\frac{d}{8}\left(-\frac{2}{\pi i}\frac{e^{i\varphi_0}-e^{i(2\alpha_0-\varphi_0)}}{\bar\zeta-\bar\zeta_0}+F_{31}^0e^{i(\varphi_0+\alpha_0)}-F_{33}^0e^{i(3\alpha_0-\varphi_0)}\right),$$

$$B_{21}=\frac{d}{8}\left(-\frac{2}{\pi i}\frac{e^{-i\varphi_0}-e^{-i(2\alpha_0-\varphi_0)}}{\zeta-\zeta_0}+F_{31}^0e^{-i(\varphi_0+\alpha_0)}-F_{33}^0e^{-i(3\alpha_0-\varphi_0)}\right),$$

$$B_{11}=B_{22}=-d\left(\frac{\chi}{2\pi i}\frac{\cos(\varphi_0-\alpha_0)}{r_0}+\left(\frac{\chi}{4}F_{31}^0-(0{,}5-\nu)\gamma_2^2\Phi_{11}^0\right)\cos(\varphi_0-\alpha_0)\right),$$

$$f_1(s)=p_1(s)+ip_2(s),\ f_2(s)=p_1(s)-ip_2(s),\quad M_1=i\omega_0e^{i\varphi_0},\ M_2=\overline{M}_1,$$

$$\Phi_{l\,j}^0=\frac{\gamma_1^l H_j^{(1)}\left(\gamma_1 r_0\right)-\gamma_2^l H_j^{(1)}\left(\gamma_2 r_0\right)}{\gamma_1^2-\gamma_2^2},\quad \zeta_0-\zeta=r_0e^{i\alpha_0},\quad F_{31}^0=-\frac{2i}{\pi r_0}+\Phi_{31}^0,\quad F_{33}^0=-\frac{2i}{\pi r_0}+\Phi_{33}^0,$$

$N_1=-N_2=\gamma_1\tau_1e^{-i\gamma_1\eta_0}\sin\varphi_0$ для P-хвилі, $N_1=N_2=-i\gamma_2\tau_2e^{-i\gamma_2\eta_0}\sin\varphi_0$ - для SV.

Аналіз ядер системи рівнянь (9) показує, що ядра B_{11} и B_{22} є сингулярними, а B_{12} и B_{21} - неперервними.

Необхідні для замикання алгоритму три додаткові умови випливають із законів поступального та обертального руху абсолютно жорсткого тіла. Для поступального руху, виходячи з другого закону Ньютона, отримуємо:

$$\int_L S_1 ds = -\omega^2 \rho_0 S_0 B_1, \ \int_L S_2 ds = -\omega^2 \rho_0 S_0 B_2, \tag{10}$$

а рівняння, що описує обертальний рух, запишемо у вигляді:

$$\int_L (S_1(\eta - a_2) - S_2(\xi - a_1)) ds = -\omega^2 J_A \omega_0, \tag{11}$$

де S_1 и S_2 - амплітуди компонент вектору напружень на контурі L; S_0 - площа включення, обмеженого контуром L; J_A - момент інерції включення відносно точки $A(a_1, a_2)$; постійні B_1 та B_2 визначаються згідно (4).

Чисельна реалізація. Як і в [1-3], для чисельної реалізації алгоритму використовувався метод механічних квадратур [6]. Розпаралелювання алгоритму здійснювалося аналогічно [3].

Як приклад розглянемо середовище, що містить жорстке циліндричне включення ромбічної форми зі закругленими кутами [7]:

$$\xi(\beta) = a(\sin\beta - \vartheta\sin 3\beta), \ \ \eta(\beta) = -b(\cos\beta + \vartheta\cos 3\beta), 0 \le \beta \le 2\pi \tag{12}$$

де при $\vartheta = 0.14036$ контур має форму ромба (у випадку $\vartheta = 0$ контур має еліптичну форму $\xi = a\sin\beta, \ \eta = -b\cos\beta, \ 0 \le \beta \le 2\pi$).

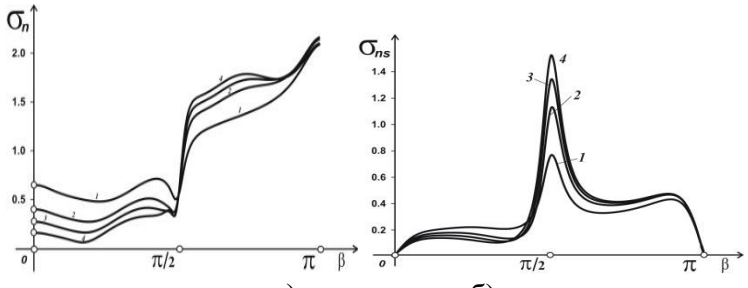
На границі включення проводилося обчислення безрозмірних напружень

$$\sigma_n = |\tau_n| / P, \ \sigma_S = |\tau_S| / P, \ \sigma_{ns} = |\tau_{ns}| / P, \tag{13}$$

де τ_n, τ_{ns} - амплітуди нормального та тангенціального напруження на L; τ_S визначається із співвідношення $\tau_S + \tau_n = \tau_{11} + \tau_{22}$; $P = \gamma_1 \tau_1 (\lambda + 2\mu)$ - максимальне напруження в P-хвилі (1), та $P = \gamma_2 \tau_2 \mu$ у разі SV-хвилі (2).

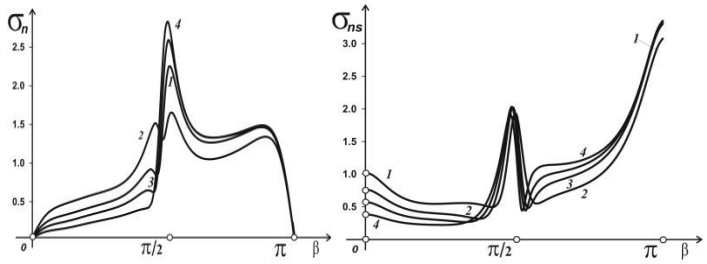
В роботі [3], де також використано метод [2], наведено чисельні результати, які дозволяють перевірити достовірність результатів цієї роботи. Зазначимо, що при дифракції поздовжньої (1) або поперечної (2) хвилі на жорсткому включенні з граничними умовами (4) та додатковими умовами (10)-(11) напруження σ_S завжди менше σ_n і пов'язане з ним співвідношенням $\sigma_s = \nu \cdot \sigma_n / (1 - \nu)$.

Також треба зазначити, що згідно з рекомендаціями [6] додаткові умови (10)-(11) можна моделювати як в будь-якій одній точці контуру, так і в інтегральному вигляді. Проте подальші високоточні численні дослідження показали, що при збільшенні значення безрозмірного хвильового числа $\gamma_{1,2} a$ для P- та SV-хвиль спостерігаються суттєві відмінності результатів. Тому тут використані лише «інтегральні» додаткові умови [6].



а) б)

Рис. 1. Розподіл напружень разі набігання P- хвилі при $b/a = 0{,}5$ та $\nu = 0{,}3$



а) б)

Рис. 2. Розподіл напружень разі набігання SV -хвилі при $b/a=0{,}5$ та $\nu=0{,}3$

На рис. 1 (а, б) та 2 (а, б) наведено розподіл напружень σ_n і σ_{ns} вздовж контуру ромбічного жорсткого включення (12) у разі набігання P- та SV -хвилі відповідно при $b/a=0{,}5$ та $\nu=0{,}3$, що для порівняння відповідають [3]. Проте тут, на відміну від [3], $\gamma_1 a=\gamma_2 a=3{,}0$. Як і в [3], криві 1, 2, 3 та 4 відповідають значенням $\rho_0/\rho=0{,}5;\ 1{,}0;\ 2{,}0$ та $5{,}0$. Очевидно, що спостерігається збіг загального вигляду кривих. Проте суттєва залежність результатів від значення хвильового числа.

Висновки. У P-випадку напруження σ_n досягає локальних максимумів в лобовій $\beta=180^0$ та тіньовий $\beta=0^0$ точках відповідно. У SV -випадку напруження σ_n у цих точках дорівнюють нулю і досягають максимуму в околі точки зісковзування ($\beta=90^0$).

Напруження σ_{ns} в P-випадку в лобовій і тіньовій точках дорівнюють нулю, а їх максимум досягається поблизу точки зісковзування. У SV -випадку напруження σ_{ns} приймає максимальне значення в лобовій точці і має локальний максимум поблизу точки $\beta=90^0$.

При збільшенні параметра ρ_0/ρ спостерігається збільшення напружень σ_n у P-випадку та σ_{ns} у SV -випадку поблизу лобової точки та їх зменшення в околі тіньової точки. А максимум напружень σ_{ns} у P-випадку та σ_n у SV -випадку зміщується з тіньової області ($60^0<\beta<90^0$) у освітлену ($90^0<\beta<120^0$). Крім того, у P-випадку переважають напруження σ_n, а значення параметра ρ_0/ρ практично не впливає на значення цього напруження в точці зісковзування $\beta=90^0$.

Список літератури

1. Назаренко А.М. Вычислительные методы в задачах дифракции упругих волн на системах неоднородностей на базе сингулярных интегральных уравнений. Сумы: СумГУ, 2015. 220 с.
2. Панченко Б.Є. Розв’язання двовимірних задач дифракції пружних хвиль на циліндричних неоднорідностях. Автореферат дисертації на здобуття вченого ступеня кандидата фізико-математичних наук. Суми: Сумський державний університет, 1996.19 с
3. Панченко Б.Є., Северин М.В. Математичне моделювання дифракції плоских гармонічних хвиль на жорсткому циліндричному включенні довільного поперечного перерізу. *Проблеми керування та інформатики*. 2024. № 3, С. 33-45.
4. Shibahara M., Taniquchi Y., Application of the integral equation method to the elastodynamic boundary-value problems. *Bull JSME*. 1983. Vol.26. No. 222. P.2054-2059.
5. Shibahara M., Taneto S., Kuroyanagi O., Diffraction of steady stress waves by arbitrary shaped discontinuities in elastic medium. *Bull JSME.* 1980. Vol.23. No. 178. P.493-500.
6. Панасюк В.В., Саврук М.П., Назарчук З.Т. Метод сингулярных интегральных уравнений в двумерных задачах дифракции. К., 1984. 344 с.

7. Гузь А.Н., Немиш Ю.Н. Методы возмущений в пространственных задачах теории упругости. К.: Вища школа, 1982. 352с.

DIFFRACTION OF PLANE HARMONIC WAVES ON A RIGID CYLINDRICAL INCLUSION OF AN ARBITRARY CROSS SECTION

Panchenko B.E.[1], Gunchenko Yu.O.[2], Tymoshenko L.M.[3],
Martynovych L.Ya.[4], Severin M.V.[5]

[1,5] State University of Intellectual Technologies and Communication
1, Kuznechna, Odesa, 65000, Ukraine
[2,4] Odesa National University named after Mechnikov
2, Vsevolod Zmienko, Odesa, 65082, Ukraine
[3] National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
Emails: pr-bob@ukr.net[1], gunchenko@onu.edu.ua[2], l.m.timoshenko@op.edu.ua[3],
larysa.yaroslavna@onu.edu.ua[4], n_severin@ukr.net[5]

A mathematical model based on the method of singular integral equations was tested by high-precision numerical research. The boundary value problem of the diffraction of plane elastic standing waves on a rigid cylindrical inclusion of arbitrary cross-section located in an infinite isotropic medium was studied. The system of singular integral equations is solved by the computational method of mechanical quadrature. Additional conditions are used in integral form. To increase the accuracy of the results, the parallelization of the algorithm was applied. An analysis of the stress-strain state of the inclusion boundary is given. The parameters of the mechanical fields on the inhomogeneities of the rhombic shape were studied. For example, it was found that when comparing the results for different types of waves interacting with the inclusion, it can be concluded that there is a fundamental difference in the distribution of contour stresses when running into a rigid inclusion of an expansion-compression wave (P-case) or a shear wave (SV -case). In the P-case, the normal contour stress reaches local maxima at the frontal and shadow points, respectively. In the SV case, the normal stress at these points is zero and reaches a maximum around the slip point. The tangential stress in the P-case at the frontal and shadow points is zero, and its maximum is reached near the slip point. In the SV case, the normal stress takes a maximum value at the frontal point and has a local maximum near the slip point. When the relative inclusion density parameter increases, there is an increase in the normal stress in the P case and tangential stress in the SV case near the frontal point and their decrease around the shadow point. And the maximum of the tangential stress in the P-case and the normal stress in the SV-case shifts from the shadow area to the illuminated area. In addition, in the P-case, the normal stress prevails, and the value of the relative density parameter practically does not affect the value of this stress at the slip point.
Keywords: plane harmonic waves, diffraction, singular integral equations, numerical experiment, hard inclusion.

DOI 10.15276/imms.v14.no3.205
УДК 004.056.55

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 205-214

АЛГОРИТМ АДАПТИВНОГО ОЧИШЕННЯ ВІД ШУМУ ЗАХИЩЕННОГО ЗОБРАЖЕННЯ З КАМЕР ВІДЕОСПОСТЕРЕЖЕННЯ

А. В. Садченко, О. А. Кушніренко, М. М. Іжак, В. В. Громов, В. О. Назаров

Національний університет «Одеська політехніка»
1, Шевченка пр., м.Одеса, 65044, Україна
Email: koa@op.edu.ua

Використання водяних знаків сумісно з відео-потоком з камер відеоспостереження дозволяє не тільки вирішити стандартні задачі такі як, наприклад, підтвердження справжності зображення, захист авторських прав но і додають додаткову інформацію, яку зручно використовувати при автоматичному формування реєстрів та баз даних. В якості додаткової інформації можуть бути персональні дані водіїв, номери та час паркування транспортних засобів, якщо камера встановлена на паркінгу чи час ДТП, якщо камера призначена для фіксації саме таких подій. Враховуючі нестабільні умови в яких може відбуватися відео-фіксація подій, та недосконаліть технічної складової електронних схем камер з'являється необхідність зменшення рівня шумів на захищеному зображенні. Мета роботи полягає в розробці адаптивного та з низькою обчислювальною складністю алгоритму очищення від шуму як вихідного зображення контейнеру так і зображення ЦВЗ. Зазвичай зображення контейнеру володіє значно більшою ентропією чим ЦВЗ, що призводить до парадоксу сутність якого в тому, що оптимальний алгоритм фільтрації щодо зображення контейнеру може виявиться таким, що спотворить зображення ЦВЗ і навпаки. Тому пошук алгоритму, що може в однакової мірі поліпшити обидва компоненти захищеного зображення є актуальним завданням, що і було вирішено у даної роботі. В якості основного критерію для оцінки якості зображень після усунення шуму було обрано величину коефіцієнту кореляції між зображенням, що не містить шуму – еталонним та зображенням після фільтрації. З метою зниження обчислювальної складності алгоритму було обрано принцип медіанної фільтрації з апертурою, розмір якої залежить від дисперсії шуму. З метою знаходження оптимальних значень розміру апертури медіанного фільтру було побудовано залежності коефіцієнтів кореляції вихідного і обробленого зображень від дисперсії шуму. Дане дослідження дозволило виявити екстремуми, що є крапками оптимальних значень розмірів вікна медіанного фільтру. Було запропоновано адаптивний алгоритм медіанної фільтрації, що спочатку здійснює обробку захищеного зображення з фіксованим розміром вікна (3x3), а далі, після вилучення ЦВЗ використовуються ще два медіанних фільтра зі змінними значеннями вікна окремо для зображення контейнеру та ЦВЗ. Такій підхід дозволив досягнути значення коефіцієнтів кореляції більш ніж 0.9 при дисперсії шуму 0.3. Так як медіанний фільтр не містить ні операцій множення ні операцій підсумовування то запропонований алгоритм може бути реалізований на дешевій елементній базі, наприклад, мікроконтролерах AVR чи ARM архітектури, що коштують значно менше ніж спеціалізовані сигнальні процесори чи швидкодіюча програмована логіка.

Ключові слова: цифровий водяний знак, захищене зображення, очищення від шуму, камера відеоспостереження, рівень спотворення інформації, найменш значущий біт.

Вступ. В умовах зростаючих загроз та потреб у захисті інформації, відеоспостереження стає невід'ємною частиною стратегій безпеки як для бізнесу, так і для державних установ. При цьому, водяні знаки можуть бути додатковим рівнем захисту даних. В подальшому зображення із доданим цифровим водяним знаком будемо називати захищене зображення [1]. Використання цифрових водяних знаків (ЦВЗ, англійською DWM — Digital watermark) сумісно з зображеннями, що отримані з камер відеоспостереження [2,3] зазвичай має наступні цілі:

- Захист авторських прав: водяні знаки допомагають запобігти несанкціонованому використанню та розповсюдженню відеоматеріалів.

- Ідентифікація джерела: вони дозволяють швидко визначити, звідки було отримано відеопотік, що важливо задля забезпечення цілісності даних.
- Аудит та контроль: водяні знаки можуть використовуватись для відстеження та документування змін у відеопотоці.
- Збільшення довіри: наявність водяних знаків підвищує довіру користувачів до автентичності та безпеки відеоінформації.

Однак розробники алгоритмів обробки захищених зображень натикаються на деякі складнощі, основної серед яких є вплив зовнішніх та внутрішніх шумів.

Розглянемо можливу схему додавання (вбудовування) ЦВЗ до відео-потоку (див. рис.1).

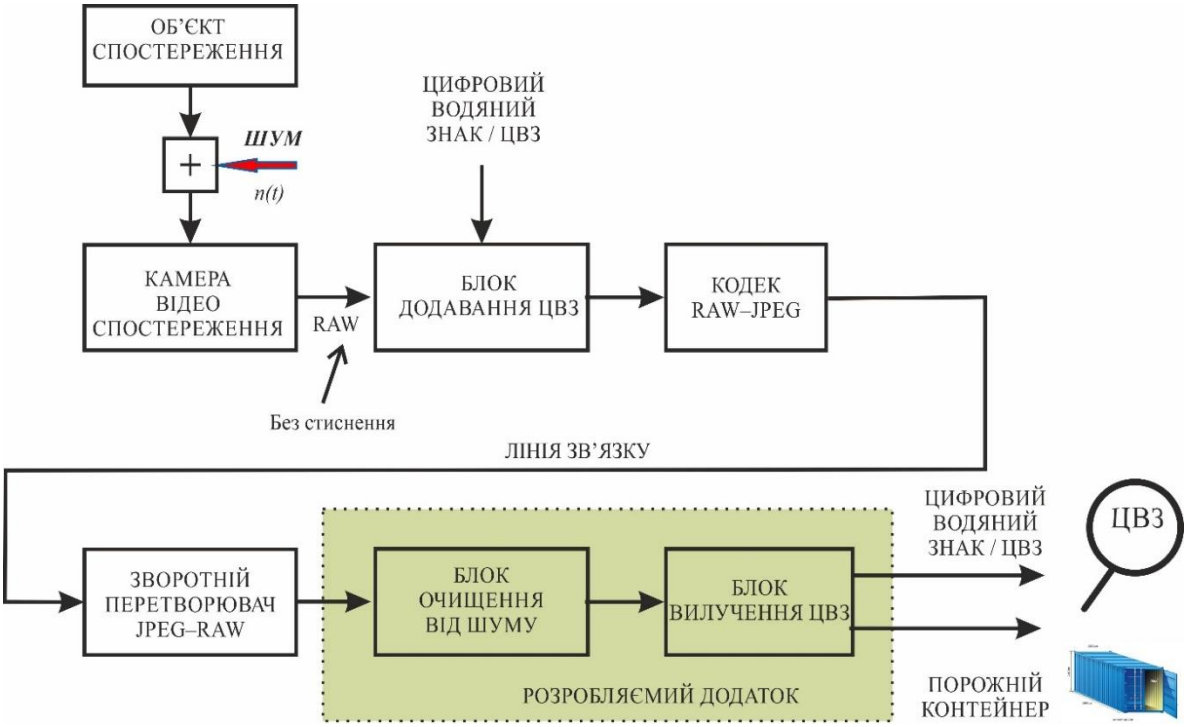


Рис. 1. Схема додавання ЦВЗ до відеопотоку з камери відеоспостереження

Із даного рисунку можна побачити, що на вхід камери спостереження поступає не тільки зображення цільового об'єкта но і шуми, що додаються до цього зображення. Цей процес на схемі зображений у блоці підсумовування, де шум позначений як n(t).

Серед типів шумів можна виділити зовнішні:

- гаусов шум – випадкові зміни яскравості пікселів, що розподілені за нормальним законом,
- шум Спарс (імпульсний шум) – різкі зміни значень пікселів, як, наприклад, "сіль та перець", що виявляються як чорні та білі точки на зображенні,
- та внутрішні:
- флуктуаційні шуми через недосконаліть сенсорів, що мають прояв як "зернистість" зображення.

Для коректного додавання ЦВЗ, особливо адитивними методами, вихідний сигнал камери повинен бути у форматі без стиснення, чи у RAW форматі.

RAW формат (Raw — «сирий») [4] – це нескомпресованій або слабко стислий формат файлів, який використовується для зберігання зображень, отриманих із цифрових камер. Він зберігає дані, отримані з сенсора камери, без значних змін та втрат. Саме до зображення у форматі RAW здійснюється додавання ЦВЗ за адитивним алгоритмом, наприклад, LSB [5]. Далі відбувається стиснення відеопотоку чи окремих, вже захищених, зображень в формат jpg[4] щодо подальшого розповсюдження по каналу

зв’язку убік отримувача повідомлення. Вже на цьому етапі можливі пошкодження ЦВЗ, так як формат jpg – це стиснення із втратами. В подальшому, на боці отримувача повідомлень, здійснюється зворотне перетворення стисненого відеопотоку чи окремих зображень до нестисненого вигляду з ціллю зниження рівня шумів у блоці «очищення від шуму» та вилучення ЦВЗ у відповідному модулі. При декодування формату jpg та обробці захищеного зображення у блоці «очищення від шуму» можливе додаткове спотворення ЦВЗ яке унеможливить його ідентифікацію.

В такому сенсі вибір правильного підходу для очищення захищеного зображення від шумів таким чином, щоб вилучений ЦВЗ мав припустимий рівень спотворень є актуальним завданням.

Мета і задачі дослідження. Мета роботи полягає в підвищенні рівня розбірливості вилученого ЦВЗ одночасно зі зниженням рівня шуму щодо зображення графічного контейнера.

Для досягнення цієї мети необхідно виконати наступні задачі.

1. Провести аналіз типів шумів, що оказують найбільший вплив на якість захищених зображень на виході камери відеоспостереження.
2. Вивчити найбільш відомі підходи до фільтрації шумів на цифрових зображеннях.
3. Дослідити ефективність та придатність ти чи інших алгоритмів зниження шуму щодо вихідного сигналу камери відеоспостереження.
4. Розробити удосконалений алгоритм зниження шуму щодо захищеного зображення та дослідити його ефективність за критерієм максимуму коефіцієнта кореляції між вихідними та обробленими зображеннями ЦВЗ та контейнеру.

Виконання цих задач дозволить досягти поставленої мети підвищення рівня розбірливості ЦВЗ та одночасного поліпшення якості зображення контейнера.

Основна частина. Застосування фільтрів для видалення шуму має враховувати тип шуму та його розподіл, щоб мінімізувати втрату важливої інформації.

1. Лінійні фільтри: наприклад, фільтри Гауса, які зменшують шум, але можуть розмивати контури.
2. Нелінійні фільтри: наприклад, медіани, які добре працюють при видаленні імпульсного шуму, зберігаючи контури, однак додатково виникає проблема втрати дрібних деталей.
3. Вейвлет-фільтри: дозволяють виділяти різні частотні компоненти зображення та усувати шум на певних рівнях.
4. Адаптивні фільтри: такі як фільтри, які використовують інформацію про статистику локального контексту, можуть враховувати зміни у зображенні для покращення результатів.

Додаткові проблеми в задачах фільтрації:

1. Збереження деталей зображення.
   Один із основних викликів очищення зображень від шуму – це необхідність балансувати між видаленням шуму та збереженням деталей зображення. Занадто агресивне зменшення шуму може призвести до розмиття або втрати важливих структур, таких як межі об'єктів, текстури або дрібні деталі. Це особливо важливо для завдань, де точність та збереження дрібних деталей критичні, наприклад, у медичній візуалізації чи розпізнаванні об'єктів.
2. Артефакти.
   Деякі методи фільтрації можуть призвести до появи артефактів, таких як "втрата текстури", коли алгоритми намагаються покращити зображення, але починають помилково інтерпретувати текстури чи структури. Артефакти можуть виникнути при використанні низькоякісних моделей або алгоритмів, які не враховують контекст зображення.

3. Обчислювальна складність.
   Часто методи очищення зображень потребують значних обчислювальних ресурсів, особливо якщо вони ґрунтуються на складних алгоритмах, таких як нейромережні підходи або фільтри, що використовують адаптивні методи. Це може бути проблемою для реального часу обробки зображень або в умовах обмежених обчислювальних потужностей.

Сучасні методи очищення зображень все частіше використовують штучний інтелект та машинне навчання, такі як нейронні мережі (наприклад, автоенкодери, GAN–мережі та інші методи глибокого навчання), які навчаються на великій кількості даних та можуть ефективно відокремлювати шум від корисної інформації; методи на основі статистики (наприклад, алгоритми, що базуються на Байєсівських підходах), які оцінюють ймовірність того, що кожен піксель зображення є шумом або корисною інформацією.

Таким чином можна зробити висновок, що очищення зображень від шуму – це багатозадача проблема, що включає вибір відповідного методу фільтрації, налаштування параметрів для мінімізації втрат якості та обчислювальні витрати. З кожним роком з'являються все ефективніші підходи, в тому числі на основі штучного інтелекту, але завдання залишається актуальним, особливо для додатків у областях, де якість зображення критична.

Розглянемо наступний алгоритм щодо аналізу спотворень зображення контейнеру та ЦВЗ на тлі імпульсного шуму, що згідно попередньому аналізу оказує найбільший вплив на якість відновлених зображень.

Крок 1. Отримуємо вихідне зображення *J*, наприклад таке як показано на рис.2а.

а)вихідне зображення

б) ЦВЗ

в) вихідне зображення з доданим ЦВЗ

Рис. 2. Зображення з камери відеоспостереження та ЦВЗ

Крок 2. Визначаємо розмір вихідного зображення *J* по рядкам lines та стопчикам colums:

$$[lines,colums] = size(J);$$

Крок 3. Змінюємо масштаб ЦВЗ (DWM) у відповідності до розміру вихідного зображення, що знайдений на кроці 2:

$$DWM = imresize(DWM,[lines\ colums]);$$

Отримаємо зображення ЦВЗ, як показано на рис 2 б).

Крок 4. Бінарізуємо ЦВЗ:

$$DWM = gray2bin(DWM).$$

Крок 5. Зменьшуємо максимальне значення яскравості вихідного зображення на 5% з метою додавання ЦВЗ:

$$J = 0.95 \cdot J.$$

Крок 6. Додаємо ЦВЗ:

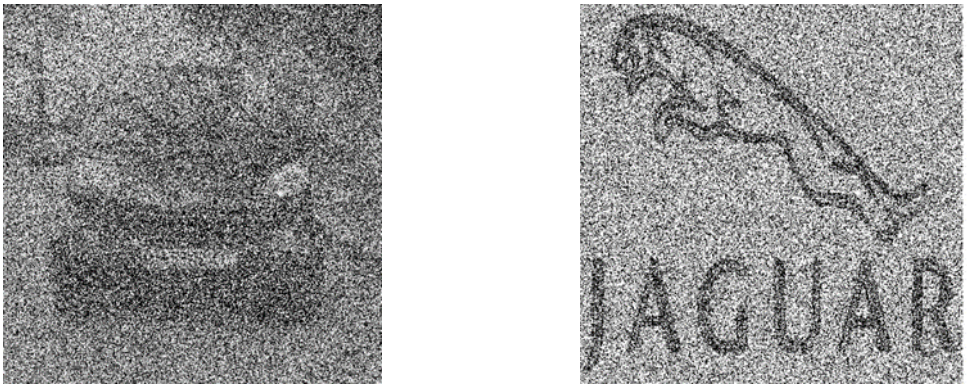
$$Jmix = J + DWM.$$

Отримаємо захищене зображення.

Крок 7. Імітуємо наявність шуму з дисперсією $\sigma^2 = 0.5$:

$$Jmix\_noise = Jmix + n(\sigma).$$

Вигляд захищеного зображення з шумом показаний на рис.3а.



а) захищеное зображення з шумом б) вилучене ЦВЗ з шумом

Рис. 3. Зображення з камери відеоспостереження при наявності шуму з дисперсією $\sigma^2 = 0.5$

Крок 8. Витягуємо зображення ЦВЗ:

$$DWM\_extraction = Jmix\_noise - J.$$

Вигляд вилученого ЦВЗ з шумом показаний на рис.3б.

Як можна побачити з рисунка 3, якість зображень не дозволяє визначити навіть марку автомобіля не кажучи вже про його номер.

Крім суб'єктівного (візуального) критерію оцінки якості відтвореного зображення зручно використовувати критерій максимуму коєфіцієнта кореляції між відтвореними та вихідними зображеннями.

Для розглянутих зображень коефіцієнти кореляції складають:

- для зображення контейнеру R=0.32,
- для зображення ЦВЗ R=0.11.

Для зниження рівня шуму будемо використовувати медіанний алгоритм фільтрації зображень. Опишемо алгоритм медіанної фільтрації.

Крок 1. Визначення вікна.

Встановлюється розмір вікна (зазвичай непарний), наприклад, *3 x 3* або *5 x 5*. Це вікно переміщатиметься за зображенням.

Крок 2. Пробіг по пікселям.

Для кожного пікселя зображення:

- визначається розташування вікна, центрованого на поточному пікселі;
- у вікні вибираються всі пікселі, включаючи поточний.

Крок 3. Сортування значень.

Значення яскравості (інтенсивності) пікселів у вікні сортуються у порядку зростання інтенсивності.

Крок 4. Знаходження медіани.

Знаходиться медіана із відсортованих значень:

- Якщо кількість значень непарна, медіаною буде середнє значення.
- Якщо кількість значень парна, медіаною може бути обрано одне з двох середніх значень.

Крок 5. Заміна значення.

Значення поточного пікселя замінюється на знайдену медіану.

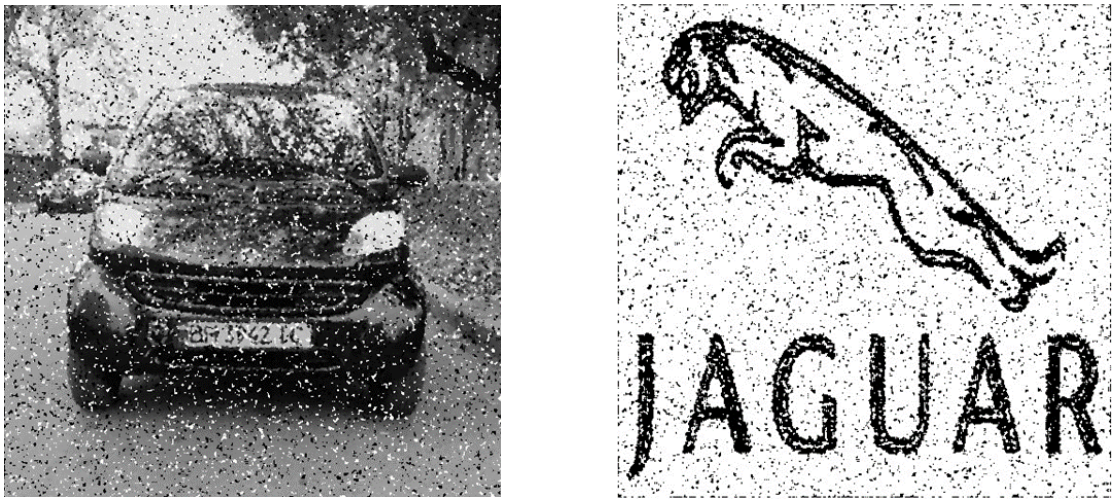
Крок 6. Повторення.

Процес повторюється для кожного пікселя у зображенні, крім меж (де вікно виходить за межі).

Крок 7. Обробка меж.

Для обробки пікселів на краях зображення можна використовувати різні стратегії, наприклад копіювати значення крайових пікселів або використовувати зменшені розміри вікон.

Результати щодо використання даного алгоритму з вікном [3x3] показані на рисунку 4.



а) зображення контейнеру б) ЦВЗ

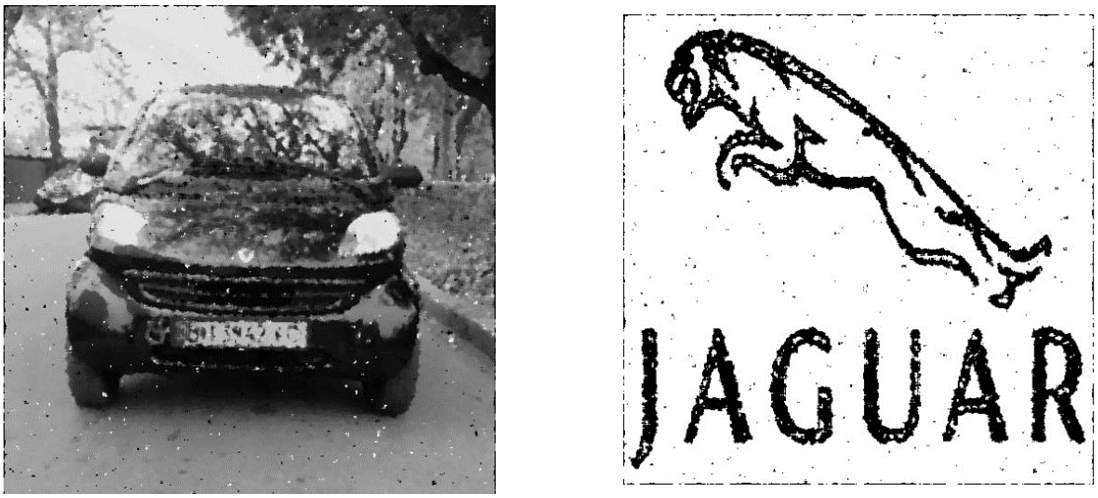
Рис. 4. Зображення контейнеру та ЦВЗ на виході медіаного фільтру з вікном [3x3] при наявності шуму з дисперсією σ^2 =0.5

Для розглянутих зображень коефіцієнти кореляції після виконання алгоритму медіанної фільтрації складають:

- для зображення контейнеру R=0.76,
- для зображення ЦВЗ R=0.51,

що значно краще ніж до застосування алгоритму фільтрації.

Збільшимо вікно (апертуру) медіанного фільтру до [5x5] та виконаємо процедуру оброблення зображень.



а) зображення контейнеру б) ЦВЗ

Рис. 5. Зображення контейнеру та ЦВЗ на виході медіаного фільтру з вікном [5x5] при наявності шуму з дисперсією σ^2 =0.5

Коефіцієнти кореляції після виконання алгоритму медіанної фільтрації з вікном [5x5] складають: для зображення контейнеру R=0.92; для зображення ЦВЗ R=0.78, що вище ніж для випадка фільтрації з вікном [3x3].

Отриману процедуру очищення зображення контейнеру та ЦВЗ від імпульсного шуму можна представити у вигляді наступної блок-схеми, рис. 6.

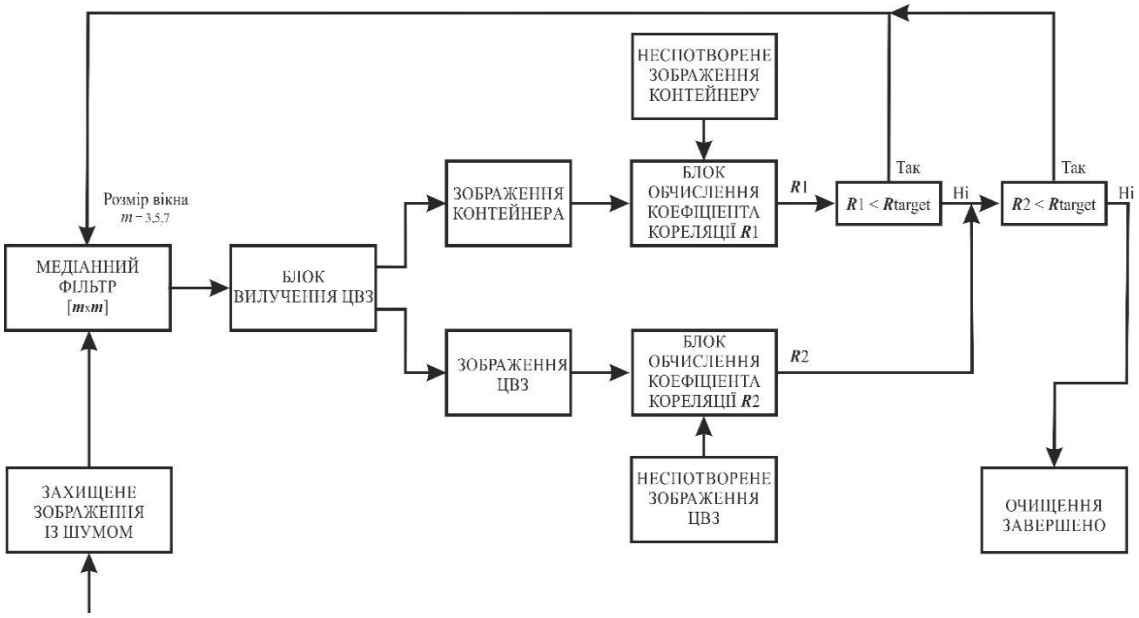


Рис. 6. Процедура очищення зображення контейнеру та ЦВЗ від імпульсного шуму

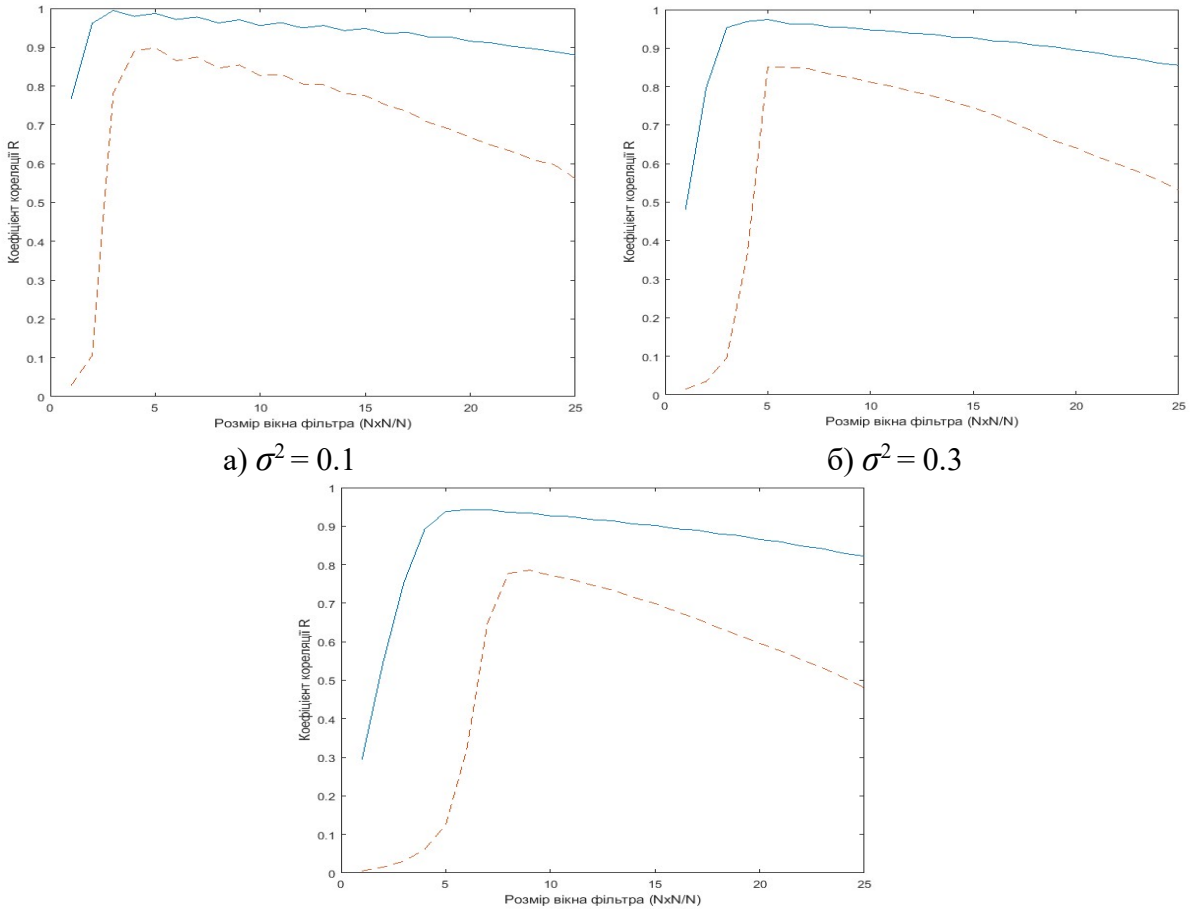
В даної процедурі захищене зображення спочатку піддається медіанної фільтрації з мінімальним вікном [3x3], а далі, після вилучення ЦВЗ, відбувається обчислення коефіцієнтів кореляції (*R*1 та *R*2) між вихідними та відновленими зображеннями з метою встановлення факту чи є очищення від шуму достатнім для подальшого використання зображень. Якщо хоч один з обчислених коефіцієнтів кореляції меншій за цільове значення, то потрібно збільшити розмір вікна медіанного фільтру до значення [5x5] і повторити аналіз якості знов. Максимальне значення вікна медіанного фільтру не може перевищувати [*Integer(0.02lines), Integer(0.02colums)*], де *lines* – кількість рядків, а *colums* – кількість стовпчиків вихідного зображення. *Integer(k)* – ціла частина числа k. Також слід зауважити, що розмір вікна медіанного фільтру повинен бути непарним числом.

Як можна побачити із аналізу роботи алгоритму медіанної фільтрації щодо захищеного зображення поліпшення коефіцієнтів кореляції для зображення контейнеру та для зображення ЦВЗ відбувається за різними законами та з різною швидкістю. Це відбувається завдяки різної кількості інформації, що міститься в зображені контейнеру та в зображенні ЦВЗ, не дивлячись на їхнє однакове розрізнення по рядкам та стовпчикам. Тому основний недолік запропонованої процедури очищення захищеного зображення від шуму – це відсутність можливості одночасно забезпечити рівний рівень якості оброблених зображень.

Для пошуку оптимального значення розміру вікна медіанного фільтру побудуємо графікі залежності коефіцієнту кореляції *R* від розміру вікна [*N*x*N*] при фіксованому значені дисперсії шуму, рис. 7.

З рисунку 7 можна побачити, що оптимальні значення розміру вікна медіанного фільтру щодо зображення контейнеру та щодо зображення ЦВЗ відрізняються. Так, наприклад, при дисперсії шуму $\sigma^2 = 0.1$, оптимальний розмір вікна для зображення контейнера буде [3x3], а для зображення ЦВЗ - [5x5], при дисперсії шуму $\sigma^2 = 0.3$ – оптимальні розміри вікна співпадають для обох зображень- [5x5], при дисперсії шуму $\sigma^2 = 0.5$ оптимальні значення вікна [5x5] та [7x7] відповідно.

Модифікуємо процедуру очищення захищеного зображення від шуму, як показано на рисунку 8.



а) $\sigma^2 = 0.1$ б) $\sigma^2 = 0.3$

в) $\sigma^2 = 0.5$

Рис. 7. Графікі залежності коефіцієнту кореляції *R* від розміру вікна [*N*x*N*] медіанного фільтру при фіксованому значені дисперсії шуму σ^2, де на графіке позначається зображення контейнеру як «—», а зображення ЦВЗ «---»

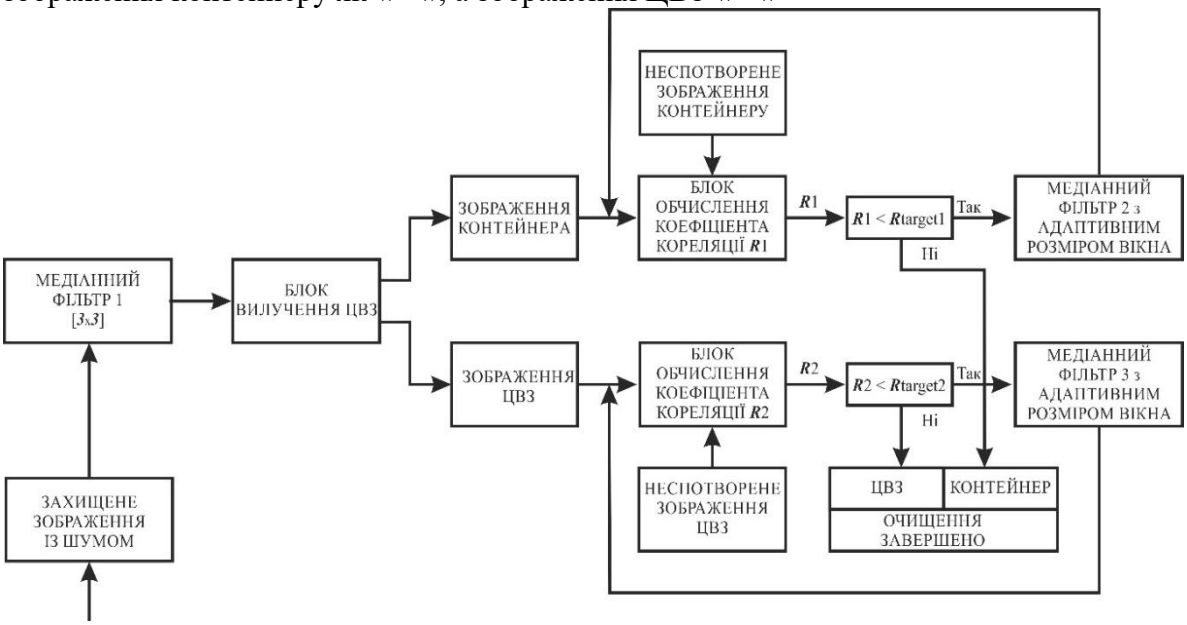
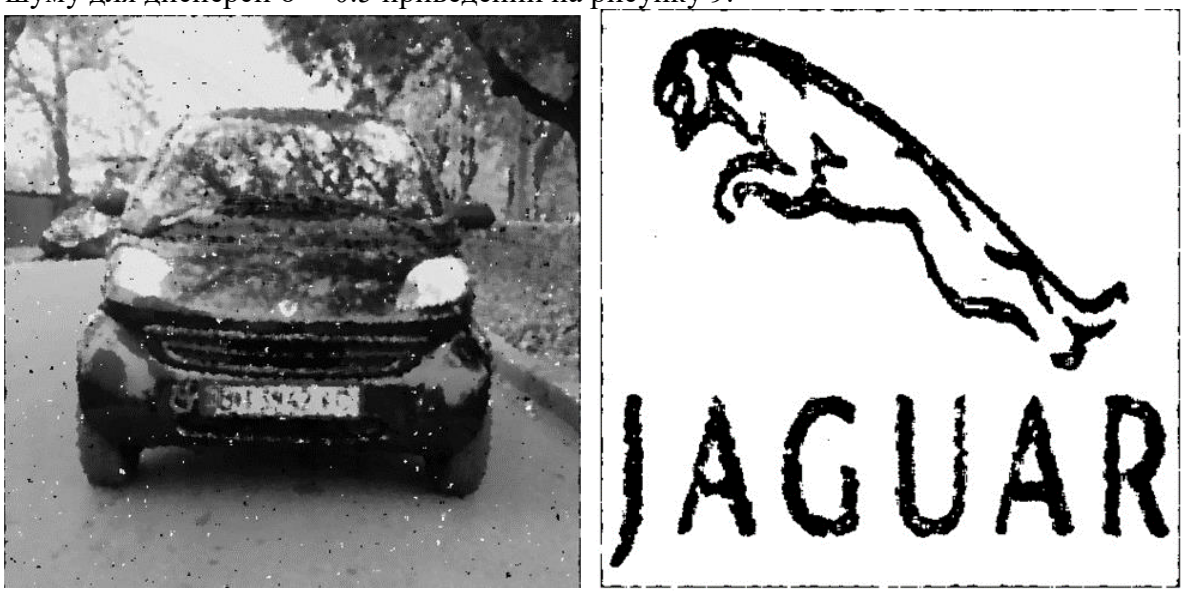


Рис. 8. Модифікована процедура очищення зображення контейнеру та ЦВЗ від імпульсного шуму з трьома медіанними фільтрами

Особливості модифікованої процедури очищення захищеного зображення від шуму полягають у наступному. Спочатку захищене зображення проходить скрізь медіанний фільтр з мінімальним вікном [3x3], далі відбувається вилучення ЦВЗ у

відповідному блоці і кожне зображення окремо піддається медіанної фільтрації з адаптивним розміром вікна, що вибирається виходячи з максимально досяжного значення коефіцієнту кореляції.

Результат роботи модифікованої процедури очищення захищеного зображення від шуму для дисперсії σ^2 =0.5 приведений на рисунку 9.



а) зображення контейнеру, $\max\{R1\}$=0.94

б) ЦВЗ, $\max\{R2\}$=0.82

Рис. 9. Зображення контейнеру на виході медіаного фільтру з вікном [5x5], та ЦВЗ на виході медіаного фільтру з вікном [7x7] при наявності шуму з дисперсією σ^2 =0.5

Висновки. Таким чином, запропонований адаптивний алгоритм очищення від шуму захищеного зображення з камери відеоспостереження дозволяє обрати найкращі параметри медіанних фільтрів окремо для зображення ЦВЗ і зображення контейнеру за критерієм максимумів коефіцієнтів кореляції. Даний алгоритм може бути використаний при наявності у камери відеоспостереження режиму видачі зображення у нестисненому вигляді, чи RAW-форматі.

Список літератури

1. Термінологія в галузі захисту інформації в комп'ютерних системах від несанкціонованого доступу. НД ТЗІ 1.1-003-99. URL: https://tzi.ua/assets/files/1.1_003_99.pdf
2. Садченко А. В., Кушниренко О. А., Троянский А. В., Савчук Ю. А. Адаптивный алгоритм снижения уровня импульсного шума на изображениях с камер видеонаблюдения. *Технология и конструирование в электронной аппаратуре*. 2021. № 1 – 2. С. 21 – 27. URL: https://doi.org/10.15222/TKEA2021.1-2.21
3. Яремчук Ю. Є., Карпінець В. В., Зоря І. С., Козак Д. О. Підвищення стійкості цифрових водяних знаків у потокових відеозаписах на основі диференціального вбудовування енергії (DEW). *Вісник Вінницького політехнічного інституту*, 2023. № 1. С. 55 – 64. URL: https://doi.org/10.31649/1997-9266-2023-166-1-55-64
4. Gonzalez R.C., Woods R.E. Digital Image Processing. New York: Pearson, 2017. 1192 p.
5. Садченко А. В, Кушніренко О. А., Кушніренко Н. П. Модифікований адитивний метод вбудови цифрового водяного знаку. *Труди XXI МНПК «Сучасні інформаційні та електронні технології»*, Одеса. 2020. С. 21 – 23.

А. В. Садченко, О. А. Кушніренко, М. М. Іжак, В. В. Громов, В. О. Назаров

ALGORITHM OF ADAPTIVE CLEANING FROM NOISE OF PROTECTED IMAGES FROM VIDEO SURVEILLANCE CAMERAS

A. V. Sadchenko, O. A. Kushnirenko, V. V. Gromov, M. M. Izhak, V. O. Nazarov

National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
Email: koa@op.edu.ua

The use of watermarks compatible with the video stream from video surveillance cameras allows not only to solve standard tasks such as, for example, confirming the authenticity of the image, copyright protection, but also adds additional information that is convenient to use in the automatic formation of registers and databases. As additional information, there may be personal data of drivers, numbers and parking time of vehicles, if the camera is installed in the parking lot, or the time of the road accident, if the camera is designed to record such events. Taking into account the unstable conditions in which video recording of events can take place, and the imperfection of the technical component of the electronic circuits of the cameras, there is a need to reduce the noise level on the protected image. The purpose of the work is to develop an adaptive and low-computational algorithm for de-noising both the original image of the container and the image of the DWM. Usually, the image of the container has a much higher entropy than the DWM, which leads to a paradox, the essence of which is that the optimal filtering algorithm for the image of the container may turn out to distort the image of the DWM and vice versa. Therefore, the search for an algorithm that can equally improve both components of a protected image is an urgent task, which was solved in this work. The value of the correlation coefficient between the noise-free reference image and the image after filtering was chosen as the main criterion for evaluating the quality of images after noise removal. In order to reduce the computational complexity of the algorithm, the principle of median filtering with an aperture, the size of which depends on the noise dispersion, was chosen. In order to find the optimal values of the aperture size of the median filter, the dependence of the correlation coefficients of the original and processed images on the noise dispersion was constructed. This study made it possible to identify extrema, which are the points of optimal values of the window sizes of the median filter. An adaptive median filtering algorithm was proposed, which first processes the protected image with a fixed window size (3x3), and then, after extracting the DWM, two more median filters with variable window values are used separately for the image of the container and the DWM. This approach made it possible to achieve a correlation coefficient value of more than 0.9 with a noise variance of 0.3. Due to the fact that the median filter contains neither multiplication operations nor summation operations, the proposed algorithm can be implemented on a cheap element base, for example, microcontrollers of AVR or ARM architecture, which cost much less than specialized signal processors or high-speed programmable logic.

Keywords: digital watermark, protected image, noise removal, video surveillance camera, information distortion level, least significant bit.

DOI 10.15276/imms.v14.no3.215
УДК 006.91:620.1:678

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 215-225

МОДЕЛЮВАННЯ ВНУТРІШНІХ ПРОЦЕСІВ В НЕМЕТАЛЕВИХ ГЕТЕРОГЕННИХ МАТЕРІАЛАХ ПРИ АКУСТИЧНОМУ ІНФРАЧЕРВОНОМУ ТЕРМОМЕТРИЧНОМУ МЕТОДІ КОНТРОЛЮ

В.М.Тонконогий[1], М.О.Голофєєва[2], Ю.О.Морозов[3], Р.В.Горбатюк[4]

Національний університет «Одеська політехніка»
1, Шевченка пр., м. Одеса, 65044, Україна
Email: vmt47@ukr.net[1]; mgolofeyeva@gmail.com[2]; morozov@op.edu.ua[3]; ruslan.gorbatiuk@stud.op.edu.ua[4]

Однією з тенденцій розвитку промисловості є заміна традиційних конструкційних матеріалів неметалевими гетерогенними структурами. Це дає можливість отримувати матеріали із запланованими властивостями. При цьому необхідно вивчати механізми організації структури таких матеріалів на кожному масштабному рівні. Тут значну роль відіграють неруйнівні методи контролю. Для виробів з гетерогенних матеріалів перспективними є методи активного термічного контролю. Представляють інтерес процеси в матеріалах цього класу при їх контролі акустичним інфрачервоним термометричним методом. Можна зазначити, що специфіка застосування методу неруйнівного контролю, а також фізичні явища, які виникають в неметалевих гетерогенних матеріалах під час вібраційного впливу на них, не вивчені в повному обсязі для його практичного застосування. Гетерогенність структури, з одного боку, дозволяє створювати матеріали з широким діапазоном властивостей, а з іншого – ускладнює описання процесів, що викликають проявлення зазначених властивостей. Стаття присвячена математичному моделюванню цих процесів. Для комплексного аналізу міцнісних характеристик необхідно враховувати зв’язок термопружних полів, тобто необхідно одночасно визначати температурні та деформаційні поля. Розглянутий трансверсально-ізотропний простір, що можна представити як призму, яка складається із решітки із трьох взаємно перпендикулярних стрижнів, розташованих вздовж координатних осей x, y та z, причому взаємно перпендикулярна сітка у двох напрямках має однакові розміри. Всередині цього простору на довільній кусочно-безперервній поверхні розташовані дефекти структури будь-якої природи.

Вступ. Тенденцією розвитку різних галузей промисловості є широке використання неметалевих гетерогенних матеріалів. Велика різноманітність типів структур і складових компонентів дозволяє отримувати середовища із заздалегідь запланованими властивостями. Звичайно, ефективне використання матеріалів зазначеного типу, а також виробів з них неможливе без достовірного вимірювання параметрів процесів у них на всіх етапах життєвого циклу продукції.

Якщо розглядати об’єкт із неметалевого різнорідного матеріалу з точки зору метрологічного забезпечення якості, то його можна трактувати як складний. Найбільш перспективними для контролю виробів з неметалевих гетерогенних матеріалів є неруйнівні методи. У той же час існують проблеми з їх використанням, пов'язані з рядом особливостей і фізичних явищ, характерних для цього класу матеріалів. Серед них слід виділити немагнітність, низьку електропровідність, взаємодію ультразвукових хвиль з елементами армування, високі характеристики демпфування. Серед найбільш перспективних методів, що дозволяють виявити особливості будови неметалевого гетерогенного матеріалу, можна виділити активні термічні методи, що базуються на аналізі аномалій на термограмах поверхонь об’єктів дослідження, які можуть бути ознаками наявності відхилень в об'ємі матеріалу. У цьому випадку інформація про виявлені особливості будови неметалевого гетерогенного матеріалу міститься в амплітудно-часових характеристиках отриманого температурного поля.

Представляють інтерес внутрішні процеси, що відбуваються в матеріалах цього класу при їх контролі акустичним інфрачервоним термометричним методом. Специфіка застосування зазначеного методу неруйнівного контролю, а також фізичні явища, які виникають у неметалевих гетерогенних матеріалах під час вібраційного впливу на них, вивчені не в повній мірі для їхнього практичного застосування. Для комплексного аналізу міцнісних характеристик необхідно враховувати зв'язок термопружних полів, тобто необхідно одночасно визначати температурні та деформаційні поля.

Аналіз досліджень і публікацій. В роботі [1] представлена математична модель, що дозволяє отримати значення потужності нагріву тріщини, яка розташована перпендикулярно напрямку розповсюдження механічних коливань. В [2] наведений вираз, що дозволяє визначити вклад пружної енергії системи в залежності від тріщини, що утворилася. Недоліками цих моделей є те, що вони не враховують неоднорідність досліджуваного середовища.

В [3] запропонована модель, що призначена для прогнозування термомеханічного відгуку трансверсально-ізотропної термопружної тонкої прямокутної пластини при гармонійному зосередженому навантаженні з плином часу. Ця математична модель розроблена із застосуванням теорії пластини Кірхгофа для нелокальної узагальненої термопружності та теорією термопружності Гріна-Нагді. Для знаходження виразів для бічного відхилення, теплового моменту та розподілу температури для тонкої прямокутної пластини з простими опорами у трансформованій області було використано метод подвійного скінченного перетворення Фур'є. Досліджено та показано графічно зміну бічного відхилення, теплового моменту та розподілу температури з нелокальною узагальненою термопружністю та класичною теорією термопружності з різними номерами мод.

Для моделювання процесів, що протікають в неметалевих гетерогенних структурах були розроблені теорії континууму вищого порядку, такі як нелокальна теорія термопружності, теорія модифікованого парного напруження та градієнта деформації. Для вирішення проблем мікро/наноструктури Ерінгеном була розроблена теорія нелокальної механіки континууму [4-7]. В теорії нелокального континууму напружений стан в точці розглядають як функцію станів деформації всіх точок у середовищі. На відміну від цього в класичній механіці суцільного середовища напружений стан в певній точці однозначно залежить від стану деформації в цій самій точці». Лу та ін. [8] запропонували модель нелокальної пластини на основі теорії Ерінгена з використанням нелокальних теорій пластин Кірхгофа та Міндліна. Ця теорія базується на теорії пучка Ейлера–Бернуллі з такими кінематичними міркуваннями:

- товщина плити має відносно малі бічні розміри і не змінюється при деформації;
- немає деформації в середній площині пластини;
- компоненти зміщення середини поверхні є малими порівняно з товщиною пластини;
- можна знехтувати поперечною деформацією зсуву, поперечним нормальним напруженням і деформацією.

Залежно від моделі нелокальної теорії Ерінгена, Лазар і Агіасофіту [9] отримали нові результати для компонентів напружень гвинтових переміщень в анізотропних матеріалах. Зенкур [10] використав теорію нелокальної пружності для дослідження одношарового графенового листа (SLGS), закріпленого у в'язкопружному середовищі, за допомогою вібраційного аналізу. Шаат [11] обговорював залежні від розміру нанопластини Кірхгофа з модифікованими парними напруженнями. Тріпаті та ін. [12] досліджували тонку круглу пластину за допомогою квазістатичної теорії незв'язаної термопружності з використанням дробового рівняння теплопровідності. Марін [13] обговорював проблему коливань термопружності диполярних тіл. В роботі [14] досліджували теорію термопружності Гріна-Нагді для диполярного тіла, щоб довести стабільність типу Гельдера за допомогою змішаного початкового BVP. Крім того, деякі

дослідники, такі, як Шарма [15], Марін та ін. [16 - 18], Жанг та Фу [19], Аббас та Мартін [20], Каур [21], Зенкор та Абулегар [22], Кумар та Деві [23], Предхан та Фадікан [24, 25], Бхаті [26], Лата [27, 28], Махакалкар та ін. [29, 30], Гайквад і Дешмуд [31], Шен [32], Нгуен та ін. [33-38], Вентзел [39] працювали над мікро/нано технологіями, використовуючи різноманітні теорії термоеластичності. З аналізу видно, що нелокальна теорія термопружності має значний вплив на всі параметри.

Методика досліджень. Більшість розглянутих робіт розв'язувалися в такій постановці, коли фізико-механічні характеристики матеріалу вважалися такими, що не залежать від температури. Проте, для всебічного аналізу характеристик міцності необхідно враховувати зв'язаність термопружних полів, тобто необхідно одночасно визначати поля температури та деформації.

Розглянемо трансверсально-ізотропний простір, всередині якого на довільній кусочно-безперервній поверхні Ω розташовані дефекти структури будь-якої природи (типу тріщин, відшарувань або сторонніх включень).

Згідно із [40] такий простір можна представити як призму, що складається із решітки із трьох взаємно перпендикулярних стрижнів, розташованих вздовж координатних осей *x*, *y* та *z*, причому взаємно перпендикулярна сітка у двох напрямках має однакові розміри (рис. 1).

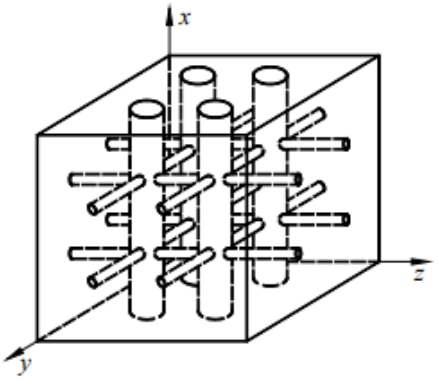


Рис.1. Трансверсально-ізотропний матеріал

Таке уявлення трансверсально-ізотропного тіла, звичайно, умовне і служить лише зручною формою наочного представлення континууму, яке має, як бачимо з рисунку 1 ізотропність властивостей по координатам *y* та *z*.

Компоненти напружень та переміщень в цьому випадку будуть [41]:

$$\begin{aligned}\boldsymbol{\sigma} &= \left\{\sigma_k\left(x,y,z\right)\right\}_{k=1}^{6} = \left\{\sigma_x,\sigma_y,\sigma_z,\tau_{xy},\tau_{yz},\tau_{xz}\right\},\\ \mathbf{u} &= \left\{u_k\left(x,y,z\right)\right\}_{k=1}^{3} = \left\{u,v,w\right\},\end{aligned} \tag{1}$$

При $\left(x,y,z\right)\notin\Omega$ задовольняють рівнянням Дюамеля-Неймана:

$$\begin{aligned}&\partial_k u_k = \sum_{j=1}^{3} s_{jk}\sigma_j + \alpha_k T, k=1,2,3,\ \ \partial_3 u_2 + \partial_2 u_3 = s_{44}\sigma_4,\\ &\partial_1 u_3 + \partial_3 u_1 = s_{44}\sigma_5,\quad \partial_2 u_1 + \partial_1 u_2 = s_{66}\sigma_6,\end{aligned} \tag{2}$$

умовам рівноваги:

$$\begin{aligned}\partial_1\sigma_1 + \partial_2\sigma_6 + \partial_3\sigma_5 + F_1 &= \rho\partial_t^2 u_1\\ \partial_1\sigma_6 + \partial_2\sigma_2 + \partial_3\sigma_4 + F_2 &= \rho\partial_t^2 u_2\\ \partial_1\sigma_5 + \partial_2\sigma_4 + \partial_3\sigma_3 + F_3 &= \rho\partial_t^2 u_3\end{aligned} \tag{3}$$

та рівнянню теплопровідності:

$$\sum_{j=1}^{3}\lambda_j\partial_j^2 T - c_\varepsilon\partial_t T - T\partial_t\sum_{j=1}^{3}\beta_j\partial_j u_j = 0. \qquad (4)$$

де $\partial_k = \dfrac{\partial}{\partial x_k}, k=1,2,3,\ \partial_t^k = \dfrac{\partial^2}{\partial t^k}, k=1,2$, $F_k, k=\overline{1,3}$ – складові об'ємних сил;

s_{kj} – коефіцієнти узагальненого закону Гука трансверсально-ізотропного середовища;

λ_i – коефіцієнти теплопровідності;

α_k – коефіцієнт теплового розширення;

c_ε – питома теплоємність.

$$-\beta_1 = c_{11}\alpha_1 + c_{12}\alpha_2 + c_{13}\alpha_3,$$
$$-\beta_2 = c_{21}\alpha_1 + c_{22}\alpha_2 + c_{23}\alpha_3,$$
$$-\beta_3 = c_{13}\alpha_1 + c_{22}\alpha_2 + c_{33}\alpha_3,$$

На відміну від класичного рівняння теплопровідності, рівняння (4) містить складову, яка пов'язує приріст температури із швидкістю змінення об'єму тіла, що деформується під впливом механічного гармонійного навантаження.

Для запису умов на поверхні Ω, де можливі розриви всіх компонент векторів $\boldsymbol{\sigma}$ та $\mathbf{u}$, введемо в кожній точці поверхні Ω локальну систему координат (N,M,S) (рис. 2).

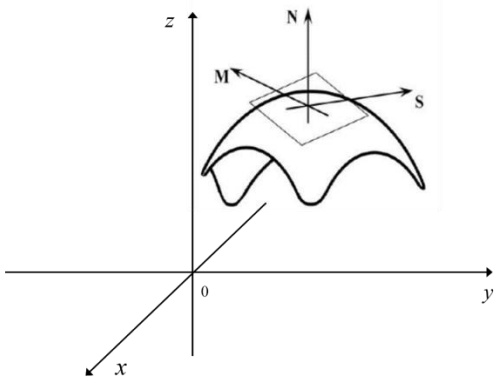


Рис. 2. Локальна система координат (N,M,S) в кожній точці поверхні Ω

Для цього, в кожній точці поверхні проведемо дотичну площину P та нормальний вектор $\mathbf{n}$ до неї. Напрямок осі N співпадає з напрямком вектору $\mathbf{n}$. Дві інші осі M,S оберемо взаємно перпендикулярно в площині P таким чином, щоб після обертання осей, направлення осей (N,M,S) співпадали з направленням відповідних осей (Z,X,Y). Тоді в новій системі координат напруження та переміщення позначимо так:

$$\boldsymbol{\sigma}_N = \{\tilde{\sigma}_k(x,y,z)\}_{k=1}^{6} = \{\sigma_N,\sigma_S,\sigma_Z,\tau_{SZ},\tau_{ZN},\tau_{NS}\},$$
$$\mathbf{u}_N = \{\tilde{u}_k\}_{k=1}^{3} = \{u_M, v_S, w_N\}$$

В залежності від контактної взаємодії з простором на поверхні Ω можуть бути відомі шість із наступних величин:

$$\tilde{\chi}^{\pm}=\left\{\tilde{\chi}_k^{\pm}\right\}_{k=1}^{6},\ \tilde{\chi}_k^{\pm}=\tilde{\zeta}_k^{+}\left(x_1,x_2,x_3\right)\pm\tilde{\zeta}_k^{-}\left(x_1,x_2,x_3\right),\left(x_1,x_2,x_3\right)\in\Omega$$

$$\left\{\tilde{\zeta}_k^{\pm}\right\}_{k=1}^{6}=\{\sigma_3^{\pm}\left(x_1,x_2,x_3\right),\sigma_5^{\pm}\left(x_1,x_2,x_3\right),\sigma_6^{\pm}\left(x_1,x_2,x_3\right),u_1^{\pm}\left(x_1,x_2,x_3\right),u_2^{\pm}\left(x_1,x_2,x_3\right),u_3^{\pm}\left(x_1,x_2,x_3\right),$$

$$q_1^{\pm}\left(x_1,x_2,x_3\right),q_2^{\pm}\left(x_1,x_2,x_3\right),q_3^{\pm}\left(x_1,x_2,x_3\right),T\left(x_1,x_2,x_3\right)\},$$

$$\left(\sigma_k^{\pm},u_k^{\pm},q_k^{\pm},T^{\pm}\right)=\lim_{x_i^{\pm}\to x_i}\left(\sigma_k^{\pm}\left(x_1^{\pm},x_2^{\pm},x_3^{\pm}\right),u_k^{\pm}\left(x_1^{\pm},x_2^{\pm},x_3^{\pm}\right),q_k^{\pm}\left(x_1^{\pm},x_2^{\pm},x_3^{\pm}\right),T^{\pm}\left(x_1^{\pm},x_2^{\pm},x_3^{\pm}\right)\right);i=\overline{1,3}$$

Точка $\left(x^{+},y^{+},z^{+}\right)$ знаходиться збоку нормалі $\mathbf{n}$, а точка $\left(x^{-},y^{-},z^{-}\right)$ - з протилежного боку. Для усунення невизначеності, на поверхні Ω будемо вважати відомими наступні стрибки:

$$\tilde{\chi}_k^{-}=\tilde{\zeta}_k^{+}\left(x,y,z\right)-\tilde{\zeta}_k^{-}\left(x,y,z\right),\ k=\overline{1,10},\ \left(x,y,z\right)\in\Omega\ . \tag{5}$$

Розв'язання крайової задачі (1 – 3), (4) та (5) необхідно шукати в класі $C_{0,5}{}^{1}(\Re^{3})\cap L_1(\Re^{3})$, де $C_{0,s}^{1}$ – простір функцій, що є безперервними за всіма похідними до m – го порядку за виключенням поверхні Ω;

$L_1(\Re^{3})$ – простір функцій, що інтегровані в $\Re^{3}$.

Температурні умови на дефекті виражають собою умови неідеального контакту між поверхнями особливості структури, а фізично – опір, який здійснює зазначений артефакт на розповсюдження тепла. Ці умови для площини тріщини лежать в площині XOY та мають вигляд [42, 43]:

$$\begin{aligned}&\lambda^{*}\Delta\left(\zeta_4^{+}+\zeta_4^{-}\right)+2\lambda_3\left[\partial_3\zeta_4^{+}\big|_{x_3=h+0}-\partial_3\zeta_4^{-}\right]=0,\\&\left(\lambda^{*}\Delta-12h^{*}\right)\left(\zeta_4^{+}-\zeta_4^{-}\right)+6\lambda_3\left[\partial_3\zeta_4^{+}+\partial_3\zeta_4^{-}\right]=0\end{aligned}\ , \tag{6}$$

де λ^{*} та h^{*} – коефіцієнти, що характеризують теплопроникність дефекту в повздовжньому та поперечному напрямках.

Дефекти, на поверхні яких температура задовольняє вищенаведеним умовам, називають теплопровідними. Якщо $\lambda^{*}=0,h^{*}\neq 0$, то особливості структури називаються теплопроникними. А у випадку, коли $\lambda^{*}=h^{*}=0$, то – теплоізольованими.

Розв'язання задач (1 – 4) з урахуванням умов (5) та (6) дозволяє отримати розподіл температур та механічних напружень поблизу дефекту. Використовуючи методику, що наведена в роботі [44], можна звести проблему до розв'язання системи сингулярних інтегральних рівнянь та отримати розподіл температурних полів поблизу особливостей структури неметалевого гетерогенного матеріалу будь-якої природи (типу тріщин, відшарувань або сторонніх включень).

Введемо позначення:

$$\mathbf{v}=\{\upsilon_k(x,y,z)\}_{k=1,\ldots,9}=\{\sigma_x,\sigma_y,\sigma_z,\tau_{yz},\tau_{xz},\tau_{xy},u,v,w\} \tag{7}$$

$$\mathbf{D}=\begin{Vmatrix}\mathbf{D}_0 & \mathbf{O}_{3\times3}\\ -\mathbf{S} & \mathbf{D}_0^{\cdot}\end{Vmatrix},\ \mathbf{F}^{T}=\mathbf{F}_0{}^{T}+\mathbf{F}_*{}^{T};$$

$$\mathbf{F}_0{}^{T}=\left\{F_{0j}\right\}^{9}=-\delta_0\left\|P_1,P_2,P_3,\mathbf{O}_{1\times6}\right\|;\mathbf{F}_*{}^{T}=\left\{F_{*j}\right\}^{9}=\left\|\mathbf{O}_{1\times3},\beta_1T,\beta_2T,\beta_3T,\mathbf{O}_{1\times3}\right\|;$$

$$\mathbf{S}=\begin{Vmatrix}\mathbf{S}_1 & \mathbf{O}_{3\times3}\\ \mathbf{O}_{3\times3} & \mathbf{S}_2\end{Vmatrix},\qquad \mathbf{D}_0=\begin{Vmatrix}\partial_1 & 0 & 0 & 0 & \partial_3 & \partial_2\\ 0 & \partial_2 & 0 & \partial_3 & 0 & \partial_1\\ 0 & 0 & \partial_3 & \partial_2 & \partial_1 & 0\end{Vmatrix},$$

$$\mathbf{S}_1 = \begin{Vmatrix} s_{11} & s_{12} & s_{13} \\ s_{21} & s_{11} & s_{13} \\ s_{13} & s_{13} & s_{33} \end{Vmatrix}, \qquad \mathbf{S}_2 = \begin{Vmatrix} s_{44} & 0 & 0 \\ 0 & s_{44} & 0 \\ 0 & 0 & s_{66} \end{Vmatrix},$$

$\partial_1 = \dfrac{\partial}{\partial x}$, $\partial_2 = \dfrac{\partial}{\partial y}$, $\partial_3 = \dfrac{\partial}{\partial z}$,

$s_{kj} = \theta(z)s_{kj}^{+} + \theta(-z)s_{kj}^{-}$,

$\delta_0 = \delta(x-x_0, x-x_0, x-x_0)$ – дельта функція Дірака;

s_{kj} – коефіцієнти узагальненого закону Гука;

$\mathbf{O}_{k\times l}$ – нульова матриця розмірності $k \times l$.

Тоді рівняння (1), (2) представимо так:

$$\mathbf{D}[z, \partial_1, \partial_2, \partial_3]\mathbf{v} = F, \qquad \mathbf{v}, \mathbf{F} \in \Im(\Re^3) \tag{8}$$

Відповідно до роботи [41], подання розв'язків системи (3.8) через стрибки (3.6), будемо називати розривним розв'язком для кусково-однорідного анізотропного простору в класі диференційованих функцій.

Продовжимо матричне рівняння (3.8) на весь простір. Для цього, скориставшись властивостями узагальнених функцій повільного зростання $\Im(\Re^3)$, розшукувані функції продовжимо в простір $\Im(\Re^3)$, носієм сингулярності яких є поверхня Ω.

Формули зв'язку між звичайними і узагальненими похідними [44], урахувавши умови (7) представимо так:

$$\partial_k \upsilon_j = \tilde{\partial}_k \upsilon_j - v_j(x,y,z)\kappa_k \delta(\Omega)$$

$$\kappa_1 = \cos(N,X), \kappa_1 = \cos(N,Y), \kappa_1 = \cos(N,Z), \tag{9}$$

де $v_j(x,y,z) = \langle \upsilon_j \rangle_{\Omega}^{-}$, $\delta_{k,j}$ – символ Кронекера, $\delta(\Omega)$ – функція Дірака, яка зосереджена на поверхні Ω.

Врахувавши формули (9) крайову задачу (8), (9) зведемо до наступної крайової задачі у просторі $\Im(\mathrm{R}^3)$.

$$\mathbf{D}[z, \tilde{\partial}_1, \tilde{\partial}_2, \tilde{\partial}_3]\mathbf{v} = \tilde{\mathbf{f}}, \tag{10}$$

де

$$\tilde{\mathbf{f}} = \{\tilde{f}_j\}^9, \tilde{f}_1 = (v_1^{-}\kappa_1 + v_6^{-}\kappa_2 + v_5^{-}\kappa_3)\delta(\Omega) - P_1, \tilde{f}_2 = (v_5^{-}\kappa_1 + v_2^{-}\kappa_2 + v_4^{-}\kappa_3)\delta(\Omega) - P_2$$

$$\tilde{f}_3 = (v_5^{-}\kappa_1 + v_4^{-}\kappa_2 + v_3^{-}\kappa_3)\delta(\Omega) - P_3, \tilde{f}_4 = v_7^{-}\kappa_1\delta(\Omega), \tilde{f}_5 = v_8^{-}\kappa_2\delta(\Omega), \tilde{f}_6 = v_9^{-}\kappa_3\delta(\Omega),$$

$$\tilde{f}_8 = (v_7^{-}\kappa_3 + v_9^{-}\kappa_1)\delta(\Omega), \tilde{f}_9 = (v_7^{-}\kappa_2 + v_8^{-}\kappa_1)\delta(\Omega),$$

$$\tilde{f}_{10} = \lambda_1 v_7^{-}\kappa_1\delta(\Omega), \tilde{f}_{11} = \lambda_2 v_7^{-}\kappa_2\delta(\Omega), \tilde{f}_{12} = \lambda_3 v_7^{-}\kappa_3\delta(\Omega),$$

$$\tilde{f}_{13} = v_{11}^{-}\kappa_1\delta(\Omega) + v_{12}^{-}\kappa_2\delta(\Omega) + v_{13}^{-}\kappa_3\delta(\Omega),$$

Розв'язки поставленої задачі можна подати так:

$$\upsilon_k(x,y,z) = \sum_{j=1}^{13} w_{kj} * \tilde{f}_j =$$

$$= \sum_{j=1}^{13} \int_0^t \iiint_{\square^3} w_{kj}(x,y,z,x_0,y_0,z_0,t-t_0)\tilde{f}_j(x_0,y_0,z_0,t_0)dx_0, dy_0, dz_0 dt, \tag{11}$$

де функції $w_{kj}(x,y,z) \in \Im(\Re^3)$ – компоненти системи фундаментальних розв'язків $\mathbf{w}_j = \{w_{kj}\}_{k=1,\ldots,13}$ задачі (2) – (3), тобто $\mathbf{w}_j$ є розв'язками системи крайових задач:

$$\mathbf{D}[z, \widetilde{\partial}_1, \widetilde{\partial}_2, \widetilde{\partial}_3]\mathbf{w}_j = \mathrm{f}^0, \qquad \mathbf{w}_\mathrm{j}, \mathrm{f}^0 \in \Im(\Re^3) \tag{12}$$

$$w_{kj}(x,y,x)\big|_{(x,y,z)\to\infty} = 0, \qquad k = 1,\ldots,13, \tag{13}$$

де $\mathbf{f}^0 = \{f_{kj}^0\}^{13}\{\delta_{kj}\}_{k=1}^{13}\delta(x-x_0, y-y_0, z-z_0, t-t_0)\}$,

Застосуємо до (12) перетворення Лапласа за часом $\overline{\upsilon}_k(x,y,z,p) = L[\upsilon_k]$ та тривимірне перетворення Фур'є по x, y і z з параметрами $\alpha_1, \alpha_2, \alpha_3$ відповідно $\overline{\mathrm{W}}_{k,j}(\alpha_1,\alpha_2,\alpha_3,p) = \mathrm{F}_3[\overline{\upsilon}_k]$, отримаємо матричну крайову задачу у просторі $\Im(\Re^3)$.

$$\mathbf{M}[-i\alpha_1, -i\alpha_2, -i\alpha_3, p]\mathbf{W}_j = \mathbf{f}_j^*, j = \overline{1,13} \tag{14}$$

де $\mathbf{W}_j = \{\mathrm{W}_{kj}\}^{13}, \mathbf{f}_j^* = \{\delta_{kj}\}^{13} e_0^*, e_0^* = e^{i\alpha_1 x_0 + i\alpha_2 y_0 + i\alpha_3 z_0} e^{-t_0 p}$

Безпосередньо із рівняння дістанемо трансформанти фундаментального розривного розв'язку

$$\mathbf{W}_j = \mathbf{M}^{-1}\mathbf{f}_j^*, j = \overline{1,13} \tag{15}$$

Тут

$$\mathbf{M} = \begin{Vmatrix}
(-i\alpha_1) & 0 & 0 & 0 & (-i\alpha_3) & (-i\alpha_2) & \rho p^2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & (-i\alpha_2) & 0 & (-i\alpha_3) & 0 & (-i\alpha_1) & 0 & \rho p & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & (-i\alpha_3) & (-i\alpha_2) & (-i\alpha_1) & 0 & 0 & 0 & \rho p^2 & 0 & 0 & 0 & 0 \\
s_{11} & s_{12} & s_{13} & 0 & 0 & 0 & (-i\alpha_1) & 0 & 0 & \tilde{\alpha}_1 & 0 & 0 & 0 \\
s_{21} & s_{11} & s_{13} & 0 & 0 & 0 & 0 & (-i\alpha_2) & 0 & \tilde{\alpha}_2 & 0 & 0 & 0 \\
s_{13} & s_{13} & s_{33} & 0 & 0 & 0 & 0 & 0 & (-i\alpha_3) & \tilde{\alpha}_3 & 0 & 0 & 0 \\
0 & 0 & 0 & s_{44} & 0 & 0 & 0 & (-i\alpha_3) & (-i\alpha_2) & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & s_{44} & 0 & (-i\alpha_3) & 0 & (-i\alpha_1) & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & s_{66} & (-i\alpha_2) & (-i\alpha_1) & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & T_0 p & 0 & 0 & \lambda_1(-i\alpha_1) & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & T_0 p & 0 & \lambda_2(-i\alpha_2) & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & T_0 p & \lambda_3(-i\alpha_3) & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -c_\varepsilon p & (-i\alpha_1) & (-i\alpha_2) & (-i\alpha_3)
\end{Vmatrix}$$

Застосуємо до (15) обернене перетворення Лапласа та Фур'є подання для системи фундаментальних розв'язків $\mathbf{w}_j = \{w_{kj}\}_{k=1..13}$, які підставляємо у (11).

Склад вимірювального сигналу розпишемо так (рис.3.3):

$$h(t) = a_0 h_c(t) + b_0 h_d(t), \tag{16}$$

де $h_c(t)$ – когерентна складова;
$h_c(t)$ – дифузійна складова;
a_0, b_0 – коефіцієнти рівняння.

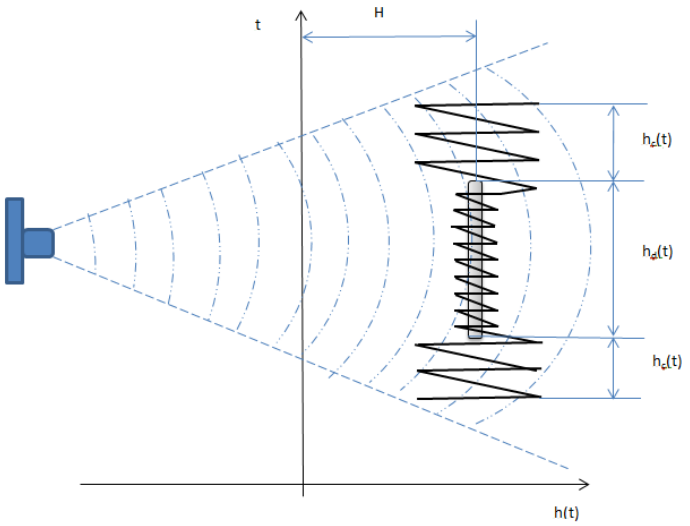


Рис. 3. Когерентні та дифузійні складові вимірювального сигналу

Висновки. Таким чином, при вирішенні проблеми розробки методів вимірювання параметрів процесів, що відбуваються в гетерогенних матеріалах, необхідно спиратися на глибоке розуміння спеціальних розділів математики. Для комплексного аналізу характеристик міцності необхідно враховувати зв'язок термопружних полів, тобто необхідно одночасно визначати температурне та деформаційне поля.

Неметалевий гетерогенний матеріал представлено у вигляді трансверсально-ізотропного простору, всередині якого на довільній кусково-суцільній поверхні розташовані дефекти довільного характеру (такі як тріщини, відшаровані та невідшаровані включення). Для запису умов на поверхні Ω, за яких можливі розриви всіх компонент векторів, введена локальна система координат у кожній точці поверхні Ω. Записані температурні умови для плоскої тріщини, які виражають умови неідеального теплового контакту між поверхнями дефекту, а фізично – опір, який дефект надає поширенню тепла.

Розроблена математична модель термомеханічних процесів, що протікають в неметалевих гетерогенних матеріалах при їх контролі акустичним інфрачервоним термометричним методом, яка відрізняється тим, що приймає до уваги зв'язність механічних та температурних полів з урахуванням неоднорідності структури матеріалу.

Список літератури

1. New developments in thermoelastic stresses. Analysis by infrared thermography. *IV Conferencia Panamericana de END Buenos Aires*. 2007. P. 34-38.
2. Vavilov V.I., Nesteruk D., Khorev VL. Ultrasonic and inductive IR Thermographic Procedures as Newly - Emerged Technigues in Thermal NDT. *Annual Journal of Electronics.* 2012. V. 6. №2. P. 102-109.
3. Kaur I., Lata P., Singh K. Forced Flexural Vibrations in a Thin Nonlocal Rectangular Plate with Kirchhoff's Thin Plate Theory. *International Journal of Structural Stability and Dynamics*. doi: 10.1142/S0219455420501072
4. A.C. Eringen, Theory of nonlocal thermoelasticity, Int. J. Eng. Sci. 12 (1974) 1063–1077. https://doi.org/10.1016/0020-7225(74)90033-0
5. A.C. Eringen, On differential equations of nonlocal elasticity and solutions of screw dislocation and surface waves, J. Appl. Phys. 54 (1983) 4703–4710. https://doi.org/10.1063/1.332803
6. Eringen A.C., Vistas of nonlocal continuum physics, *Int. J. Eng. Sci. 30.* 1992. P.1551–1565. URL: https://doi.org/10.1016/0020-7225(92)90165-D
7. Cemal Eringen A., Nonlocal Continuum Field Theories. New York, NY: Springer, 2004. URL: https://doi.org/10.1007/b97697.

8. Lu P., Zhang P., Lee H., Wang C., Reddy J. Non-local elastic plate theories. *Proc. R. Soc. A Math. Phys. Eng. Sci.* 2007. V.463 P.3225–3240. URL: https://doi.org/10.1098/rspa.2007.1903
9. Lazar M., Agiasofitou E. Screw dislocation in nonlocal anisotropic elasticity. *Int. J. Eng. Sci.* 2011. V.49 P. 1404–1414. URL: https://doi.org/10.1016/j.ijengsci.2011.02.011.
10. Zenkour A.M.. Vibration analysis of a single-layered graphene sheet embedded in visco-Pasternak's medium using nonlocal elasticity theory. *J. Vibroengineering*. 2016. V.18. P.2319–2330. URL: https://doi.org/10.21595/jve.2016.16585
11. Tripathi J.J., Warbhe S.D., Deshmukh K.C., Verma J. Fractional Order Thermoelastic Deflection in a Thin Circular Plate. *Int. J. Applications and Applied Mathematics.* 2017. No. 12. P.898–909.
12. Marin M. Some Estimates on Vibrations in Thermoelasticity of Dipolar Bodies. *J. Vib. Control.* No. 16. P. 33–47. URL: https://doi.org/10.1177/1077546309103419
13. Marin M., Öchsner A. The effect of a dipolar structure on the Hölder stability in Green–Naghdi thermoelasticity. *Contin. Mech. Thermodyn*. 2017. No. 29. P.1365–1374. URL: https://doi.org/10.1007/s00161-017-0585-7
14. Sharma J.N. Thermoelastic Damping and Frequency Shift in Micro/Nanoscale Anisotropic Beams. *J. Therm. Stress.* 2011. V.34. 650–666. URL: https://doi.org/10.1080/01495739.2010.550824
15. Marin M., Vlase S., Ellahi R.,. Bhatti M.M. On the Partition of Energies for the Backward in Time. *Problem of Thermoelastic Materials with a Dipolar Structure, Symmetry.* 2019. No.11. P. 863. URL: https://doi.org/10.3390/sym11070863
16. Zhang J., Fu Y. Pull-in analysis of electrically actuated viscoelastic microbeams based on a modified couple stress theory. *Meccanica*. 2012. V.47. P. 1649–1658. URL: https://doi.org/10.1007/s11012-012-9545-2.
17. Abbas I.A., Marin M. Analytical solution of thermoelastic interaction in a half-space by pulsed laser heating. *Phys. E Low-Dimensional Syst. Nanostructures*. 2017, V.87. P. 254–260. URL: https://doi.org/10.1016/j.physe.2016.10.048
18. Sharma J.N., Kaur R., Transverse Vibrations in Thermoelastic-Diffusive Thin MicroBeam Resonators. *J. Therm. Stress*. 2014. V.37. P. 1265–1285. URL: https://doi.org/10.1080/01495739.2014.936252.
19. Zenkour A.M., Abouelregal A.E. Thermoelastic Vibration of an Axially Moving Microbeam Subjected to Sinusoidal Pulse Heating. *Int. J. Struct. Stab. Dyn*. 2015. V.15. P. 1450081. URL: https://doi.org/10.1142/S0219455414500813.
20. Kumar R., Devi S. Interactions of Thermoelastic Beam in Modified Couple Stress Theory. *Int. J. Appl. Math.* 2017. No.12. P. 910–923.
21. Pradhan S.C., Phadikar J.K. Nonlocal elasticity theory for vibration of nanoplates, *J. Sound Vib.* 2009. V. 325. P. 206–223. URL: https://doi.org/10.1016/j.jsv.2009.03.007
22. Bhatti M.M., Ellahi R., Zeeshan A., Marin M., Ijaz N. Numerical study of heat transfer and Hall current impact on peristaltic propulsion of particle-fluid suspension with compliant wall properties. *Mod. Phys. Lett.. B.* 2019. V.33 P. 1950439. URL: https://doi.org/10.1142/S0217984919504396.
23. Pradhan S.C., Kumar A. Vibration analysis of orthotropic graphene sheets using nonlocal elasticity theory and differential quadrature method. *Compos. Struct.* 2011. V.93. P. 774–779. URL: https://doi.org/10.1016/j.compstruct.2010.08.004.
24. Pradhan S.C., Kumar A. Vibration analysis of orthotropic graphene sheets embedded in Pasternak elastic medium using nonlocal elasticity theory and differential quadrature method. *Comput. Mater. Sci.* 2010. V. 50. P.239–245. URL: https://doi.org/10.1016/j.commatsci.2010.08.009.
25. Marin M. Lagrange identity method for microstretch thermoelastic materials. J. *Math. Anal. Appl.* 2010. V.363.. P. 275–286. URL: https://doi.org/10.1016/j.jmaa.2009.08.045.

26. Marin M. The Lagrange identity method in thermoelasticity of bodies with microstructure. *Int. J. Eng. Sci.* 1994, V.32. P.1229–1240. URL: https://doi.org/10.1016/0020- 7225(94)90034-5.
27. Lata P., Kaur I. Thermomechanical interactions in transversely isotropic magnetothermoelastic medium with fractional order generalized heat transfer and hall current, *Arab J. Basic Appl. Sci*. 2020. V. 27. P. 13–26. URL: https://doi.org/10.1080/25765299.2019.1703494.
28. Lata P., Kaur I. Effect of time harmonic sources on transversely isotropic thermoelastic thin circular plate. *Geomech. Eng*. 2019. No.19. P. 29–36. URL: https://doi.org/10.12989/gae.2019.19.1.029.
29. Kaur I., Lata P. Transversely isotropic thermoelastic thin circular plate with constant and periodically varying load and heat source. *Int. J. Mech. Mater. Eng*. 2019. URL: https://doi.org/10.1186/s40712-019-0107-4.
30. Lata P., Kaur I. Study Transversely Isotropic Thick Circ. Plate Due to Ring Load with Two Temp. Green Nagdhi Theory Type-I, II III. *Proc. Int. Conf. Sustain. Comput. Sci*. 2019, P. 1753–1767. URL: https://doi.org/http://dx.doi.org/10.2139/ssrn.3356884
31. Lata P., Kaur I. Transversely isotropic thick plate with two temperature & GN type-III in frequency domain, *Coupled Syst. Mech.* 2019. No. 8. P. 55–70. URL: https://doi.org/10.12989/csm.2019.8.1.055.
32. Mahakalkarr A., Varghese V., Dhakate T. Thermoelastic Bending Vibrations of a Simply Supported Rectangular Plate with Internal Heat Generation// Evol. Qual. Paradig. Innov. Sustain. Dev. Manag. Inf. Technol., Indore, India, 2019. P. 183–193.
33. Gaikwad M.N., Deshmukh K.C. Thermal deflection of an inverse thermoelastic problem in a thin isotropic circular plate. *Appl. Math. Model*. 2005. URL: https://doi.org/10.1016/j.apm.2004.10.012.
34. Shen Z.B., Tang H.L., Li D.K., Tang G.J. Vibration of single-layered graphene sheet-based nanomechanical sensor via nonlocal Kirchhoff plate theory. *Comput. Mater. Sci.* 2012. V.61. P. 200–205. URL: https://doi.org/10.1016/j.commatsci.2012.04.003.
35. Nguyen N.T. Hui D. Lee J. Nguyen-Xuan H. An efficient computational approach for size-dependent analysis of functionally graded nanoplates. *Comput. Methods Appl. Mech. Eng*. 2015. V.297. P. 191–218. URL: https://doi.org/10.1016/j.cma.2015.07.021
36. Nguyen T.N., Ngo T.D., Nguyen-Xuan H. A novel three-variable shear deformation plate formulation: Theory and Isogeometric implementation. *Comput. Methods Appl. Mech. Eng.* 2017. V.326. P.376–401. URL: https://doi.org/10.1016/j.cma.2017.07.024.
37. T.N. Nguyen, C.H. Thai, H. Nguyen-Xuan, On the general framework of high order shear deformation theories for laminated composite plate structures: A novel unified approach, Int. J. Mech. Sci. 110 (2016) 242–255. https://doi.org/10.1016/j.ijmecsci.2016.01.012.
38. Ventsel E., Krauthammer T., Carrera E.. Thin Plates and Shells: Theory, Analysis, and Applications. *Appl. Mech. Rev*. 2002. V.55. P. B72–B73. URL: https://doi.org/10.1115/1.1483356
39. Синюк О.М. Математична модель анізотропних властивостей полімерних матеріалів. *Herald of Khmelnytskyi National University*. 2015. No. 1. P.221.
40. Kryvyi O. F., Morozov Yu. O. Solution of the problem of heat conduction for the transversely isotropic piecewise homogeneous space with two circular inclusions. *J.Math. Sci.*. 2019. V. 243. No.1. P. 162–182. URL: https://doi.org/10.1007/s10958-019-04533-1.
41. Тонконогий В.М., Голофєєва М.О. Морозов Ю.О. Моделювання внутрішніх процесів в неметалевих гетерогенних матеріалах при акустичному інфрачервоному термометричному методі контролю // Важке машинобудування. Проблеми та перспективи розвитку. *Матеріали XXII Міжнародної науково-технічної конференції*. Краматорськ-Тернопіль: ДДМА, 2024. С. 193-195.

42. Kryvyi O.F., Yu.O. Morozov. Fundamental solutions for a piecewise-homogeneous transversely isotropic elastic space. J. Math. Sci. 2019. V.270. No. 1. P. 143–156. URL: https://doi.org/10.1007/s10958-023-06337-w
43. Kryvyi O., Morozov Yu. The influence of mixed conditions on the stress concentration in the neighborhood of interfacial inclusions in an inhomogeneous transversely isotropic space. *Int. Conf. on Theoretical, Applied and Experimental Mechanics (Structural Integrity, 16).* 2020. P. 204–209. URL: https://doi.org/10.1007/978-3-030-47883-4_38

SIMULATION OF INTERNAL PROCESSES IN NON-METALLIC HETEROGENEOUS MATERIALS USING THE ACOUSTIC INFRARED THERMOMETRICAL CONTROL METHOD

V.M.Tonkonogyi[1], M.O.Golofeyeva[2], Yu.O.Morozov[3], R.V.Gorbatiuk[4]

National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
Emails: vmt47@ukr.net[1]; mgolofeyeva@gmail.com[2];
morozov@op.edu.ua[3]; ruslan.gorbatiuk@stud.op.edu.ua[4]

One of the trends in the development of industry is the replacement of traditional structural materials with non-metallic heterogeneous structures. This makes it possible to obtain materials with planned properties. At the same time, it is necessary to study the mechanisms of organizing the structure of such materials at each scale level. Non-destructive control methods play a significant role here. Methods of active thermal control are promising for products made of heterogeneous materials. Of interest are the processes in materials of this class when they are controlled by the acoustic infrared thermometric method. It can be noted that the specifics of the application of the nondestructive testing method, as well as the physical phenomena that occur in non-metallic heterogeneous materials during the vibration effect on them, have not been fully studied for its practical application. The heterogeneity of the structure, on the one hand, makes it possible to create materials with a wide range of properties, and on the other hand, it makes it difficult to describe the processes that cause the manifestation of these properties. The article is devoted to the mathematical modeling of these processes. For a comprehensive analysis of the strength characteristics, it is necessary to obtain the relationship of thermoelastic fields, that is, it is necessary to simultaneously determine the temperature and deformation fields. A transversally isotropic space is considered, which can be represented as a prism, which consists of a grid of three mutually perpendicular rods located along the coordinate axes x, y, and z, and the mutually perpendicular grid in two directions has the same dimensions. Inside this space, structural defects of any nature are located on an arbitrary piecewise continuous surface.

DOI 10.15276/imms.v14.no3.226
УДК 004.94

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 226-237

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ КРИТИЧНИХ УМОВ ВПЛИВУ ЗОВНІШНЬОГО КОНТЕНТУ НА КОРИСТУВАЧА

Г. В. Шаповалов, О. Павленко

Національний університет «Одеська політехніка»
1, Шевченка пр., Одеса, 65044
Email: shapovalov@op.edu.ua

У роботі виконано математичне моделювання впливу небезпечного контенту на користувача глобальної мережі. Виконано прогнозування критичних станів, за яких можливі кількісні та якісні переходи впливу інформації на користувача. На основі проведених досліджень отримано експериментальні дані щодо залежності конверсії – кількості успішних дій, що є метою впливу інформації на користувача, наприклад реклами покупок чи кандидатур на виборах та інших вхідних даних від впливу на користувача мережі. Шляхом кореляційного аналізу визначено найбільш впливові дані, пов'язані з результатами конверсій, а методами апроксимації – функціональні залежності результатів впливу від вхідних даних, таких як ціна/конверсія, кліки/конверсія та покази/конверсія. Порівняння результатів моделювання свідчить про існування простору співіснування фаз другого порядку навколо досліджуваних просторів, що свідчить про можливість небезпечного впливу рекламної інформації на користувача. Результати моделювання свідчать про те, що за отриманими даними виявлено простір, за виконання умов існування якого користувач мережі може змінити свою позицію щодо рекламованих товарів за рахунок впливу рекламних кампаній. Небезпека полягає в тому, що користувач мережі може придбати речі, які, загалом, йому не потрібні. Такий зовнішній вплив є небезпечним і відповідає маніпулюванню одними учасниками глобальної мережі іншими. Така ситуація відповідає стану користувача, коли у фазі стійкого ставлення до контенту, що надходить до нього, виникає нове якісне ставлення, тобто кількісний вплив на користувача призводить до якісних змін у сприйнятті інформація, яка надходить до нього із зовнішнім змістом. Таке втручання є несанкціонованим, тому містить небезпеку зовнішнього контролю користувача та має бути ретельно досліджено.

Ключові слова: математичне моделювання, вплив небезпечного контенту, конверсія, небезпека зовнішнього контролю користувача.

Вступ. В сучасному інформаційному просторі користувач становиться повністю незахищеним від зовнішнього впливу інформації, яка надходить до нього у вигляді реклами товарів, залучення до різноманітних подій, ворожої інформації та іншого різноманітного впливу. На перший погляд, людина досвідчена завжди може відокремитися від такого впливу, але, як показує практика вплив контенту може легко перевищити «захисний інформаційний бар'єр» різних за віком, освітою та соціальною приналежністю і привести до самих різних наслідків, пов'язаних з метою контенту. Дослідження, пов'язані з впливом реклами на користувача фінансуються рекламними компаніями, але негативний вплив на користувача при цьому не досліджується.

Метою дослідження є аналіз впливу зовнішнього контенту на користувача та прогнозування критичних явищ, за якими виконуються умови зміни стану користувача під впливом зовнішнього контенту.

Аналіз впливового зовнішнього контенту та формування матриці даних. В роботі виконано математичне моделювання впливу небезпечного контенту на користувача глобальної мережі. Метою роботи було прогнозування критичних станів, за якими можливі кількісно-якісні переходи впливу інформації на користувача. На основі проведених досліджень були отримані експериментальні данні стосовно залежності конверсії - кількості успішних дій, які є метою впливу інформації на користувача

(наприклад, реклама для покупки або кандидатури на виборах та інших вхідних даних щодо впливу) від:

1) зображення - зображення товару або іншого об’єкту впливу, яке використовується для залучення клієнтів на сайті;
2) назви - найменування товару або послуги;
3) ідентифікатора рекламодавця (або, наприклад, продавця) - унікальний номер, який присвоюється продавцю для ідентифікації;
4) ідентифікатора позиції - унікальний номер, що присвоюється, наприклад, кожному товару або позиції;
5) статусу - поточний статус позиції (наприклад, активно, неактивно або знято з продажу);
6) ціни - ціна товару або послуги;
7) кликів - кількість кліків, здійснених користувачами на товар (рекламу).
8) показів - кількість показів рекламного оголошення;
9) CTR (Click-Through Rate) - показник «кликабельності», розраховується як відношення кількості кликів до кількості показів (у відсотках);
10) коду валюти - валюта, у якій вказана ціна товару (якщо йдеться про вплив реклами на покупця);
11) середня ціна за клік - середня вартість кліку по рекламному оголошенню;
12) витрати - сума, витрачена на рекламу;
13) вартість/конверсія - вартість однієї конверсії, що розраховується як відношення витрат на рекламу до кількості конверсій;
14) відсоток отриманих показів на верхній позиції в пошуковій системі (ВПП) на верхній позиції в пошуковій мережі;
15) коефіцієнт усіх конверсій;
16) загальна цінність усіх конверсій (виражається в грошовому еквіваленті);
17) коефіцієнт конверсії - відношення кількості конверсій до кількості кликів (у відсотках).
18) цінність конверсії у відношенні до її вартості;
19) загальна кількість усіх конверсій;
20) середня вартість замовлення - середня сума, витрачена клієнтами на одне замовлення (якщо йдеться про вплив реклами на покупця);
21) кількість проданих одиниць - загальна кількість товарів, проданих за звітний період.
22) канал - джерело, через яке прийшов користувач;
23) відсоток отриманих показів у пошуковій мережі - відсоток усіх можливих показів, які були отримані в пошуковій мережі;
24) відсоток втрачених показів у пошуковій мережі (рейтинг) - відсоток показів, втрачених через недостатньо високий рейтинг оголошення;
25) відсоток отриманих кликів - відсоток усіх можливих кликів, які були отримані.

За кореляційним аналізом було визначено найбільш впливові данні, що пов’язані з результатами конверсії та за методами апроксимації отримано функціональні залежності результатів впливу від вхідних даних, таких, як ціна/конверсія (x_1), кліки/конверсія (x_2) та покази/конверсія (x_3).

Дослідження залежності конверсії від ціни. За отриманими даними (Табл. 1.) було побудовано залежність конверсії від ціни товару. Результати наведено на Рис.1.

Таблиця 1.

Витяг з результатів дослідження залежності конверсії від ціни товару, кількості кликів по рекламі та кількості показів в мережі.

Ціни (грн.)	Кліки (кількість)	Покази (кількість)	Конверсії (кількість)
370,00	5 628	215 851	588,66
249,00	1 914	94 394	248,73
299,00	1 717	98 793	235,36
87,00	1 588	118 366	106,53
270,00	1 246	62 340	49,5
550,00	1 202	74 674	11
429,00	1 176	109 555	167,38
360,00	1 104	69 478	40
900,00	818	47 158	22
159,00	726	47 859	18
49,00	678	79 013	54,84
270,00	597	41 603	10,5
65,00	363	59 021	25,45
1 500,00	266	12 808	1
590,00	260	30 026	4
390,00	159	7 370	3,08
75,00	159	34 923	14,95
119,00	122	7 689	3
142,00	120	11 798	8
950,00	91	24 767	0
500,00	73	21 747	3
149,00	43	6 333	1
350,00	11	2 375	1

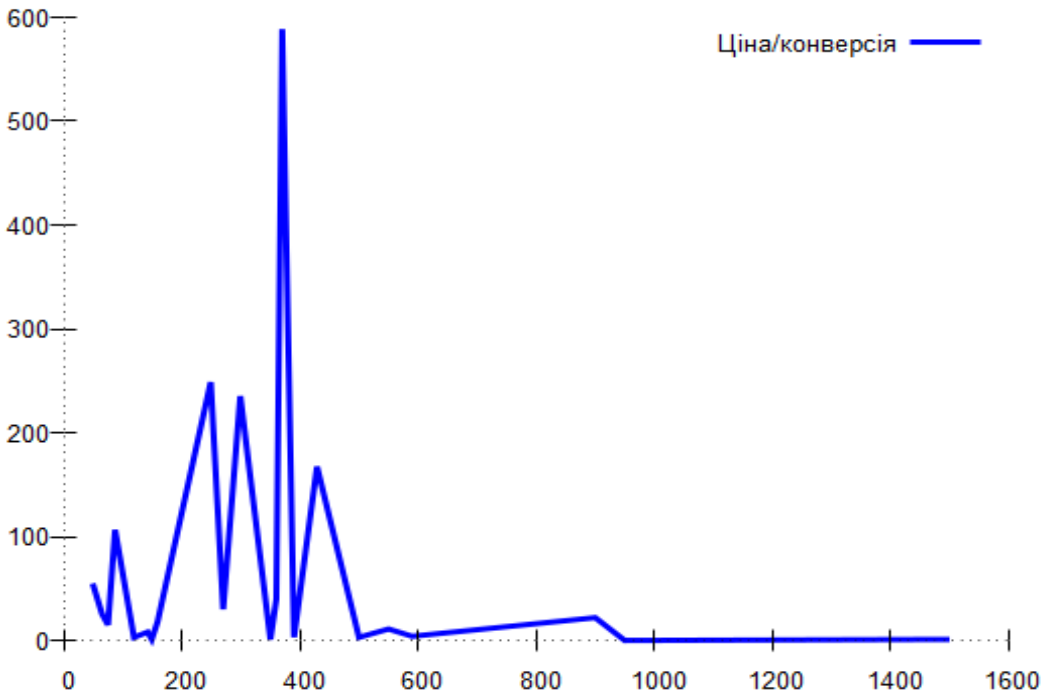


Рис 1. Результати дослідження залежності ціна/конверсія

Результати моделювання залежності конверсії від ціни наведено на Рис.2:

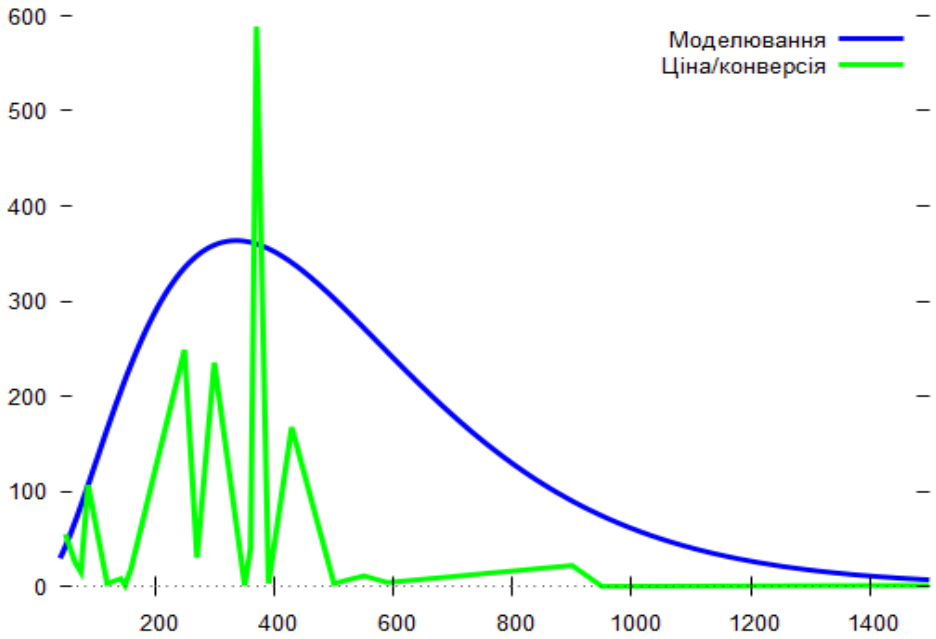


Рис. 2. Результати моделювання залежності ціна/конверсія

Результати моделювання вказують на можливість застосування Гамма-розподіл Ерланга [1-4] для опису залежності ціна/конверсія. Обчислення параметрів розподілу надали змогу отримати залежність:

$$f(x_1) = \frac{{x_1}^{2,015} e^{-0.006x_1}}{45,125} \quad (1)$$

Дослідження залежності конверсії від кліків по рекламним повідомленням. Результати дослідження залежності конверсії від кількості кліків по рекламним об'явленням наведено на рис.3.

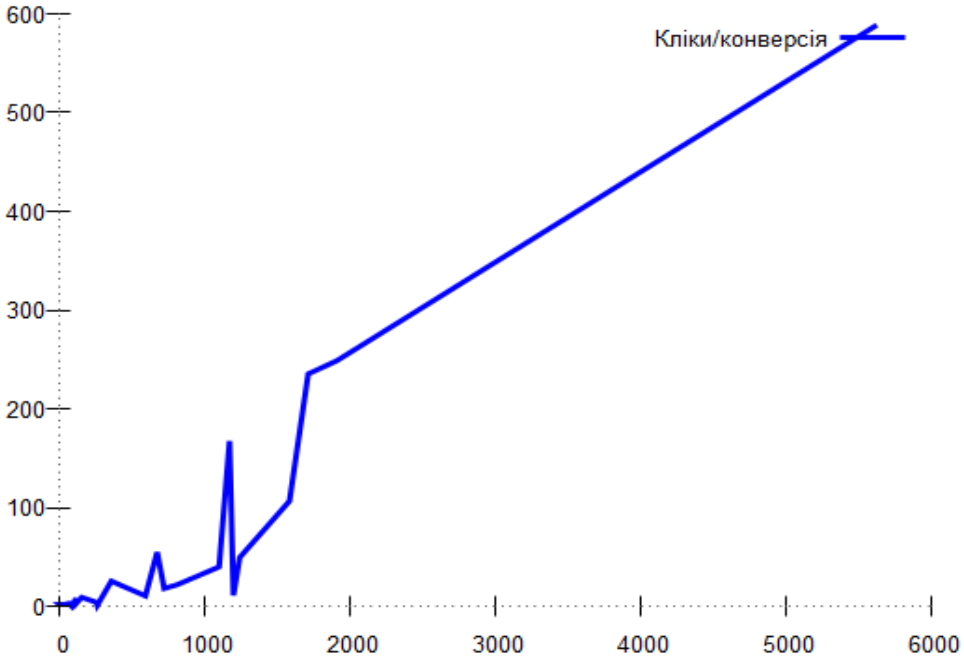


Рис 3. Результати дослідження залежності кількість кликів / конверсія

Результати моделювання залежності конверсії від кількості кліків наведено на Рис. 4:

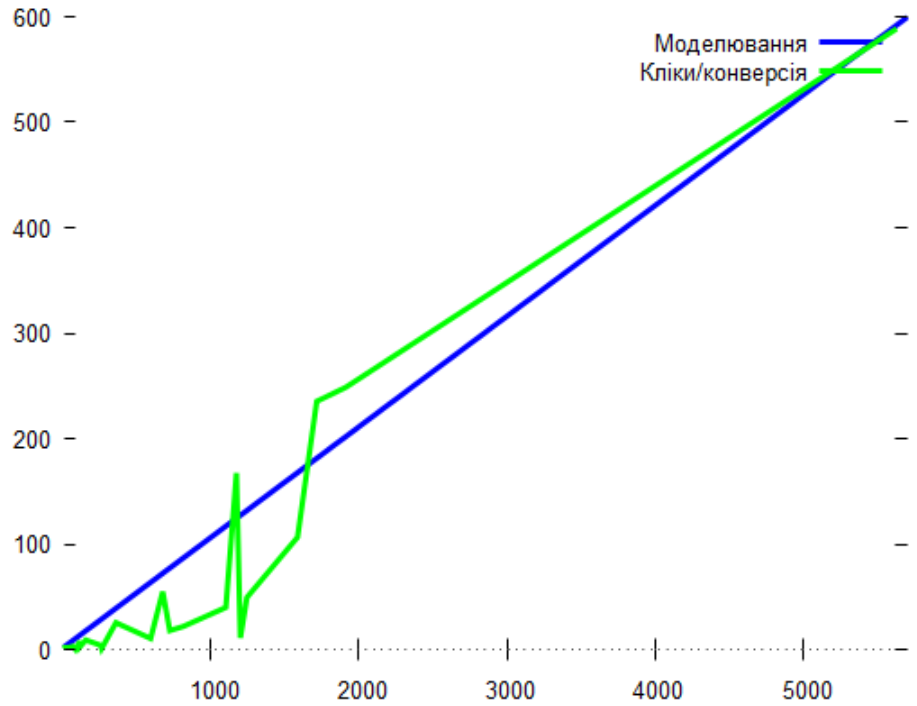


Рис. 4. Результати моделювання залежності кількість кліків/конверсія

Результати моделювання вказують на можливість лінійну залежність кількість кліків / конверсія. Обчислення параметрів лінійної залежності надали змогу отримати функцію:

$$f(x_2) = 0.105x_2 + 1.05 \quad (2)$$

Дослідження залежності конверсій від показів. Для моделювання впливу кількості показів контенту від їх впливу на користувача за даними кількість показів / конверсія було отримано спочатку було отримано графік відповідної залежності за даним Таблиці 1. Результати наведено на Рис. 5

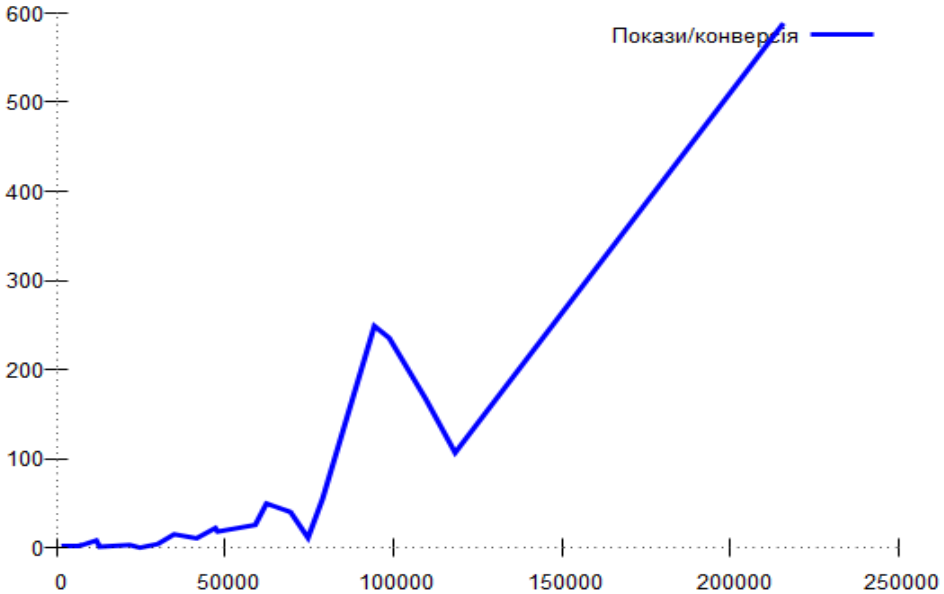


Рис 5. Результати дослідження залежності кількість показів / конверсія

За отриманими даними було проведено моделювання досліджуваної залежності. Результати моделювання залежності кількість показів / конверсія наведено на Рис. 6:

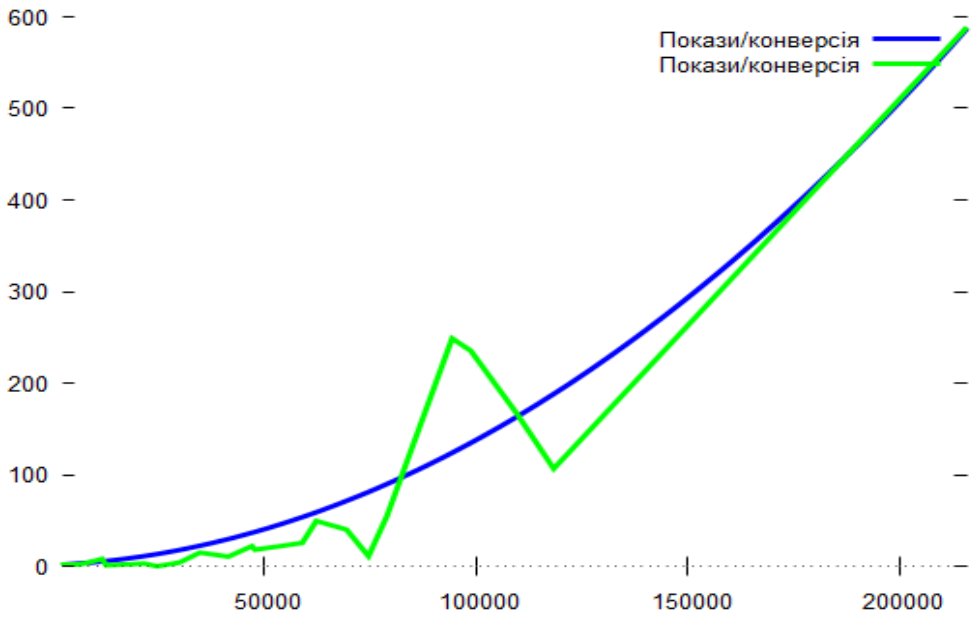


Рис 6. Результати моделювання залежності кількість показів / конверсія

Результати обчислення параметрів отриманої залежності покази / конверсія та отримана функціональна залежність наведена у (3):

$$f(x_3) = 1{,}165 \cdot 10^{-8} {x_3}^2 + 0.0002x_3 + 1.245 \tag{3}$$

Моделювання впливу зовнішнього контенту на користувача. За результатами (1) – (3) було отримано функцію впливу у вигляді:

$$f(x_1, x_2, x_3) = 1.165 \cdot 10^{-8} x_3^2 + 2.0 \cdot 10^{-4} x_3 + 0.105x_2 + 0.022 x_1^{2,02} e^{-0.006x_1} + 2.295 \quad , \tag{4}$$

де змінними x_1 моделюється залежність ціна/конверсія, x_2 – кліки/конверсія та x_3 – покази/конверсія.

Для з'ясування умов, за якими користувач може якісно змінити своє ставлення до зовнішнього інформаційного навантаження, в роботі проведено аналіз умов, за якими одночасно дорівнюють нулю значення похідних функції впливу з першої по третю включно, а значення четвертої похідної функції впливу приймають додатні значення [5-9]:

$$\frac{df}{dX} = \frac{d^2 f}{dX^2} = \frac{d^3 f}{dX^3} = 0, \quad \frac{d^4 f}{dX^4} > 0 \tag{5}$$

$$X = (x_1, x_2, x_3)$$

Такий стан відповідає умовам, коли у досліджуваному фазовому просторі виконуються умови одночасного існування двох рівнозначних фаз, тобто виникає простір співіснування фаз другого порядку. Для знаходження такого простору було застосовано диференціально-топологічний підхід [5-7], за яким послідовно було знайдено умови існування стабільної фази та біфуркаційного простору на фазових діаграмах в координатах залежності конверсії від ціни та показів за фіксованою

кількістю кліків. Було застосовано алгоритм, за яким, за яким знаходились в аналітичній формі вирази похідних цільової функції впливу послідовно від першої до четвертої похідної включно. Для кожної знайденої похідної в досліджуваній області $x_1 \in [40.0; 150.0]$ (грн.), $x_2 \in [11; 1628]$ (кількість кліків), $x_3 \in [2375.0; 215851.0]$ (кількість показів) було обчислено та проведено аналіз топології та знайдені контури, вздовж яких досліджувані похідні приймають нульові значення. Для знаходження аналітичних виразів похідних та обчислення їх нульових контурів було застосовано систему комп'ютерної алгебри MAXIMA [10]

Для аналізу умов існування стабільної фази було знайдено простори, в яких одночасно виконуються умови нульових значень компонентів градієнта цільової функції, тобто функції впливу, та додатніх значень другої похідної:

$$\frac{df}{dX} = 0, \quad det\frac{d^2f}{dX^2} > 0 \tag{6}$$

$$X = (x_1, x_2, x_3)$$

$$\frac{df}{dX} = \left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, \frac{\partial f}{\partial x_3}\right) = 0, \qquad det\frac{d^2f}{dX^2} = det\begin{pmatrix} \frac{\partial^2 f}{\partial {x_1}^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \frac{\partial^2 f}{\partial x_1 \partial x_3} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial {x_2}^2} & \frac{\partial^2 f}{\partial x_2 \partial x_3} \\ \frac{\partial^2 f}{\partial x_3 \partial x_1} & \frac{\partial^2 f}{\partial x_3 \partial x_2} & \frac{\partial^2 f}{\partial {x_3}^2} \end{pmatrix}$$

Для отримання перерізів нульових контурів похідних фіксувалися значення змінної x_2 (кількість кліків, $x_2 = 1000$) і отримувався переріз у площині (x_1, x_3), тобто (ціна, покази). Для цього спочатку було отримано поверхню досліджуваної функціональної залежності, яку наведено на рис. 7. Аналіз отриманої поверхні показав наявність нульових контурів для за $x_1 \in (300-400)$ та $x_1 > 800$. З метою уточнення сигнатури простору у знайдених областях було обчислено положення нульових контурів та топологію простору навколо них. Результат моделювання поверхні першої похідної функції конверсії наведено на Рис. 7.

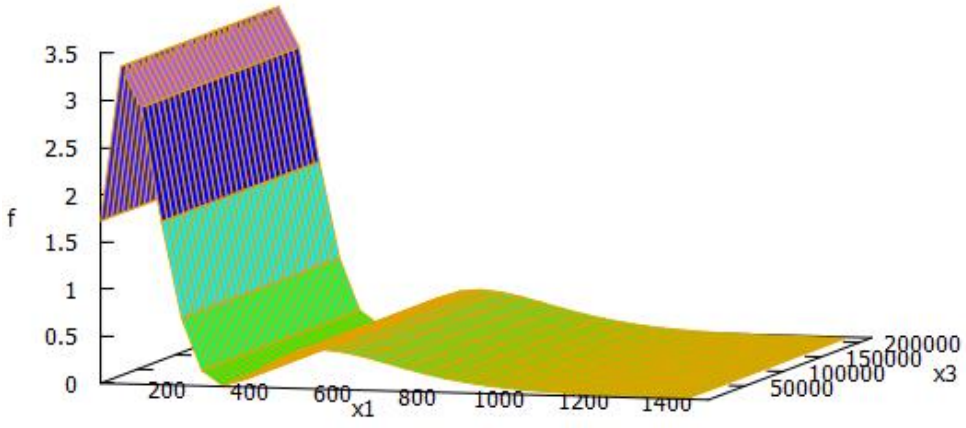


Рис. 7. Результат моделювання поверхні першої похідної функції конверсії

Результати обчислення нульових контурів першої похідної функції конверсії наведено на Рис. 8:

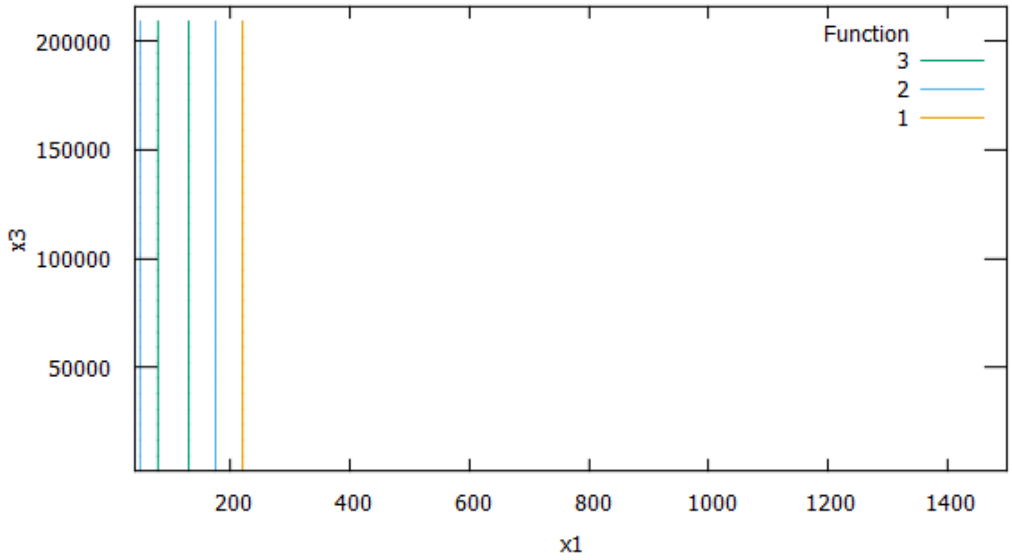


Рис. 8. Результати обчислення нульових контурів першої похідної функції конверсії

Результати моделювання (Рис.7 – 8) та безпосереднє обчислення значень похідних у досліджуваних областях вказують на наявність двох нульових контурів першої похідної функції конверсії.

Результати моделювання поверхні другої похідної функції конверсії наведено на Рис. 9. Аналіз топології отриманого простору свідчить про існуванні нульових контурів досліджуваної похідної.

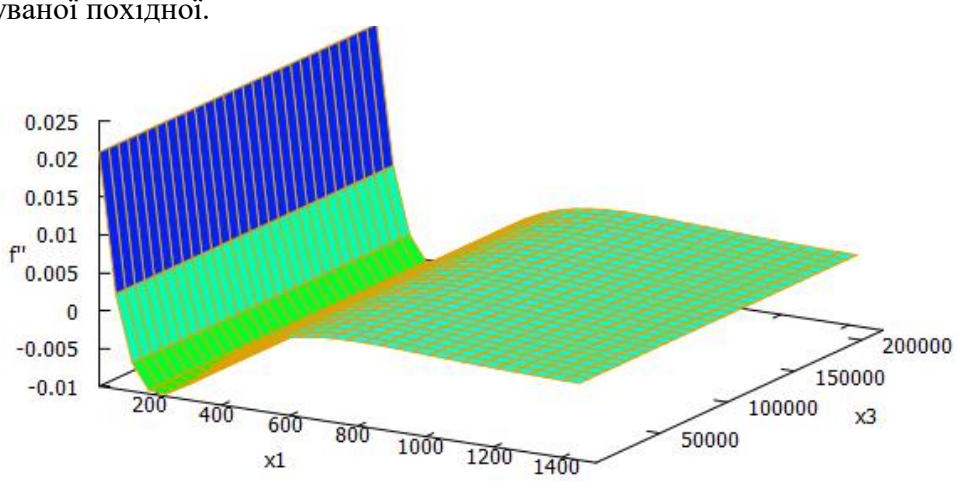


Рис. 9. Результат моделювання поверхні другої похідної функції конверсії

З метою визначення виконання умови (6) було обчислено положення нульових контурів другої похідної функції конверсії, та отримані результати нанесено на переріз існування досліджуваної похідної. Результати наведено на Рис. 10.

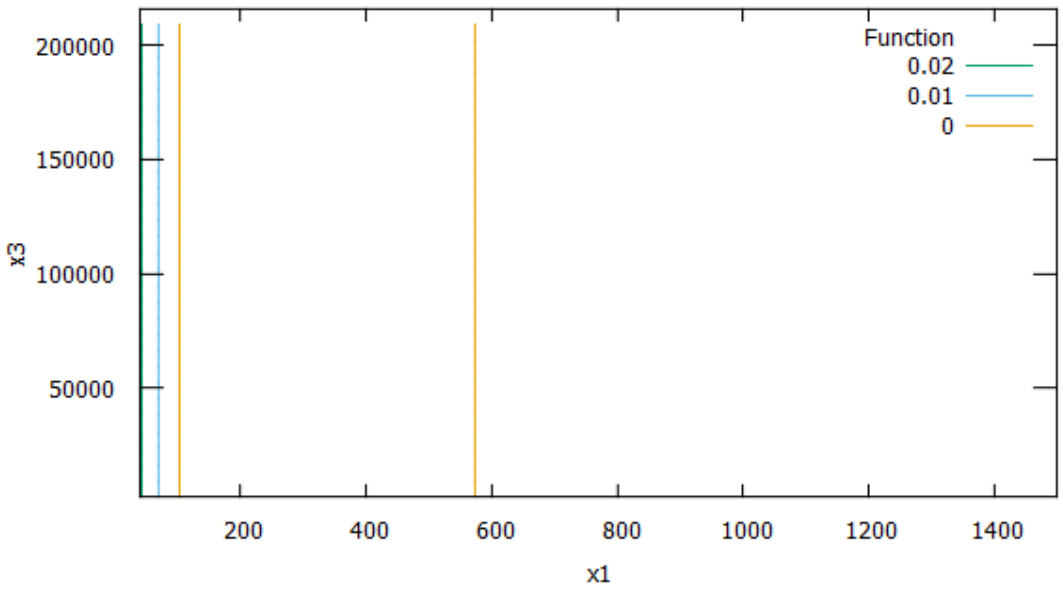


Рис.10. Результати обчислення нульових контурів другої похідної функції конверсії

За результатами моделювання поверхні другої похідної функції конверсії та обчислень положень її нульових контурів видно, що умова (6) виконується тільки для одного контуру, що знайдено за аналізом першої похідної функції конверсії (поблизу $x_1 = 1000$)

Для аналіз виконання умов існування біфуркаційного простору було перевірено умови [5 – 9]:

$$\frac{df}{dX} = \frac{d^2f}{dX^2} = 0, \quad \frac{d^3f}{dX^3} > 0 \tag{7}$$

$$X = (x_1, x_2, x_3)$$

На наступному кроці, для перевірки виконання умов (5) було отримано аналітичний вигляд третьої похідної функції впливи та проведено її топологічний аналіз. Результати побудови поверхні третьої похідної наведено на Рис. 11:

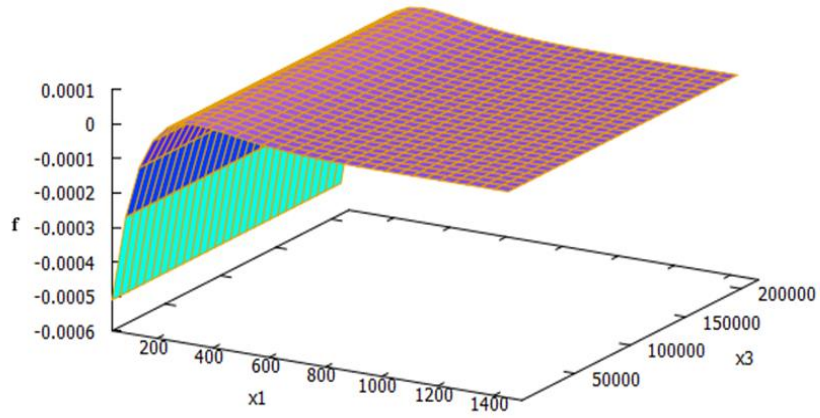


Рис. 11. Результат моделювання поверхні третьої похідної функції конверсії

Результати обчислення нульових контурів третьої похідної функції конверсії наведено на Рис. 12:

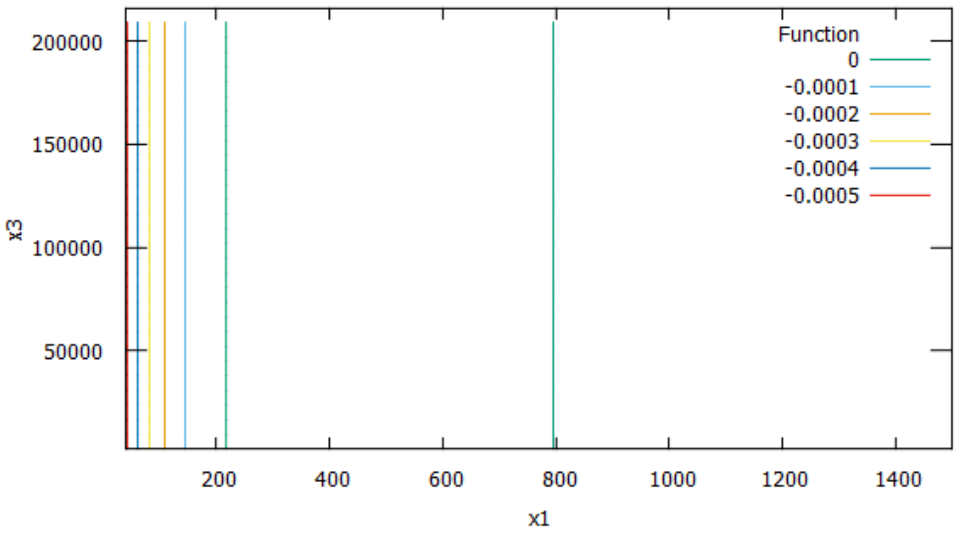


Рис. 12. Результати обчислення нульових контурів третьої похідної функції конверсії

Перевірка виконання умов існування простору співіснування фаз другого порядку було проведено за аналогічним алгоритмом. Умови існування простору співіснування фаз другого порядку, тобто за якими можливо якісне змінення стану досліджуваної системи наведено у (5). Виконання таких умов відповідає зміні фази досліджуваної системи, тобто користувач стрибком змінює своє відношення до продукту, який рекламується в мережі і від стану, коли він не звертає уваги на рекламу продукції переходить у стан не тільки зацікавленості, а й готовності придбати продукцію, що рекламується. Результат моделювання поверхні четвертої похідної функції конверсії наведено на Рис. 13:

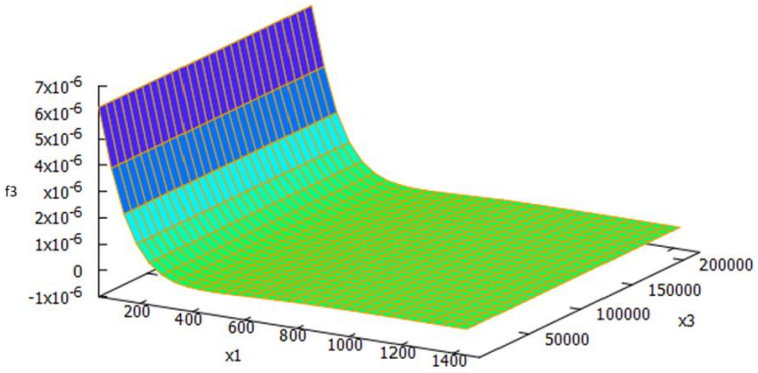


Рис. 13. Результат моделювання поверхні четвертої похідної функції конверсії

Результати обчислення нульових контурів четвертої похідної функції конверсії наведено на Рис. 14:

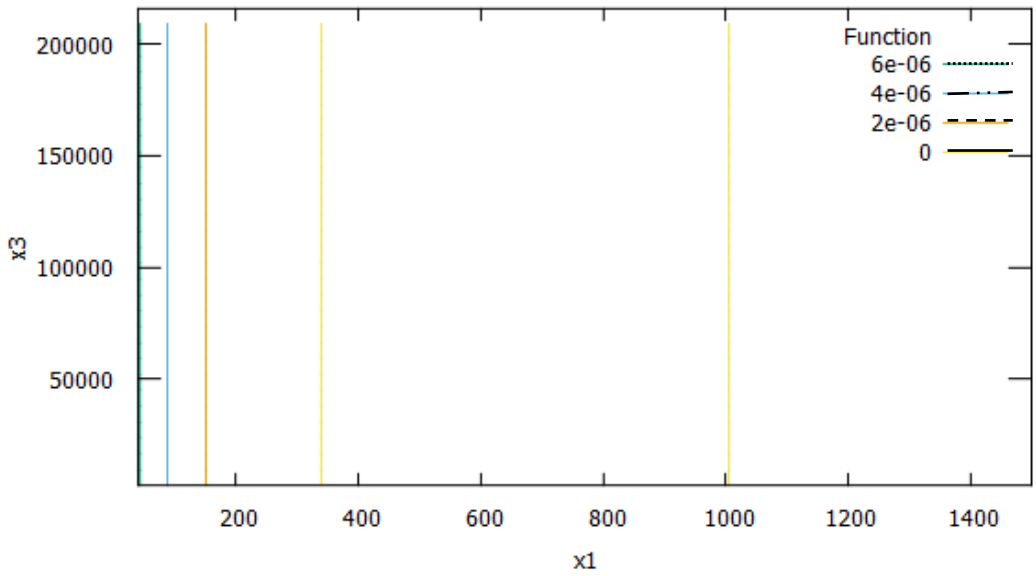


Рис. 14. Результати обчислення нульових контурів четвертої похідної функції конверсії

Висновки. Співставлення результатів моделювання, наданих на рис. 7 – 11, свідчать про існування простору співіснування фаз другого порядку навколо області змінної $x_1 \in (800; 1000)$ за фіксованим значенням змінної $x_2 = 1000$ вздовж $x_3 \in (50000; 20000)$. Це свідчить про можливість небезпечного впливу рекламної інформації на користувача. Результати моделювання вказують, що за отриманими даними знайдено простір, за виконанням умов існування якого користувач мережі може змінити своє становище до продукції, яка рекламується завдяки впливу рекламних акцій. Небезпека полягає у тому, що користувач мережі може придбати речі, які, взагалі, йому не потрібні. Такий зовнішній вплив є небезпечним і відповідає маніпуляції одних учасників глобальної мережі іншими. Знайдені за умовою (8) простори співіснування фаз відповідають станам, в яких досліджувана система перебуває одночасно у двох фазах, тобто один стабільний стан співіснує з іншим стабільним станом. Таке становище відповідає стану користувача, коли у фазі стійкого відношення до контенту, який до нього надходить, зароджується нове якісне відношення, тобто кількісний вплив на користувача призводить до якісних змін у сприйнятті інформації, яка надходить до нього із зовнішнім контентом. Таке втручання є несанкціонованим, тому містить небезпеку зовнішнього керування користувачем і повинно бути ретельно досліджено.

Список літератури

1. Choi K.P. On the medians of gamma distributions and an equation of Ramanujan. *Proceedings of the American Mathematical Society* . 121 (1): 245—251. doi:10.1090/S0002-9939-1994-1195477-8. JSTOR 2160389.
2. Adell, J. A., Jodrá P. On a Ramanujan equation connected with the median of the gamma distribution. Transactions of the American Mathematical Society. 2008. V. 360 (7). P. 3631—3644. doi:10.1090/S0002-9947-07-04411-X
3. Jodrá P. Computing the Asymptotic Expansion of the Median of the Erlang Distribution. *Mathematical Modelling and Analysis*. 2012. V.17 (2). P.281—292. doi:10.3846/13926292.2012.664571
4. Chatfield C. Goodhardt, G.J.. A Consumer Purchasing Model with Erlang Inter-Purchase Times. *Journal of the American Statistical Association* . 68: 828—835. JSTOR 2284508.
5. Cahn J. On spinodal decomposition. J.*Cahn.Acta Met*. 1961. V. 9. P. 795 — 801.
6. Okada K., Suzuki I. Classical calculations on the phase transition I. Phase diagram in four-dimensional space for the system with one order parameter. *J. Phys. Soc. Jap.* 1982. V. 51. No 10. P. 3250 — 3257.

7. Shapovalov H., Kazakov A., Berber I. Mathematical modeling of critical phenomena in biomedical systems. *International Science Journal of Engineering & Agriculture*. 2023. V. 2. No. 4, P. 1-8. doi: 10.46299/j.isjea.20230204.01.
8. Shapovalov H., Kazakov A., Oleynyk V. Mathematical modeling of critical phenomena according to the plebiansky-demyansky metric. *The level of development of science and technology in the XXI century 'Monographic series «European Science».* 2023. P. 123 - 130
9. Shapovalov H., Kazakov A., Ksendziyk G. Computer Simulation of Critical Phenomena in Materials Cyber Systems Elements. *12 International Conference on Applied Innovations in AT (ICAIIT), Koethen, Germany*, 2024. V.12. No 1. P.. 167 - 172
10. Maxima – система комп'ютерної алгебри для роботи з символьними та чисельними виразами. URL:https://uk.vessoft.com/software/windows/download/maxima

MATHEMATICAL MODELING OF CRITICAL IMPACTS OF EXTERNAL CONTENT ON THE NETWORK USER

H.V. Shapovalov, O. Pavlenko

National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
Email: shapovalov@op.edu.ua

The paper has performed mathematical modeling of the impact of dangerous content on the user of the global network. The prediction of critical states, under which quantitative and qualitative transitions of the impact of information on the user are possible, has been performed. Based on the conducted research, experimental data were obtained regarding the dependence of conversion - the number of successful actions that are the purpose of the impact of information on the user, such as advertising for purchases or candidacies in elections and other input data on the impact on the network user. By correlation analysis, the most influential data related to the results of conversions were determined, and by approximation methods, functional dependencies of the impact results on input data, such as price/conversion, clicks/conversion and impressions/conversion, were obtained. Comparison of modeling results indicates the existence of a space of coexistence of second-order phases around the studied spaces, which indicates the possibility of a dangerous impact of advertising information on the user. The modeling results indicate that according to the obtained data, a space has been found, under the fulfillment of the conditions of the existence of which the network user can change his position towards the products advertised due to the influence of advertising campaigns. The danger lies in the fact that the network user can purchase things that, in general, he does not need. Such external influence is dangerous and corresponds to the manipulation of some participants of the global network by others. Such a situation corresponds to the state of the user when, in the phase of a stable attitude towards the content that comes to him, a new qualitative attitude arises, that is, the quantitative impact on the user leads to qualitative changes in the perception of the information that comes to him with external content. Such intervention is unauthorized, therefore it contains the danger of external control of the user and should be carefully investigated.

DOI 10.15276/imms.v14.no3.238
УДК 004.92

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 238-248

РОЗРОБКА ТА АНАЛІЗ АЛГОРИТМІВ ДЛЯ МОДЕЛЮВАННЯ, СИМУЛЯЦІЇ ТА ОПТИМІЗАЦІЇ ФІЗИЧНИХ ВЛАСТИВОСТЕЙ ТКАНИН У 3D-МОДЕЛЮВАННІ

В.Г. Шатохіна[1], Л.В. Бовнегра[2]

Національний університет «Одеська політехніка»
1, Шевченка пр., Одеса, 65044, Україна
Emails: v.g.shatokhina@op.edu.ua[1], dlv5@ukr.net[2]

Дослідження присвячене розробці алгоритмів для 3D-моделювання та симуляції тканини, що має ключове значення для досягнення високої реалістичності текстильних об’єктів. Із розвитком технологій зростають вимоги до якості та точності моделей, особливо для моделювання тканини, яке потребує значних обчислювальних ресурсів. Стаття зосереджена на алгоритмах програми 3ds Max для моделювання тканини та можливості їх інтеграції з CLO 3D з метою підвищення продуктивності та якості фінального зображення. Наукова й практична значущість роботи полягає у застосуванні спеціалізованих алгоритмів, що дають змогу знизити витрати часу на обробку і збільшити точність візуальних ефектів, сприяючи оптимізації процесів у 3D-дизайні й комп'ютерній графіці, де важливим критерієм є реалістичність. Методологія дослідження охоплює теоретичний аналіз алгоритмів моделювання тканини в 3ds Max, їх порівняння з алгоритмами CLO 3D, а також розробку і тестування комбінованого підходу, що об'єднує переваги обох програм. Результати експериментальних симуляцій підтверджують, що створення базових моделей у CLO 3D із подальшою обробкою в 3ds Max забезпечує оптимальний баланс між реалістичністю та продуктивністю, знижуючи обчислювальні витрати та підвищуючи якість моделювання у складних 3D-проєктах. Цінність роботи полягає у розширенні можливостей реалістичного моделювання тканини завдяки поєднанню функцій спеціалізованих програм для текстильного моделювання. Внесок цього дослідження полягає у створенні ефективної та гнучкої методології для моделювання тканини, що перевершує підходи з використанням лише однієї програми. Практичне значення результатів відображається у можливості застосування комбінованого підходу в анімації та кіно, де економія ресурсів є важливим фактором для реалізації масштабних проєктів.

Ключові слова: моделювання тканин, 3ds Max, CLO 3D, симуляція, оптимізація, комп'ютерна графіка, 3D-моделювання.

Вступ. Моделювання тканини є одним із основних завдань у 3D-дизайні. Створення реалістичного одягу, драпіровок, меблевих оббивок чи будь-якого іншого матеріалу з гнучкою структурою значно впливає на враження від 3D-моделей [1]. Відтворення природнього вигляду та поведінки тканини є критичним для різних галузей, як-от кіновиробництво, анімація, геймдизайн, моделювання інтер'єрів та віртуальна мода [2]. Точне моделювання динаміки тканини, її реакції на рухи персонажів або зовнішні сили, такі як вітер і гравітація, дозволяють створювати сцени, які виглядають реалістично та емоційно достовірно.

Актуальними проблемами в симуляції тканин є досягнення реалістичності поведінки матеріалів при збереженні продуктивності та ефективного використання обчислювальних ресурсів. Ці виклики вимагають постійного вдосконалення алгоритмів і пошуку оптимальних підходів для створення деталізованих і ресурсозберігаючих симуляцій.

Сучасні вимоги до якості візуальних ефектів спонукають до надзвичайної деталізації симуляції тканини. Однак модель поведінки тканини вимагає точної передачі таких характеристик, як текстура, вага, товщина, пружність, які змінюються залежно від

типу матеріалу. Використання стандартних алгоритмів у програмах на кшталт 3ds Max може бути обмеженим у реалістичному відтворенні різноманітних тканин, які ставлять вимоги до розробки точніших симуляцій [3].

Симуляція тканини – це складний обчислювальний процес, який вимагає значних ресурсів. Під час рендерингу анімації з тканинами потрібна висока потужність для обробки рухів, взаємодії з іншими об'єктами та фізичних сил, що впливають на тканину. Це може спричинити сильні навантаження на систему, особливо у великих сценах або під час моделювання тонких і легких тканин.

Оптимізація є ключем до забезпечення якості та ефективності симуляції, оскільки велика деталізація може виявитися надлишковою і перевантажити рендеринг. Важливо зберегти баланс між обчислювальними витратами та симуляцією, яка потребує високих показників у таких параметрах, як полігональність об'єктів, налаштування текстури та точність фізичних властивостей тканини.

Точність моделювання тканини значно впливає на якість та реалістичність проєктів. У кінематографі та анімації детально відтворена тканина дозволяє точно передавати рухи, текстури та взаємодію матеріалу з іншими об’єктами, додаючи сценам емоційної глибини та правдоподібності. В індустрії моди реалістичне моделювання дає можливість дизайнерам і замовникам побачити точний вигляд одягу ще до фізичного пошиття, що прискорює процес створення і зменшує витрати на виробництво. У геймдизайні точна симуляція тканини додає деталі і природність в ігровому середовищі, що забезпечує занурення гравця у віртуальний світ. Однак збільшення реалістичної динаміки тканин вимагає значних ресурсів, особливо у великих проєктах або інтерактивних середовищах. Це підкреслює значну важливість оптимізації, оскільки вона дозволяє досягти необхідного рівня реалістичності без перевантаження обладнання, забезпечуючи при цьому стабільну продуктивність і якість.

Аналіз досліджень і публікацій. У наукових роботах, присвячених 3D моделюванню одягу, автори розглядають кілька ключових тем. Зокрема, досліджується використання 3D-технологій у дизайні одягу, де аналізуються техніки та художні засоби для створення костюмів за допомогою інноваційних технологій. Це дозволяє зрозуміти переваги та виклики, пов'язані з впровадженням 3D-технологій у процес дизайну одягу [4]. Інша важлива тема стосується сучасних інформаційних технологій у дизайні одягу. Автори розглядають концепції тривимірного моделювання одягу, удосконалення методів трансформації базових конструкцій та використання програмного забезпечення для 3D моделювання. Це включає аналіз ефективності різних програм та їх вплив на процес проєктування. Також значна увага приділяється цифровим трансформаціям у дизайні одягу. У статтях систематизуються способи використання цифрових 3D-технологій у сучасних дизайнерських колекціях та аналізується їхній вплив на етапах проєктування. Це допомагає зрозуміти, як цифрові інструменти можуть покращити процес створення одягу та підвищити його ефективність.

Проте жоден з авторів не розглядає 3D моделювання одягу саме з точки зору оптимізації та комбінованого підходу, що не допомагає поглибитися у суть проблеми та розглянути цю галузь з технічної точки зору.

Метою даної роботи є аналіз алгоритмів моделювання тканини в 3ds Max і розробка комплексного підходу до моделювання тканини з використанням 3ds Max разом із CLO3D для підвищення реалістичності та продуктивності. Це з'єднання дозволяє оптимізувати процеси моделювання, підвищити реалістичність текстильних матеріалів та зменшити навантаження на обчислювальні ресурси. Такий підхід спрямований на вдосконалення продуктивності моделювання складних тканинних структур, зокрема для індустрії, де видимої якості та обчислювальної ефективності.

Основний розділ. У 3ds Max для симуляції тканин використовують різні методи, серед яких найбільш розширеними є модифікатор Cloth та система MassFX для обчислення фізики.

Модифікатор Cloth — один з основних інструментів для моделювання тканини, який надає можливість налаштування параметрів жорсткості, пружності, ваги та взаємодії тканини з іншими об'єктами в сцені. Цей модифікатор дозволяє створювати основні ефекти згинання та розтягування, відтворюючи основні властивості матеріалу. Проте він має обмеження в деталізації моделювання, особливо при роботі з легкими тканинами чи складними формами, де моделювання виглядає менш реалістичним, ніж у спеціалізованих програмах.

Система MassFX — підходить для створення базових симуляцій взаємодії тканин із фізичним середовищем, зокрема для колізій та реакцій на сили, таких як гравітація. MassFX також забезпечує певні можливості для динаміки тканини, але ця система більш пристосована до моделювання жорстких тіл і має обмеження у відтворенні деталей динаміки м'яких матеріалів, таких як тканина.

Обмеження цих методів у 3ds Max полягають у складності налаштувань дуже реалістичних симуляцій для тонких тканин і драпіровок, які потребують більше параметрів, ніж ті, що доступні в стандартному інтерфейсі. Крім того, через високі навантаження на обчислювальні ресурси, симуляція може бути непридатною для великих сцен чи інтерактивного середовища, що обмежує гнучкість у прикладному інструменті для спеціалізованих проєктів.

Переваги алгоритмів у 3ds Max:

- Наявність гнучкого налаштування, адже модифікатор Cloth надає можливість виявляти широкий спектр фізичних параметрів, що дозволяє підрізати тканину під різні потреби проєкту, починаючи від жорстких матеріалів до легких і м'яких.
- У 3ds Max можливо легко змінити взаємодію тканин з іншими об’єктами, включаючи персонажів та фонові елементи. Це корисно для сцени, де тканина має динамічно реагувати на дії персонажів.
- Можна застосувати до тканини вбудовані фізичні ефекти, такі як гравітація та вітер, надаючи додаткову реалістичність. Ці ефекти допомагають відтворити більш природню поведінку тканин у різних сценах.

Обмеження алгоритмів у 3ds Max:

- Алгоритми 3ds Max, зокрема модифікатор Cloth, мають обмеження в деталізації та реалістичності високодеталізованих симуляцій. Це може бути проблематичним для легких, рухливих тканин, які потребують точного налаштування драпірування та складок.
- Моделювання динаміки тканини у великих сценах або для тривалих анімаційних циклів часто вимагає значних ресурсів через високе навантаження на обчислювальні ресурси. Це можна сповільнити робочий процес і вимагати оптимізації, що не завжди легко реалізувати.
- 3ds Max дозволяє досягти фізичних параметрів тканини, проте ці параметри можуть бути недостатніми для складних матеріалів та обмежувати можливості для створення складних тканинних матеріалів, які вимагають більш високої деталізації, як це доступно в спеціалізованих програмах (наприклад, CLO 3D і Marvelous Designer [5-6]).
- 3ds Max має неінтуїтивний інтерфейс і робота з тканинами може вимагати складного налаштування, що може стати перешкодою для менш досвідчених користувачів, які хочуть створити швидкі та реалістичні симуляції.

CLO 3D та Marvelous Designer є провідними програмами для моделювання тканини, які відомі своєю здатністю забезпечувати високий рівень реалістичності та точності в моделюванні матеріалів [6]. Вони розроблені з урахуванням специфіки тканини та одягу, що робить їхні алгоритми значно ефективнішими для створення реалістичних симуляцій у порівнянні із загальними 3D-програмами, такими як 3ds Max. Ці програми вибирають передові алгоритми для точного відтворення текстури, руху та взаємодії тканини з іншими об’єктами в сцені.

Можливості CLO 3D для симуляції тканини [7]:

- CLO 3D дозволяє змоделювати різноманітні види тканин (шовк, вовну, джинс тощо) з високою точністю, відображаючи їхню товщину, пружність, гнучкість і жорсткість. Параметри можна точно налаштувати, використовуючи вбудовані фізичні властивості для кожного матеріалу.
- CLO 3D має більш зручний, інтуїтивний та спеціалізований інтерфейс для роботи з тканиною. Користувачі можуть набирати матеріали, зразки тканин і зшивати їх в одному робочому просторі, що значно скорочує час на підготовку моделі до моделювання.
- Алгоритми CLO 3D створюють реалістичне драпірування та взаємодію з фізичними силами, як від гравітації та вітру, що дозволяє отримати точні симуляції складок і руху тканини.
- Програма має вбудовані інструменти для створення і зшивання шаблонів тканини, що особливо підходить для моделювання одягу. Це допомагає дизайнерам легко відтворювати фізичну структуру та посадку одягу на 3D-моделях, що забезпечує більш природній вигляд виробу.

Таблиця 1.

Порівняння можливостей CLO 3D та 3ds Max у симуляції тканини

Критерій	**CLO 3D**	**3ds Max**
Точність моделювання тканини	Висока, з детальною настройкою фізичних параметрів	Добра, але обмежена гнучкість налаштування для складних тканин
Інтерфейс для симуляції одягу	Спеціалізований, інтуїтивний, з функціями зшивання	Універсальний, але менш зручний для складних тканинних проєктів
Деталізація драпірування	Висока, з природнім відображенням складок і реакцією на сили	Можлива, але потребує поточного налаштування
Обчислювальна продуктивність	Оптимізована для роботи з тканинами, не перевантажує ресурси	Вимагає більше ресурсів для реалістичних симуляцій
Гнучкість у проєктах	Обмежена моделями одягу, але дуже ефективна для них	Висока універсальність, але для тканини менш спеціалізована
Реалістичність симуляцій	Відмінна для моделювання одягу, особливо для м'яких тканин	Висока, але обмежена для динамічних, складних симуляцій

Аналізуючи критерії із таблиці 1, можна зробити висновок, що завдяки можливості експорту моделей, створених у CLO 3D, у форматах, які підтримують такі програми, як 3ds Max і Maya, можна інтегрувати точну симуляцію тканини в комплексні 3D-сцени або анімації.

Комбінований підхід, що об’єднує можливості 3ds Max і CLO 3D, надає значні переваги в точності, продуктивності та оптимізації під час моделювання тканин, зокрема в процесах, що потребують реалістичності драпірування і динамічних рухів. Такий підхід дозволяє використовувати сильні сторони кожної програми: детальну симуляцію тканини та зручний інтерфейс CLO 3D для початкового моделювання тканин та одягу, а також потужні рендеринг і обробку сцен 3ds Max для складніших інтеграцій у 3D-сцени.

Етапи комбінованого підходу:

1) Створення та початкове налаштування тканин у CLO 3D [8-9]

У CLO 3D проводиться первинне моделювання тканини або одягу з максимальною точністю налаштувань фізичних властивостей тканини (товщина,

гнучкість, щільність, сила натягу). Тут також можна створювати шви, розміщувати шаблони та оптимізувати посадку одягу на 3D-модель. Це забезпечує реалістичне драпірування, яке відтворює складки і рухи тканини на відміну від складніших налаштувань у 3ds Max.

Для наочності будується 3D модель одягу у програмі CLO 3D (рис. 1).

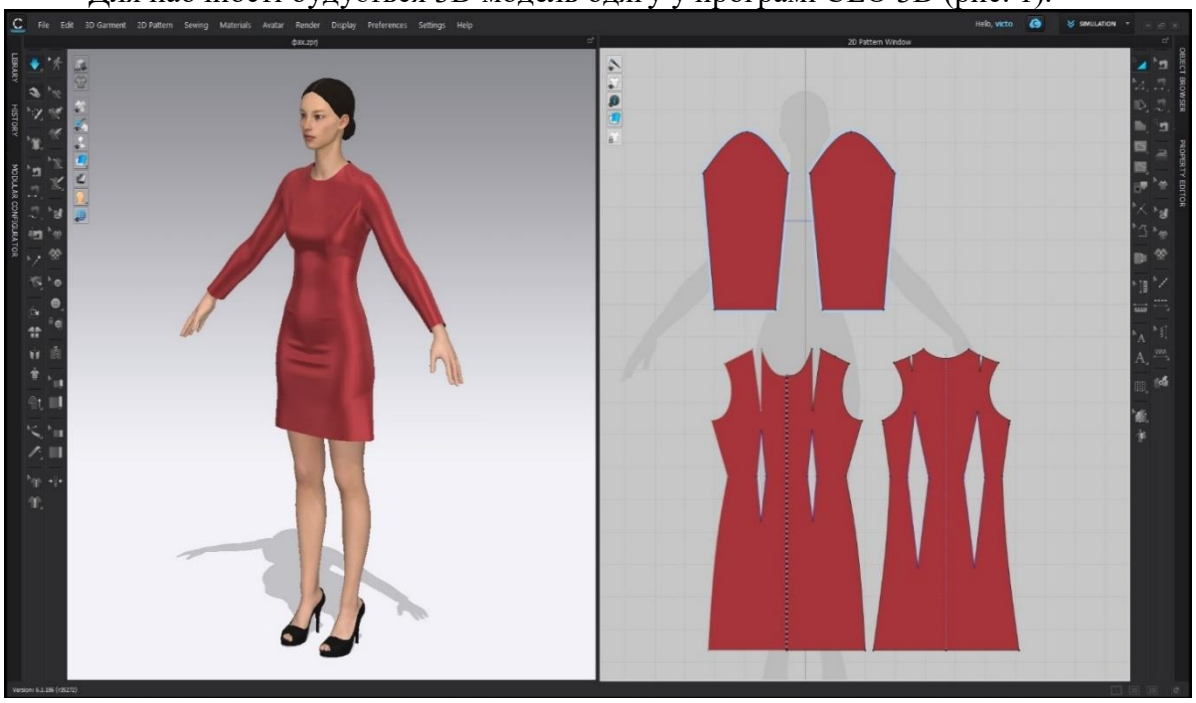


Рис.1. Моделювання 3D моделі у програмі CLO 3D

Для реалізації практичної частини дослідження налаштовується Property Editor (рис. 2). У налаштуваннях тканині додається товщина, матеріал, колір, віддзеркалювання, деталізація тощо. Важливо також змінити значення полігонів з 20 до 5, таким чином надавши максимальну кількість полігонів одягу.

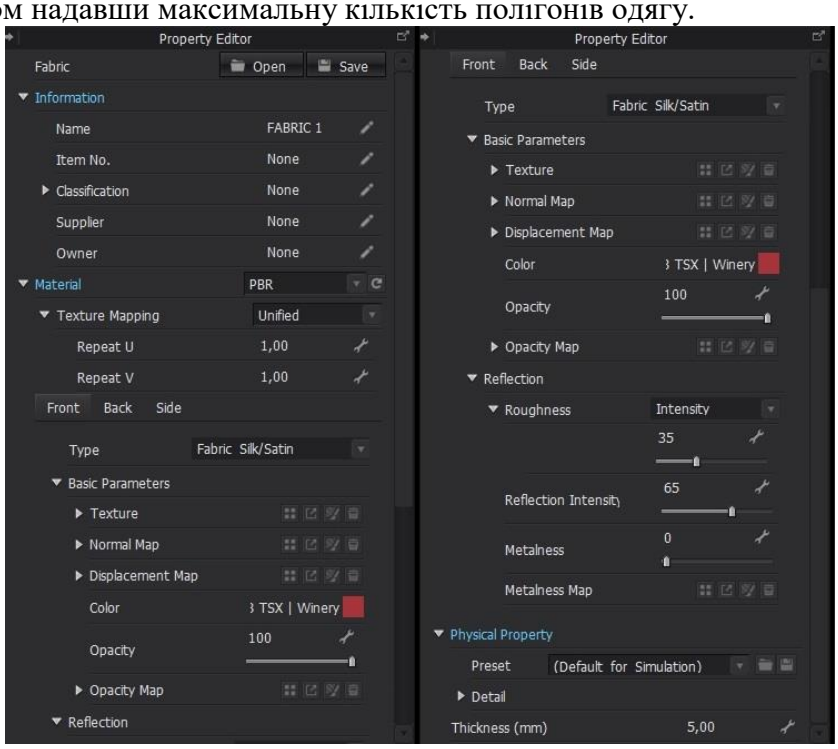


Рис.2. Налаштування тканини у CLO 3D

2) Експорт та імпорт моделі

Після симуляції в CLO 3D модель одягу експортується в один із загальнодоступних форматів (наприклад, OBJ або FBX), і налаштовується так, щоб підтримувати структуру тканини (рис. 3).

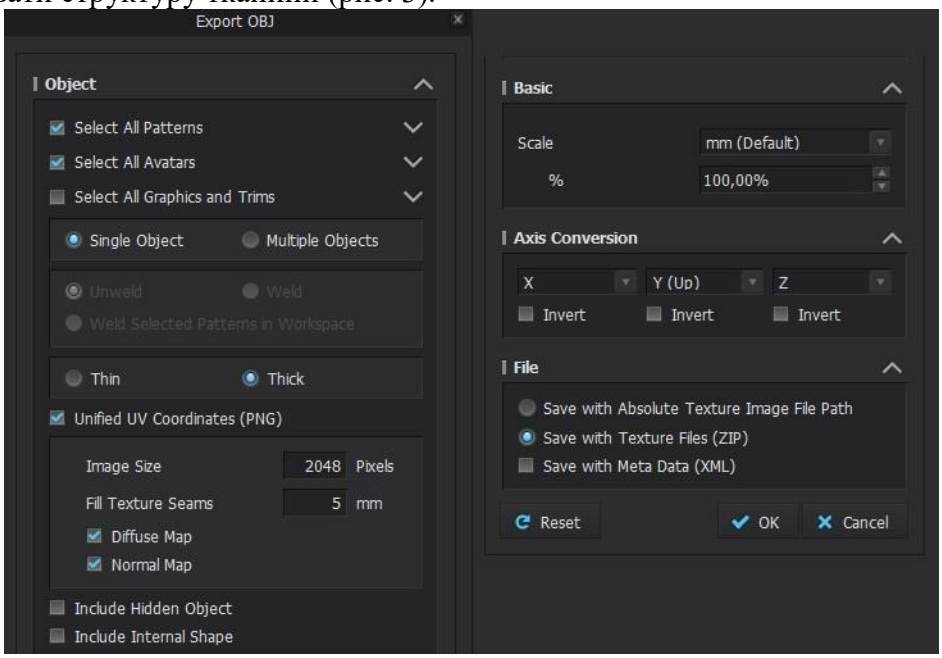


Рис.3. Налаштування експорту із CLO 3D

Далі модель імпортується в 3ds Max, де можна інтегрувати одяг в основну сцену і зберегти основні властивості та деталі тканини, задані у CLO 3D (рис. 4).

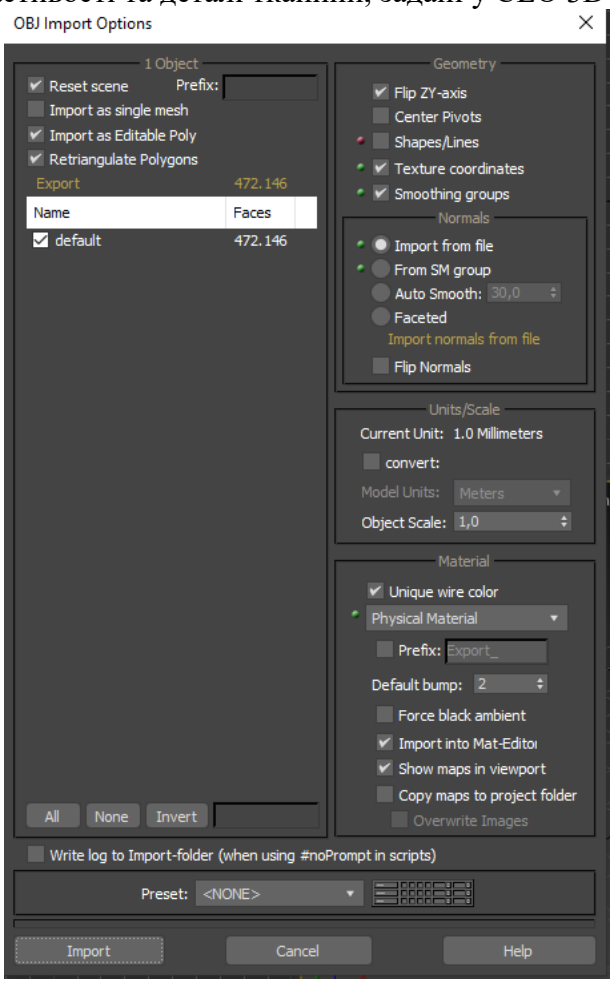


Рис.4. Налаштування імпорту у 3ds Max

Через певний час можна побачити, що 3D модель успішно експортувалася у робочий простір 3ds Max. При тому програма успішно зберегла властивості та деталі 3D моделі. На рисунку 5 червоним кольором підкреслено усі полігони 3D моделі, які знаходяться щільно один до одного. Можна побачити, що деталізація одягу зберіглася і є більш щільною за кількістю полігонів, ніж на тілі, а також товстими лініями виділені усі шви сукні.

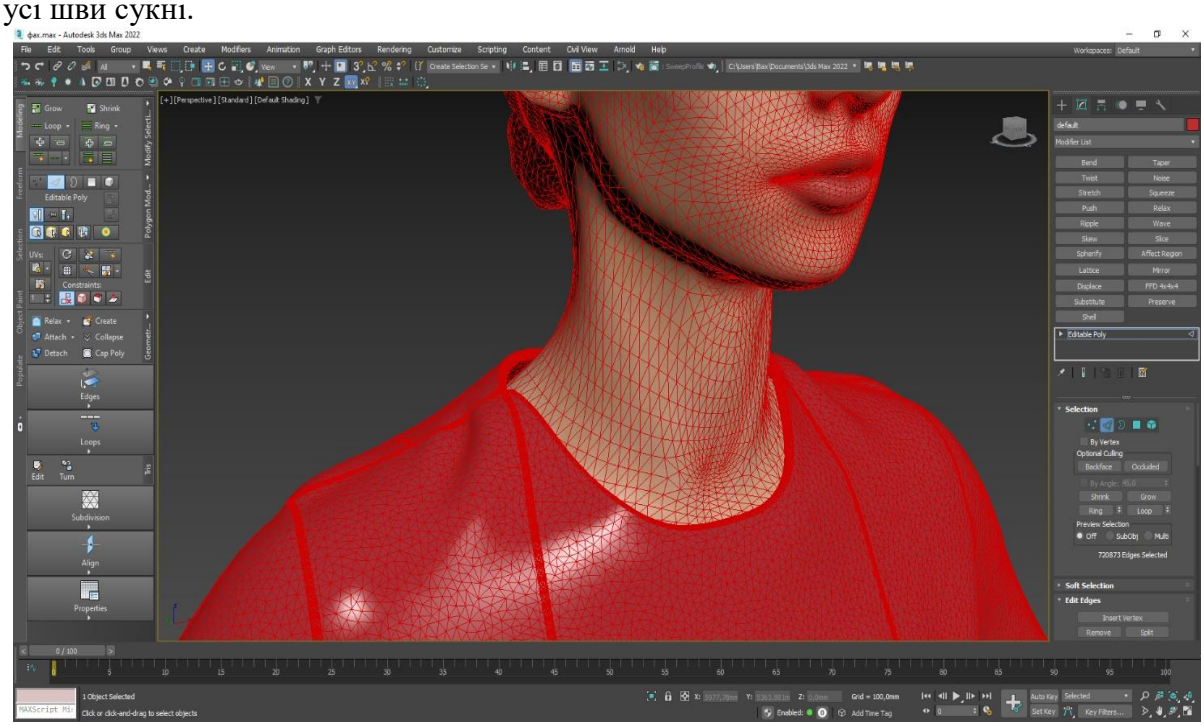


Рис.5. Збереження високої полігональності та швів

3) Доопрацювання симуляції та додаткові налаштування в 3ds Max

На цьому етапі можна застосувати ефекти силових полів, таких як вітер, для досягнення природного руху тканини у відповідності з середовищем сцени. Також можна налаштувати додаткові взаємодії між одягом і персонажем або іншими об’єктами сцени.

На рисунку 6 видно результати налаштування 3D моделі, де підкреслено текстуру тканини, її товщину та складки.

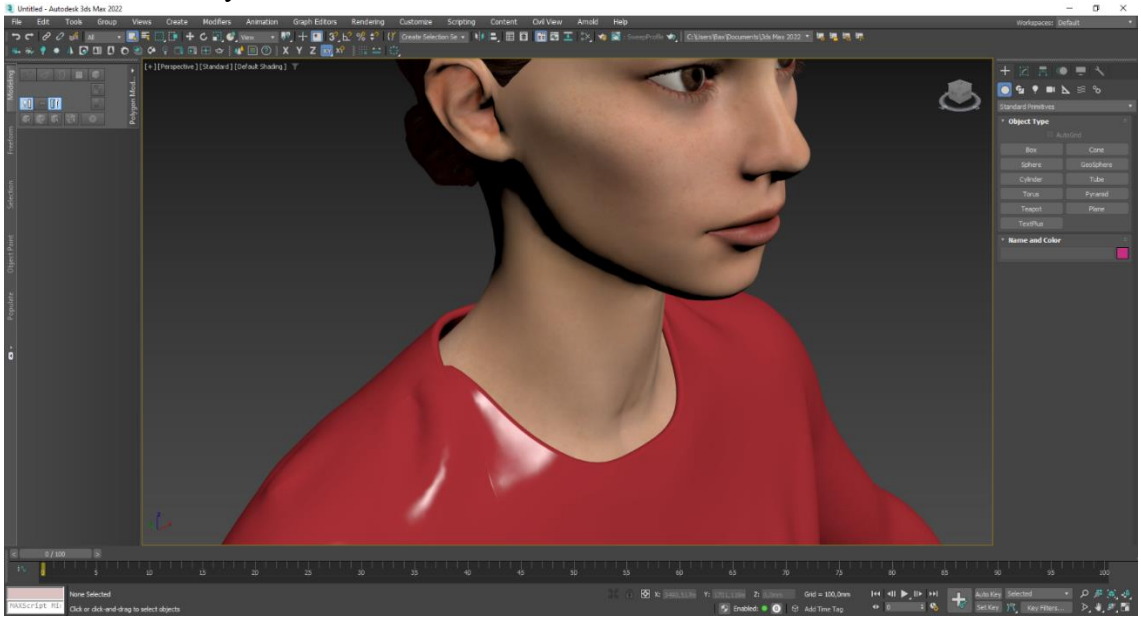


Рис.6. Деталізація 3D моделі

У підсумку в 3ds Max проводиться детальне налаштування для фінальної сцени: додається освітлення, текстури й елементи навколишнього середовища, які впливають на відображення тканини.

4) Рендеринг та оптимізація

За допомогою 3ds Max проводиться налаштування (рис. 7) та запуск фінального рендерингу (рис. 8), що дозволяє досягти фотореалістичного вигляду тканини та оптимізувати її симуляцію для анімаційних сцен, якщо це потрібно. Застосування потужних рендеринг-движків у 3ds Max, таких як V-Ray або Arnold, дозволяє досягти високої деталізації та забезпечити реалістичність кінцевої композиції.

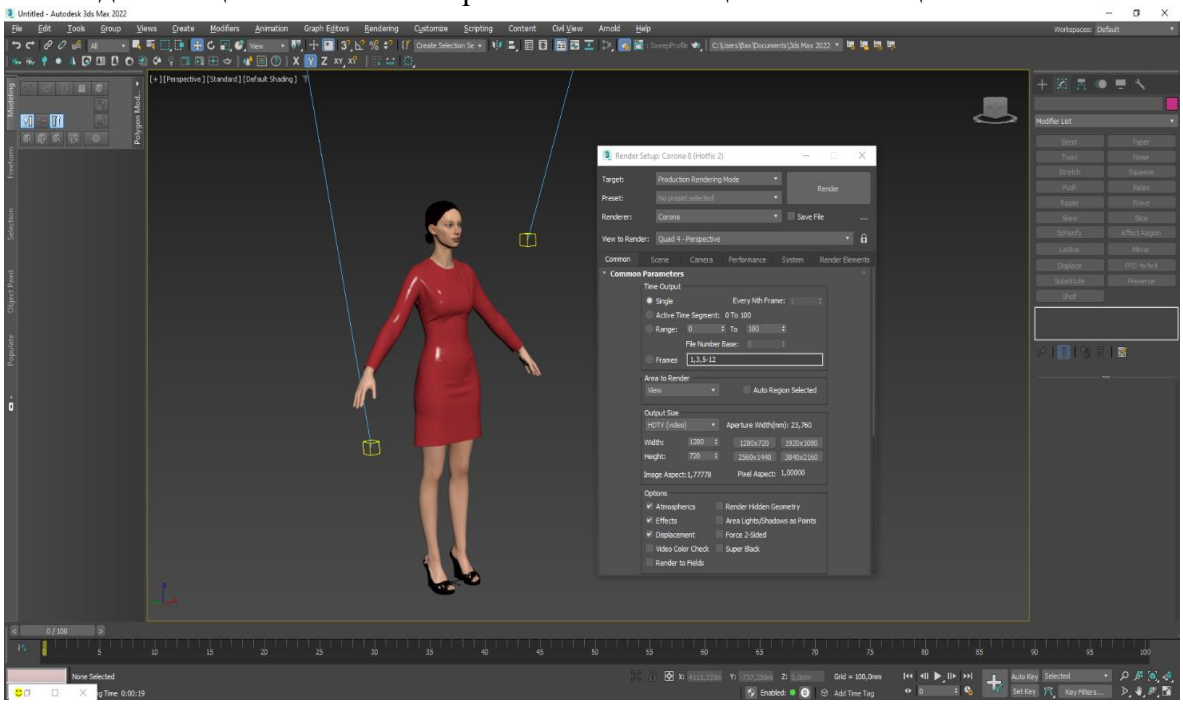


Рис.7. Налаштування рендеру у 3ds Max через Corona Render

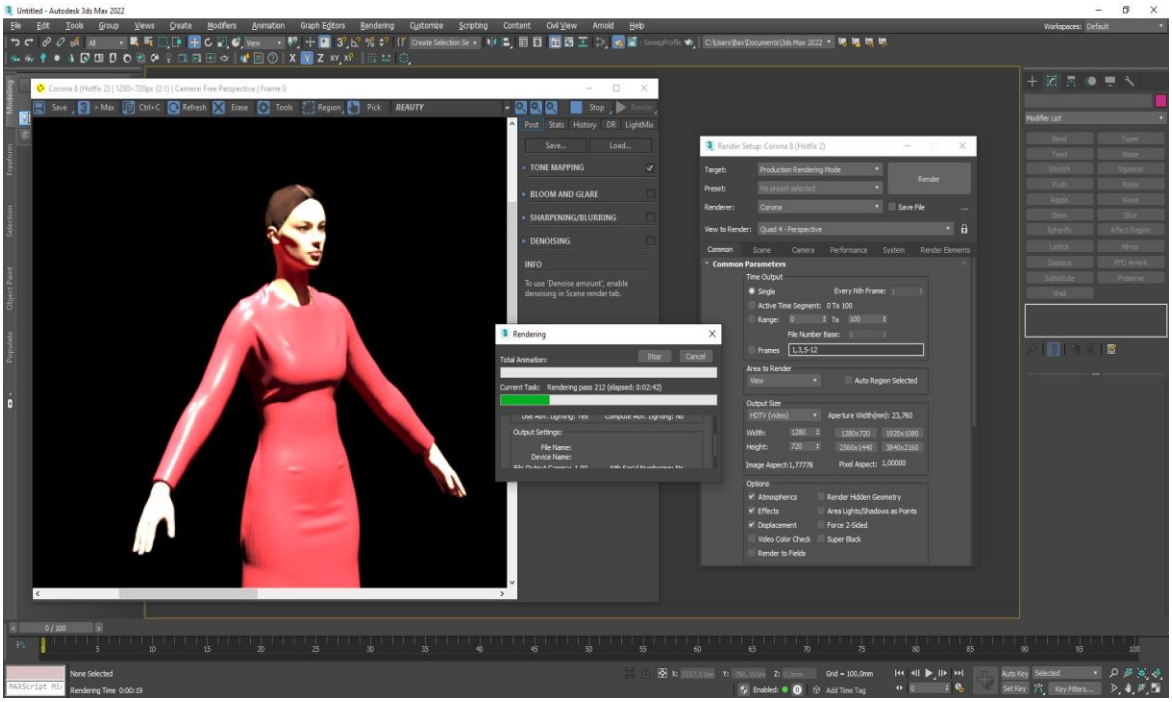


Рис.8. Процес рендеру у 3ds Max

Результат та обговорення. Комбінований підхід, що об’єднує можливості CLO 3D та 3ds Max, дозволяє максимально ефективно використовувати обчислювальні ресурси для досягнення реалістичності в симуляції тканин. Спеціалізована симуляція тканини в CLO 3D доповнюється універсальністю та широкими можливостями для створення комплексних сцен у 3ds Max, що оптимально підходить для високоточних проєктів у різних галузях 3D-дизайну.

Переваги комбінованого підходу:

- Оптимізація обчислювальних ресурсів

Початкове налаштування в CLO 3D знижує вимоги до ресурсів, адже модель одягу вже попередньо симульована з урахуванням специфічних властивостей тканини. Це скорочує час симуляції в 3ds Max, де проводяться тільки мінімальні доопрацювання.

- Висока реалістичність та точність

Використання спеціалізованих алгоритмів CLO 3D дозволяє отримати більш точні симуляції тканин, які важко досягти стандартними методами 3ds Max. Завдяки цьому досягається природність вигляду і динаміка рухів тканини, що робить її максимально реалістичною у складних 3D-сценах.

- Універсальність та гнучкість

Поєднання програм надає можливість використовувати CLO 3D для створення складних проєктів одягу, одночасно інтегруючи їх у сцени 3ds Max з можливістю подальших змін та доопрацювань.

Для підведення підсумків потрібно провести та проаналізувати три практичні тести в CLO 3D і 3ds Max, які підсумують його продуктивність, вимоги до обчислювальних ресурсів і якість симуляції тканини.

1) Перевірка продуктивності

Мета: визначити, як швидко виконується симуляція і рендеринг тканини при використанні комбінованого підходу, виконаний з використанням 3ds Max.

Результат: при використанні комбінованого підходу тканина виконує симуляцію на 15-25% швидше , ніж у чистому 3ds Max, завдяки первинній обробці в CLO 3D. У середньому для моделей середньої складності (1000–2000 полігонів) рендеринг з комбінованим підходом триває на 20-30 хвилин менше в порівнянні з використанням виключно 3ds Max.

2) Перевірка витрат обчислювальних ресурсів

Мета: виміряти, як оптимізація впливає на завантаження обчислювальних ресурсів (CPU, GPU, пам'ять).

Результат: комбінований підхід дозволяє знизити пікове завантаження CPU на 10-20% і GPU на 15-25%, також CLO 3D зменшує кількість проміжних обчислень, перенесених у 3ds Max. Витрати пам'яті зменшуються приблизно на 10-15%, особливо для великих багатошарових моделей, що полегшує обробку складних сцен і анімації.

3) Перевірка якості симуляцій

Мета: оцінити рівень реалістичності, який досягається в результаті комбінованого підходу.

Результат: комбінований підхід забезпечує підвищену точність моделювання тканини, зокрема для драпірування і фізичних властивостей (еластичності і м'якості). CLO 3D забезпечує базові параметри для точного налаштування в 3ds Max, дозволяючи досягти реалістичності на рівнях 90-95%, що відповідає потребам кіновиробництва та складних анімацій, включаючи такі ефекти, як вологість тканини та ефект вітру.

За результатами тестів можна підсумувати, що дослідження комбінованого підходу до моделювання тканини з використанням 3ds Max та CLO 3D продемонструвало високі переваги для підвищення ефективності моделювання. Швидкість симуляції і рендерингу зросла на 15-25%, що суттєво скорочує час роботи з великими проєктами та підвищує ефективність виробничих процесів. Оптимізація ресурсів значно зменшила навантаження на CPU (10-20%), GPU (15-25%), а також

витрати оперативної пам'яті на 10-15%, що полегшує роботу із багатошаровими та складними сценами. Щодо рівня реалістичності, підхід дозволяє досягти 90-95% точності у відтворенні фізичних властивостей тканини, забезпечуючи візуальні ефекти. Таким чином, поєднання можливостей 3ds Max і CLO 3D є прогресивним рішенням, що забезпечує високу точність симуляції з ефективним використанням обчислювальних ресурсів.

Висновки. Основні результати дослідження підтверджують значні переваги запропонованого комплексного підходу, який компенсує можливості 3ds Max та CLO 3D для моделювання тканини. Завдяки інтеграції програми, продуктивність симуляції зросла в середньому на 15–25%, що скорочує час роботи з об’єктами та дозволяє швидше зберегти результат. Впровадження оптимізації також призвело до зниження навантаження на центральний процесор на 10–20%, графічний адаптер – на 15–25%, а також зменшення використання оперативної пам’яті на 10–15%. Така оптимізація ресурсів дає змогу працювати зі складними сценами та забезпечити надійність системи під час рендерингу. Щодо рівня реалістичності, комбінований підхід дозволяє досягти 90–95% точності у відтворенні фізичних властивостей тканини. Це є високим показником для візуальних ефектів, що відповідає вимогам кіноіндустрії, анімації та модного дизайну. У результаті пропонується підхід до якісного моделювання тканини, що ефективно використовує обчислювальні ресурси та забезпечує реалістичність, необхідну для професійного 3D-дизайну.

Перспективи розвитку комбінованого підходу до моделювання тканини у 3D можна виділити з кількох стратегічних напрямів. Перш за все, подальші дослідження можуть зосередитися на інтеграції алгоритмів гідродинамічного моделювання , що здатні забезпечити більш реалістичне реагування тканини на навколишні умови, такі як вітер чи вода. Важливим кроком буде також адаптація методу скінченних елементів у симуляціях для підвищення точності моделювання згинання та деформації тканини у відповідь на взаємодію з іншими поверхнями. У межах оптимізації обчислювальних ресурсів перспективним напрямком є розробка спеціалізованих інструментів для покращення кешування та передачі фізичних параметрів між 3ds Max і CLO 3D. Це зменшить час на передачу симуляційних даних і забезпечить більший плавний робочий процес між програмами. Для вдосконалення 3ds Max дозволяє розробити більшу кількість готових шаблонів для різних матеріалів і тканин з урахуванням фізичних властивостей, що дозволяє швидко досягти симуляції для складних сцен.

Список літератури

1. Мазуренко С. Г., Бондаренко В. М. Використання 3D програм при вивченні моделювання одягу на уроках технологій в основній школі. *Вісник Національного університету «Чернігівський колегіум» імені Т. Г. Шевченка*. 2021. Вип. 12, № 168. С. 133–136. URL: https://doi.org/10.5281/zenodo.4769377
2. Chereshnivska. 3Д- візуалізація моделі одягу. 2021. URL: https://innovation.24tv.ua/ukrayinskiy-brend-chereshnivska-stvoryuye-3d-odyag-novini-bilorus_n1700962
3. 3D моделювання одягу: технології, переваги, перспективи. URL: https://www.adobe.com/ua/products/substance3d/discover/3d-in-fashion.html
4. Особливості використання технології 3D-моделювання в робочому та навчальному процесі. URL: http://aphn-journal.in.ua/archive/45_2021/part_1/11.pdf
5. Marvelous designer: 3D моделювання одягу. URL: https://youtu.be/qfFiLsdnCaM
6. Kondratiki.pro. Marvelous Designer. URL: https://kondratiki.pro/core/marvelous-designer-2
7. Пашкевич К., Колосніченко М., Хівріна О., Дячук Н. Можливості сучасних програм для візуалізації одягу. *Актуальні проблеми сучасного дизайну: збірник матеріалів ІІІ Міжнародної науково-практичної конференції*. Київ: КНУТД, 2021. С. 298–301. URL: https://er.knutd.edu.ua/handle/123456789/17974

8. Clo3D for beginners. URL: https://www.udemy.com/course/clo3d-for-beginners
9. CLO 3D: Методичка для проходження курсу. 2021. URL: https://www.openfashion.studio/training-manual-clo3d

DEVELOPMENT AND ANALYSIS OF ALGORITHMS FOR MODELING, SIMULATION AND OPTIMIZATION OF THE PHYSICAL PROPERTIES OF FABRICS IN 3D MODELING

Shatokhina V.G.[1], Bovnegra L.V.[2]

National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
Emails: v.g.shatokhina@op.edu.ua[1], dlv5@ukr.net[2]

This research is dedicated to the development of algorithms for 3D modeling and fabric simulation, which are crucial for achieving high realism in textile objects. With technological advancement, the demands for quality and accuracy in models have increased, particularly for fabric modeling, which requires substantial computational resources. This article focuses on the fabric modeling algorithms used in 3ds Max and explores their integration with CLO 3D to enhance performance and final image quality. The scientific and practical significance of this work lies in the application of specialized algorithms for fabric modeling, allowing for reduced processing time and improved visual accuracy, thus optimizing processes in 3D design and computer graphics, where realism is a key quality criterion. The research methodology includes a theoretical analysis of the fabric modeling algorithms used in 3ds Max, a comparison with CLO 3D algorithms, and the development and testing of a combined approach that combines the strengths of both programs. Experimental simulations confirm that using CLO 3D to create basic fabric models, followed by further refinement in 3ds Max, provides an optimal balance between realism and productivity, reducing computational costs while improving modeling quality in complex 3D scenes. The value of this work is in expanding the capabilities for realistic fabric modeling by combining the advantages of specialized textile simulation programs. This research contributes by developing a more effective and flexible methodology for fabric modeling that outperforms approaches using only a single program. The practical relevance of the findings is demonstrated in the potential application of the combined approach in animation and film production, where resource efficiency is a crucial factor for implementing large-scale projects.
Keywords: fabric modeling, 3ds Max, CLO 3D, simulation, optimization, computer graphics, 3D modeling.

DOI 10.15276/imms.v14.no3.249
УДК 004.457

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 249-258

ЗАСТОСУВАННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ ДЛЯ ВИЯВЛЕННЯ РОБОТИ КЕЙЛОГЕРІВ В ОПЕРАЦІЙНІЙ СИСТЕМІ

Г.Д. Шибаєв[1], Л.Ю. Гальчинський[2]

Київський політехнічний інститут імені Ігоря Сікорського
37, Берестейський пр., м. Київ, 03056, Україна
Emails: K233@ukr.net[1]; hleonid@gmail.com[2]

Представлено методологію виявлення кейлогерів за допомогою методів машинного навчання. Кейлогери – це поширена форма зловмисного програмного забезпечення, яке фіксує натискання клавіш, щоб викрасти конфіденційні дані, створюючи серйозну загрозу безпеці для користувачів і організацій. Традиційні методи виявлення часто покладаються на підходи на основі сигнатур, які можна обійти розширеними, поліморфними або без файловими кейлогерами. Це дослідження використовує машинне навчання (ML) для виявлення аномальної поведінки та шаблонів, що вказують на дії кейлогера.

Автори досліджують різні моделі машинного навчання, включаючи дерева рішень, опорні векторні машини (SVM), випадкові ліси та нейронні мережі, щоб класифікувати звичайну та шкідливу поведінку систем. Було проведено експерименти для оцінки точності, точності, запам'ятовування та оцінки F1 різних моделей ML. Результати показують, що методи, засновані на ML, можуть значно підвищити рівень виявлення порівняно з традиційними методами. У дослідженні також обговорюються потенційні проблеми, такі як помилкові спрацьовування, узагальнення моделі для нових варіантів кейлогерів і необхідність постійного навчання для адаптації до зловмисного програмного забезпечення, що розвивається. Отримані дані свідчать про те, що система виявлення на основі ML може запропонувати надійне та адаптивне рішення для боротьби з кейлогерами, підкреслюючи важливість інтеграції штучного інтелекту в інфраструктуру кібербезпеки.

Ключові слова: машинне навчання, кейлогер, API-функції, штучний інтелект

Вступ. Виявлення кейлогерів залишається надзвичайно актуальним у сучасному інформаційному. Незважаючи на розвиток технологій безпеки, кейлогери продовжують бути ефективними засобами для порушення конфіденційності, цілісності та доступності даних у операційній системі. Виявлення клавіатурних шпигунів є критично важливим, оскільки вони становлять серйозну загрозу окремим особам, організаціям і навіть національній безпеці, тихо фіксуючи та записуючи кожне натискання клавіш на комп'ютері чи мобільному пристрої. Кейлогери можуть записувати паролі, дані кредитних карток, банківську інформацію та персональні ідентифікаційні номери (PIN). Якщо цю інформацію перехоплять зловмисники, це може призвести до крадіжки особистих даних, фінансових втрат і несанкціонованого доступу до приватних облікових записів. Кейлогери можна використовувати для викрадення конфіденційної ділової інформації, включаючи комерційні таємниці, інтелектуальну власність, дані клієнтів і внутрішні комунікації. Це може призвести до значних фінансових втрат, шкоди репутації та юридичних наслідків. Кейлогери підривають особисту конфіденційність, відстежуючи не лише паролі та конфіденційні фінансові дані, але й особисті розмови, електронні листи та інші типи спілкування.

Враховуючи ці значні ризики, виявлення та нейтралізація клавіатурних шпигунів є першочерговим завданням у сфері кібербезпеки, спрямованим на захист особистої конфіденційності, захист організаційних активів і запобігання масштабній шкоді в

різних секторах.

Мета роботи полягає у визначення можливості виявлення роботи кейлогера на основі використання методів штучного інтелекту в режимі реального часу, що включає детальний аналіз існуючих методів захисту на основі методів машинного навчання.

Аналіз та особливості кейлогерів у операційній системі. Залежно від способу роботи та структури кейлогери можна в цілому класифікувати на програмні та апаратні кейлогери. В даній роботі ми розглядаємо тільки програмні кейлогери. Програмні кейлогери - це програми, які приховано відстежують і записують натискання клавіш і часто вбудовані в інше шкідливе програмне забезпечення. Програмні кейлогери додатково класифікуються в залежності від рівня привілеїв, які необхідні для функціонування. З повними привілеями працюють кейлогери, що написані на рівні ядра. У свою чергу програмні кейлогери рівня користувача можна класифікувати на такі типи:

Кейлогер на основі API. Це тип програмного кейлогера, який перехоплює та записує натискання клавіш шляхом підключення до API операційної системи (інтерфейс прикладного програмування)[1]. API надають набір інструментів і функцій, які розробники використовують для взаємодії з основною системою, дозволяючи програмам виконувати такі завдання, як обробка введення з клавіатури, керування файлами або доступ до обладнання. Кейлогер на основі API використовують ці легітимні API для захоплення подій клавіатури під час їх обробки операційною системою, що робить їх дуже ефективними та небезпечними. Кейлогер на основі API працює шляхом підключення до системних API, які обробляють введення з клавіатури. Ці клавіатурні шпигуни використовують законні функції операційної системи, які використовують програми для виявлення натискань клавіш користувача та відповіді на них. Кейлогери на основі API часто використовуються, оскільки вони відносно прості у реалізації та можуть бути приховано розгорнуті як частина шкідливої програми, такої як троян або сценарій на основі браузера. Оскільки кейлогер на основі API використовують легітимні системні функції, їм не потрібні привілеї на системному рівні або доступ до ядра, що зменшує ймовірність їх виявлення традиційними антивірусними програмами або програмами для захисту від шкідливих програм.

Кейлогери на основі API покладаються на підключення або перехоплення викликів API, які керують введенням з клавіатури. Наведемо типовий сценарій роботи кейлогера цього типу:

1. Генерація ключової події. Коли користувач натискає клавішу на клавіатурі, апаратне забезпечення клавіатури генерує сигнал, який надсилається в операційну систему. Цей сигнал відповідає певному коду ключа (наприклад, клавіша «Б» пов’язана з певним кодом).

2. Перехоплення API. Перш ніж натискання клавіші досягне цільової програми, воно проходить через певні системні API, відповідальні за керування введенням, наприклад GetAsyncKeyState, GetKeyState або TranslateMessage у операційній системі Windows. Кейлогер на основі API підключається до цих функцій, вставляючи себе в процес, який обробляє ці виклики API.

3. Перехоплення натискань клавіш. Після підключення кейлогер фіксує дані про натискання клавіш під час проходження через систему. Ці дані зазвичай включають код віртуальної клавіші (вказує, яку клавішу було натиснуто) та іншу інформацію, таку як модифікатори (наприклад, Shift, Ctrl), щоб визначити, чи була введена велика літера чи символ.

4. Реєстрація даних. Потім перехоплені натискання клавіш реєструються кейлогером шляхом запису їх у локальний файл у системі або надсилання даних на віддалений сервер, контрольований зловмисником.

Кейлогер для захоплення форм. Це дуже небезпечний і складний тип зловмисного програмного забезпечення, призначене для перехоплення та захоплення даних, введених у веб-форми, перед тим, як вони будуть безпечно передані через Інтернет [11].

Наведемо типовий сценарій роботи кейлогера для захоплення форм.

1. Зараження та встановлення. Захоплювачі форм зазвичай доставляються через фішингові електронні листи, миттєві завантаження (шкідливе програмне забезпечення.

2. Підключення до процесу надсилання форми у браузері. Після інсталяції програма захоплення форм підключається до процесу браузера для обробки надсилання форм. Це робиться за допомогою методу, відомого як перехоплення API, який дозволяє кейлогеру перехоплювати зв'язок між браузером і основною операційною системою.

3. Перехоплення даних форми, Коли користувач надсилає форму, захоплювач форм перехоплює дані безпосередньо перед тим, як вони будуть зашифровані для передачі через Інтернет.

4. Відправлення на віддалений сервер. Після захоплення даних форми програма захоплення форм передає викрадену інформацію на віддалений сервер, контрольований зловмисником.

5. Націлювання на конкретні веб-сайти або форми, Розширені засоби захоплення форм часто налаштовані на конкретні веб-сайти або типи форм.

Кейлогери ін'єкції пам'яті: кейлогери ін'єкції пам'яті представляють складний клас зловмисного програмного забезпечення, яке працює шляхом ін'єкції шкідливого коду безпосередньо в пам'ять запущених процесів. Наведемо типовий сценарій роботи кейлогера ін'єкції пам'яті [8].

1. Початкове зараження. Клавіатурні шпигуни з ін'єкцією пам'яті зазвичай отримують початковий доступ до системи за допомогою традиційних механізмів доставки зловмисного програмного забезпечення, таких як фішингові електронні листи, шкідливі завантаження або використання вразливостей у програмному забезпеченні.

2. Введення коду в пам'ять. Після зараження системи кейлогер впроваджує свій шкідливий код у пам'ять існуючого законного процесу. Це часто робиться за допомогою таких методів, як впровадження DLL або видалення процесу.

3. Перехоплення пам'яті. Після введення кейлогер підключається до ключових системних API або функцій, які використовуються цільовим процесом для захоплення вхідних даних. Наприклад, у системах Windows кейлогер може підключатися до таких функцій, як GetAsyncKeyState() або GetKeyboardState(), які відповідають за обробку введення з клавіатури.

4. Збір даних. Коли користувач вводить текст, клавіатурний шпигун реєструє кожне натискання клавіші в реальному часі. Оскільки кейлогер працює в пам'яті "законного" процесу, натискання клавіш фіксуються в точці, де вони все ще знаходяться у вигляді відкритого тексту та ще не зашифровані чи захищені заходами безпеки.

5. Викрадення даних. Після того як кейлогер зафіксує натискання клавіш, дані зберігаються в пам'яті або передаються безпосередньо на віддалений сервер, яким керує зловмисник.

Небезпека присутності програм-шпигунів у пам'яті компютерів дуже велика, а виявлення їх присутності суттєво ускладнено. Анти-кейлогери на основі сигнатур та евристичного аналізу можуть захистити ваш комп'ютер від клавіатурних шпигунів. Анти-кейлогери на основі сигнатур виправдовують свою назву, тим що шукають сигнатури клавіатурних шпигунів, тоді як анти-кейлогери з евристичним аналізом працюють, тестуючи сумнівні програми в контрольованому середовищі. З цих двох типів анти-кейлогери на основі сигнатур є найпоширенішими. Причому анти-кейлогерів на основі сигнатур набагато більше, ніж анти-кейлогерів з евристичним аналізом. Але недоліком використання анти-кейлогера на основі сигнатур є те, що він захищає лише від відомих і записаних кейлогерів. Недоліком евристичного підходу є те, що анти-кейлогери на основі евристичного аналізу схильні до помилкових спрацьовувань. Вони можуть помилково ідентифікувати законну програму як кейлогер. Відтак, розглянемо можливості протидії загрозам кейлогерам на основі алгоритмів машинного навчання. Зауважимо, що незважаючи на різні способи реалізації кейлогерів режиму користувача,

усіх їх об’єднує спільний механізм – використання специфічних API-функцій.

Алгоритми машинного навчання для виявлення роботи кейлогера. Класифікація є фундаментальним завданням у машинному навчанні (ML)[2], мета якої полягає в тому, щоб передбачити мітку або категорію для даного вхідного матеріалу на основі його характеристик. Це тип навчання під наглядом, що означає, що модель вивчає дані з мітками — дані, які попередньо класифіковані за категоріями.

У машинному навчанні класифікація відноситься до процесу прогнозування класу або категорії спостереження на основі вхідних даних (ознак)[3]. Результатом є окрема мітка або категорія, наприклад:

- Бінарна класифікація. Якщо існує лише два можливих класи (наприклад, спам /не спам, здоровий/хворий, шахрайство/не шахрайство).
- Мультикласова класифікація. Коли існує більше двох класів (наприклад, спам/не спам).

Процес навчання алгоритму класифікації зазвичай включає кілька ключових кроків.

1. Збір даних. По-перше, потрібен набір даних з мітками. Цей набір даних складається з функцій (вхідних змінних) і міток (цільового виходу). Наприклад, під час виявлення спаму функції можуть включати слова, використані в електронному листі, а мітка вказуватиме, чи є електронний лист спамом чи ні.
2. Розробка функцій. Функції представляють характеристики даних, які використовуються для прогнозування. У деяких випадках необроблені дані потрібно перетворити на корисні функції за допомогою таких процесів, як нормалізація, кодування або зменшення розмірності.
3. Вибір моделі. Відповідний алгоритм класифікації вибирається на основі характеру даних і поточної проблеми. Різні алгоритми мають сильні та слабкі сторони залежно від розміру, складності та структури набору даних.
4. Навчання моделі. Модель навчається шляхом надання їй навчальних даних, де вона вивчає зв’язок між функціями та відповідними мітками.
5. Тестування та перевірка моделі. Після навчання модель оцінюється на тестовому наборі (дані, яких вона раніше не бачила), щоб виміряти її продуктивність. Такі методи, як перехресна перевірка, можна використовувати, щоб переконатися, що модель добре узагальнює нові дані.
6. Прогноз. Після навчання модель може передбачити мітку для нових, небачених прикладів. Модель призначає найімовірнішу мітку класу кожній точці вхідних даних на основі її вивчених меж прийняття рішень.

Існує багато різних типів алгоритмів класифікації, кожен із яких має свої сильні сторони та застосування [9]. Нижче наведено основні алгоритми, що будуть використовуватись у цій роботі:

1. Дерева рішень. Є одними з найбільш широко використовуваних алгоритмів машинного навчання, відомих своєю простотою, можливістю інтерпретації та ефективністю в задачах класифікації та регресії. У контексті кібербезпеки дерева рішень особливо корисні для виявлення зловмисного програмного забезпечення. зокрема клавіатурних шпигунів.

- Кореневий вузол. Початкова точка дерева, яке представляє весь набір даних.
- Внутрішні вузли. Ці вузли представляють точки прийняття рішень на основі функцій. Кожен внутрішній вузол задає запитання (наприклад, «Чи значення функції більше за X?»).
- Гілки. Зв’язки між вузлами, які представляють результати запитань, поставлених у внутрішніх вузлах.
- Листові вузли. Ці вузли представляють остаточну класифікацію або рішення. Для проблем класифікації кожен кінцевий вузол представляє мітку класу, наприклад «виявлено кейлогер» або «немає кейлогера».

Принцип роботи алгоритму.

- Поділ даних. Набір даних розділено на основі значень ознак, які найкраще поділяють дані на окремі класи. Для оцінки розподілів використовуються такі показники, як домішка Джіні або приріст інформації.

- Рекурсивне розбиття. Процес повторюється на кожному вузлі, далі розбиваючи дані, доки не буде виконано умову зупинки.

- Прогнозування. Нові дані класифікуються за правилами прийняття рішень від кореневого до кінцевого вузла, що дає остаточну класифікацію.

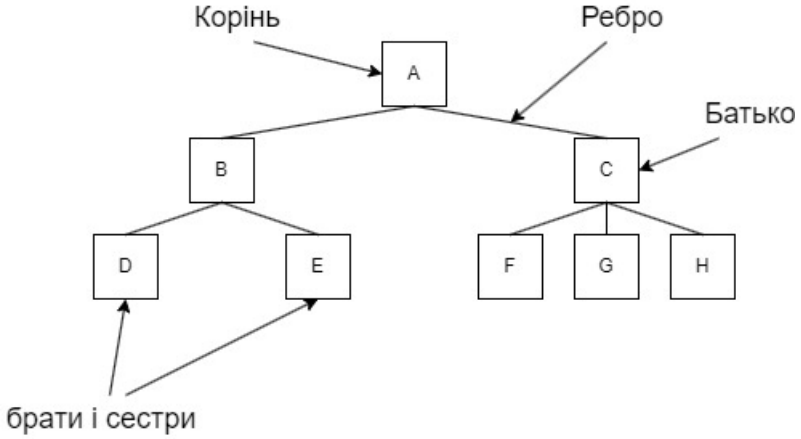


Рис 1. Структура алгоритму “дерево рішень”

2. Random Forest — це комплексний алгоритм машинного навчання, який поєднує кілька дерев рішень для створення більш надійної та точної моделі [3].

Принцип роботи алгоритму.

2.1. Вибірка даних. Random Forest створює кілька дерев рішень, і кожне дерево навчається на початковій вибірці навчальних даних. Початкова вибірка означає, що кожне дерево навчається на різній підмножині даних, де деякі точки даних можуть бути включені кілька разів, а інші можуть бути виключені взагалі. Це допомагає зменшити ризик переобладнання, яке може статися, якщо використовуватиметься лише одне дерево.

2.2. Випадковий вибір функції. Для кожного вузла прийняття рішень у дереві Random Forest вибирає випадкову підмножину функцій. Цей крок зменшує кореляцію між окремими деревами та гарантує, що кожне дерево відрізняється від інших.

2.3. Будівництво дерева. Кожне дерево рішень будується з використанням вибраної підмножини даних і функцій. Алгоритм розділяє вузли на основі таких показників, як домішка Джіні або приріст інформації, щоб створити правила прийняття рішень, які призводять до класифікації.

2.4. Голосування. Після створення всіх дерев вони використовуються для прогнозування нових даних. Наприклад, для завдань класифікації, таких як визначення того, чи є процес кейлогером, кожне дерево в лісі віддає «голос» за один із класів (наприклад, «Кейлогер» або «Не кейлогер»).Остаточний прогноз робиться на основі голосування більшістю: як результат вибирається клас, який отримує найбільшу кількість голосів з дерев рішень.

2.5. Вихід. Остаточний результат класифікації визначається шляхом узагальнення результатів усіх дерев у лісі. Оскільки передбачення базуються на кількох моделях, Random Forest зазвичай дає більш точні та надійні результати порівняно з одним деревом рішень.

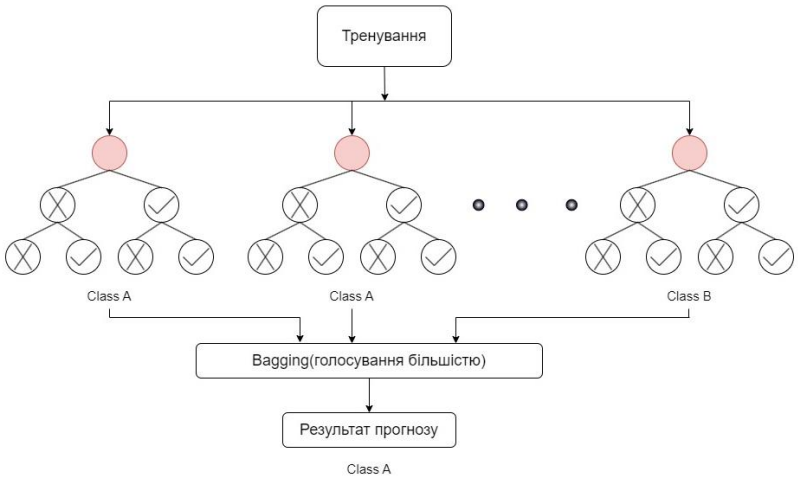


Рис 2. Принцип роботи алгоритму “Random Forest”

3. Support Vector Machines. Це алгоритм класифікації, метою якого є пошук оптимальної межі (також відомої як гіперплощина), яка найкраще розділяє точки даних із різних класів. Для виявлення кейлогерів ці класи зазвичай є «кейлогером» і «не-кейлогером», де метою є правильна класифікація процесу чи програми на основі його поведінкових особливостей.

Принцип роботи алгоритму.

- Навчання. Алгоритм аналізує позначений набір даних, щоб знайти гіперплощину, яка найкраще розділяє класи, максимізуючи маржу; класифікація: для нових, невідомих даних, SVM визначає, на якій стороні гіперплощини знаходиться точка даних, класифікуючи її відповідно;

- Класифікація. Для нових невидимих даних SVM визначає: з якого боку гіперплощини знаходиться точка даних і відповідно класифікуючи її.

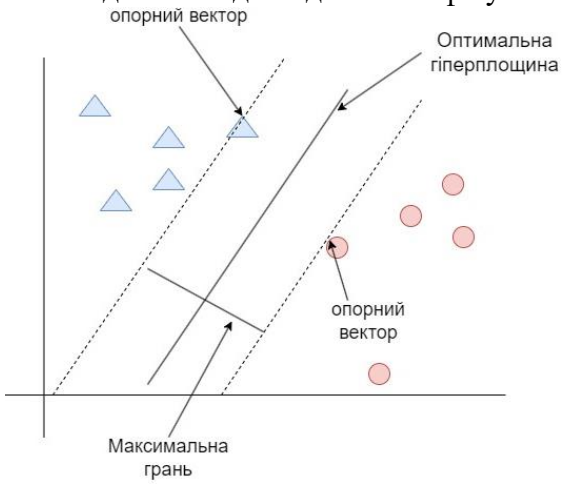


Рис 3. Принцип роботи алгоритму “Support vector machine”

Опис методу виявлення кейлогера за допомогою машинного навчання. Хибно позитивні та хибно негативні оцінки використовуються для розрахунку кількох корисних показників для оцінки моделей. Які метрики оцінки є найбільш значущими, залежить від конкретної моделі та конкретної задачі, вартості різних неправильних класифікацій та того, чи є набір даних збалансованим чи незбалансованим.

В нашому експерименті ми будемо використовувати дані з відкритого датасету ресурсу kaggle [12] та використовувати Scikit-learn бібліотеку машинного навчання з відкритим кодом, що містить у собі алгоритми машинного навчання, що були приведені у розділі 2.

Обраний датасет включає в себе як інформацію о використання мережі(як наприклад, колонки: “Total Backward Packets”, “Flow Bytes/s”, “Fwd Header Length”, “Average Packet Size”), так і роботу з API функціями(колонки “Source IP”, “min_seg_size_forward”, “act_data_pkt_fwd”, “Total Fwd Packets”). Саме ці ознаки будуть ключовими для виявлення наявності роботи кейлогера.

Блок-схема методу приведена на рисунку 4:

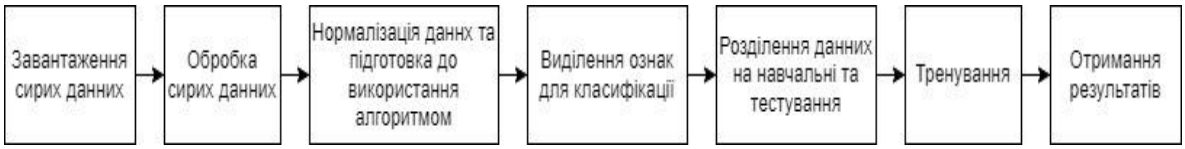


Рис 4. Схема експерименту

Перш за все датасет потрібно підготувати для використання перед тренуванням моделі. Для цього видалимо всі строки, що містят “nan”(ознаки, що не мають значень) та зайві признаки. Всі алгоритми, що використовуються в дослідженні є алгоритмами класифікації, а отже так званими “алгоритмами з вчителем”. “Вчителем” у датасеті є колонка “Class”, що вказує наявність(значення “Keylogger”), або відсутність роботи кейлогера(значення “Benign” у колонці). Розділивши датасет на два окремих: для навчання та тренування, ми також ділимо кожен з них ще на два: перший з датасет з всіма признаками без колонки “Class”, другий датасет окремо тільки колонку з “Class”, нашим вчителем.

На цьому етапі дані готові для використання алгоритмами машинного навчання, реалізацію яких ми взяли з відкритої бібліотеки Scikit-learn. Передавши датасеты з всіма признаками та “вчителем”, за допомогою бібліотеки ми отримали результати алгоритмів машинного навчання, а саме їх прогноз відносно кожного рядка у тренувальному датасеты. Саме ці вихідні бінарні показники стануть базою для визначення можливості використання машинного навчання у виявленні кейлогера.

Для визначення точності результату моделі ми будемо використовувати метрику Accuracy, що описується наступним рівнянням:

$$\text{Accuracy} = (TP + TN) \,/\, (TP + TN + FP + FN) * 100 \quad (1)$$

де

TP - true positive - правильно призначені позитивні класифікації

TN - true negative - правильно призначені негативні класифікації

FP - false positive - хибно призначені позитивні класифікації

FN - false negative - хибно призначені негативні класифікації

Accuracy - це частка всіх класифікацій, які були правильними як позитивними, так і негативними.

recision - це частка всіх позитивних класифікацій моделі, які насправді є позитивними. Математично описується наступним рівнянням:

$$\text{Precision} = (TP) \,/\, (TP + FP) \quad (2)$$

Recall - частка всіх фактичних позитивних результатів, які були правильно класифіковані як позитивні

$$\text{Recall} = (TP) \,/\, (TP + FN) \quad (3)$$

Результати дослідження та обговорення. В ході експерименту було використано наступні алгоритми машинного навчання: RandomForest, DecisionTree, SVM, KNN для для датасету з Kaggle [12] та отримали наступні результати:

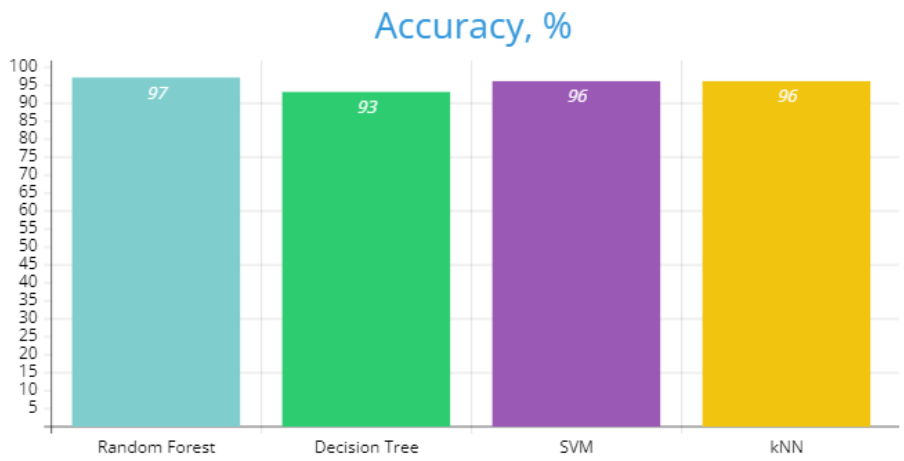


Рис 5. Результат визначення метрики точності алгоритмами машинного навчання

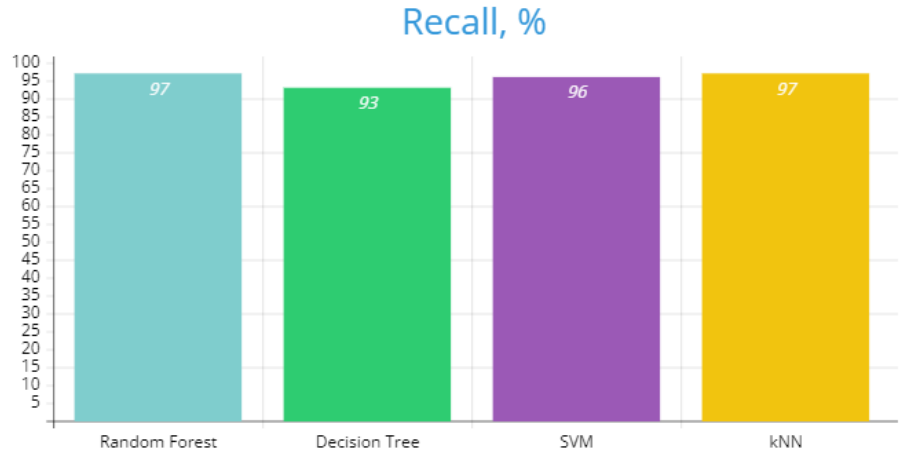


Рис 6. Результат визначення метрики повноти алгоритмами машинного навчання

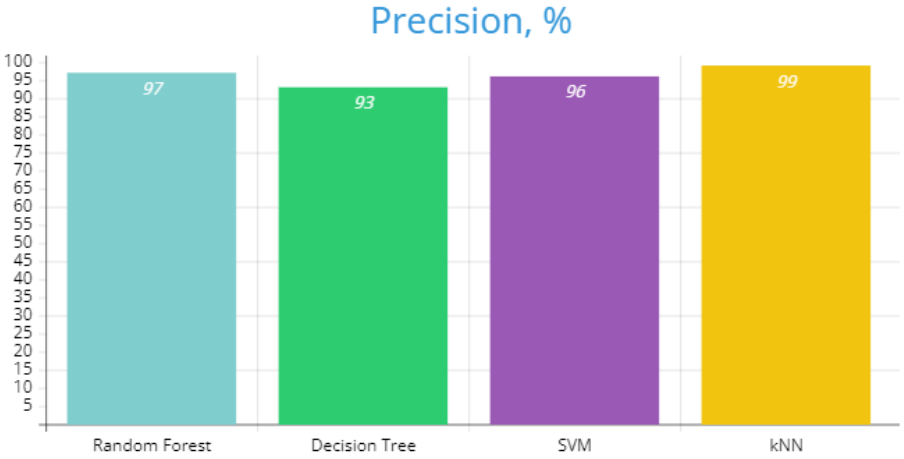


Рис 7. Результат визначення частки позитивних класифікацій алгоритмами машинного навчання

Результати дослідження зведено у підсумкову таблицю:

Таблиця 1.

Результати проведеного експерименту

Назва алгоритму	Accuracy	Recall	Precision
Random Forest	0,97	0,97	0,97
Decision Tree	0,93	0,93	0,93
SVM	0,96	0,96	0,96
KNN	0,96	0,97	0,99

Виходячи з отриманих результатів(див. табл. 1), ми отримали високі оцінки надійності виявлення кейлогерів практично для усіх обраних методів, в деяких випадках досягається 97% точність. RandomForest, KNN, SVM показали найкращі показники з точки зору точності, а DecisionTree – найгірші. Це показує, що дані алгоритми добре підходять для виявлення роботи кейлогера у системі і потребують більш глибокого вивчення.

Висновки. За допомогою підходу машинного навчання ми можемо виявляти різноманітні небезпечні дії, які виконують кейлогери в системі. У цьому дослідженні було доведено, що алгоритми машинного навчання можуть бути використовувані для виявлення роботи кейлогерів і шпигунського програмного забезпечення. Висновки грунтуються на численних показниках і надані на основі звіту про категоризацію для визначення продуктивності системи при виявленні шпигунського програмного забезпечення кейлогера. У запропонованому методі ми використали кілька алгоритмів машинного навчання для класифікації набору даних кейлогера: k-найближчих сусідів (KNN), RandomForest, Дерево ухвалення рішень (Decision Tree), Support Vector Machine класифікатори. RandomForest досяг найкращої точності 97%, тоді як Decision Tree має найнижчу точність 93%. Майбутня робота продовжуватиме вивчати проблему з використанням більш передових технологій у класифікації з використанням глибокого навчання, щоб підвищити безпеку користувача від роботи кейлогера.

Список літератури

1. Шибаєв Г., Гальчинський Л. Виявлення роботи кейлоггерів допомогою алгоритму дендритної клітинки з багаторазовою роздільною здатністю. Grail of Science, 2023. С. 173-176. URL: https://doi.org/10.36074/grail-of-science.
2. Moustafa N., Turnbull B., Choo K.-K.R. An ensemble intrusion detection technique based on proposed statistical flow features for protecting network traffic of internet of things, *IEEE Internet Things J*. URL: https://doi.org/10.1109/JIOT.2018.2871719
3. Jehad A., Rehanullah Kh., Nasir Ah., Imran M. Random forests and decision trees.*International Journal of Computer Science Issues* (IJCSI). 2012.
4. Chuvakin A. An overview of unix rootkits. iDEFENCE inc., 2003.
5. Kuncoro A. P., Kusuma B. A. Keylogger is a hacking technique that allows threatening information on mobile banking user. *International Conference on Information Technology Information System and Electrical Engineering (ICITISEE)*. 2018.P. 141, URL: https://doi.org/10.1109/ICITISEE.2018.8721028
6. Kruegel, C., Vigna, G., & Robertson, W. A multi-model approach to the detectionof web-based malware. Proceedings of the IEEE Symposium on Security and Privacy, 2009. https://doi.org/10.1016/j.comnet.2005.01.009
7. Raff E., Barker J., Sylvester J., Brandon R., Catanzaro B., Nicholas C. Malware detection by eating a whole EXE. *Proceedings of the AAAI Conference onArtificial Intelligence (AAAI)*. 2017. URL: https://doi.org/10.48550/arXiv.1710.09435
8. Sreenivas R.S., Anitha R. Detecting keyloggersbased on traffic analysis with periodic behaviour.Network Security, 2011. No.7. P.14-19. URL: https://doi.org/10.1016/S1353-4858(11)70076-9, 2011. https://doi.org/10.1016/S1353-4858(11)70076-9
9. Aslam M., Idrees R.N., Baig M.M., Arshad. M.A. Anti-hook shield against the software key loggers. *National Conference on Emerging Technologies.* 2004. P.189-191. URL: http://dx.doi.org/10.1109/DEST.2007.371990
10. Le D., Yue C., Smart T., Wang H. Detecting kernel level keyloggers through dynamic taint analysis. College of William & Mary, Department of ComputerScience, Williamsburg, VA, Tech. Rep. WM-CS-2008-05. 2008. URL: http://dx.doi.org/10.1088/1742-6596/2007/1/012005
11. Goring S.P., Rabaiotti J.R., Jones A.J. Anti keylogging measures for secure Internet login: An example of the law of unintended consequences. *Computers & Security*, V. 26. No 6. P. 421-426. URL: https://doi.org/10.1016/j.cose.2007.05.003, 2007.

https://doi.org/10.1016/j.cose.2007.05.003
12. Subhadeep Chakraborty. Detection of Keylogger using Machine Learning and Deep Learning. 2017. URL: https://doi.org/10.34740/kaggle/dsv/2625337

DETECTING THE WORK OF KEYLOGGERS IN THE OPERATING SYSTEM USING MACHINE LEARNING METHODS

H.D. Shybaiev[1], O.A. Halchynsky[2]

Ihor Sikorsky Kyiv Polytechnic Institute
37 Beresteyskyi Avenue, Kyiv, 03056, Ukraine
Emails: K233@ukr.net[1]; hleonid@gmail.com[2]

This article presents a methodology for detecting keyloggers using machine learning methods. Keyloggers are a common form of malware that captures keystrokes to steal sensitive data, posing a serious security threat to users and organizations. Traditional detection methods often rely on signature-based approaches that can be circumvented by advanced, polymorphic, or fileless keyloggers. This research uses machine learning (ML) to detect anomalous behavior and patterns indicative of keylogger activity.The authors explore various machine learning models, including decision trees, support vector machines (SVMs), random forests, and neural networks, to classify normal and malicious system behavior. Experiments were conducted to evaluate the precision, accuracy, recall, and F1 score of different ML models. The results show that ML-based methods can significantly improve the detection rate compared to traditional methods. The study also discusses potential challenges such as false positives, generalizing the model to new variants of keyloggers, and the need for ongoing training to adapt to evolving malware. The findings suggest that an ML-based detection system can offer a robust and adaptive solution to combat keyloggers, highlighting the importance of integrating artificial intelligence into cybersecurity infrastructure.

Keywords: machine learning, keylogger, artificial intelligence

DOI 10.15276/imms.v14.no3.259
УДК 005.53:165.412

Informatics and Mathematical Methods in Simulation
Vol.14 (2024), No. 3, pp. 259-267

МЕТОДИКА ПРИЙНЯТТЯ РІШЕНЬ ЗАДАЧ БАГАТОКРИТЕРІАЛЬНОГО ВИБОРУ ЗА ДОПОМОГОЮ ВДОСКОНАЛЕННЯ МЕТОДУ TOPSIS

Ю.М. Юрченко[1], Н.П. Волкова[2]

Національний університет «Одеська політехніка»
1, Шевченко пр, Одеса, 65044, Україна
Emails: yuriy090202@gmail.com[1], volkova.n.p@op.edu.ua[2]

Запропоновано методику підтримки прийняття рішень для задачі багатокритеріального вибору шляхом вдосконалення методу TOPSIS. Було проведено огляд та порівняння багатокритеріальних методів прийняття рішень, які найбільш часто застосовують для знаходження найкращого рішення в умовах багатокритеріальності, а саме методи: AHP, MAHP, ELECTRE, TOPSIS та SMART. Порівняння методів проводилось с точки зору оперативності, достовірності, стійкості та простоти реалізації. Було виявлено недоліки та переваги розглянутих методів з аналізу яких було зроблено висновок, що метод TOPSIS є простим у реалізації та характеризується високою оперативністю та достовірністю та широко застосовується для підтримки прийняття рішень в різних галузях. Основними етапами методу TOPSIS є нормалізація даних, побудова ідеального та анті-ідеального векторів, розрахунок відстаней до векторів, обчислення коефіцієнтів близькості на основі яких обирається найкраще рішення. Проте метод TOPSIS не позбавлений недоліків. Для вирішення проблеми вибору ідеального та анті-ідеального рішення, в роботі запропоновано вдосконалення методу на основі узагальненого середнього. Проведений чисельний експеримент показав, що час знаходження найкращого рішення залежить від типу узагальненого середнього. Найбільш оперативними виявилися обчислення за максимумом і мінімумом, і далі в порядку зниження оперативності: середнє арифметичне, гармонійне та геометричне. Розроблений вдосконалений метод TOPSIS є більш гнучким в налаштуванні та може бути застосований для більш широкого класу задач, що може бути корисним для застосування науковцями та практиками у багатьох галузях.

Ключові слова: багатокритеріальні методи прийняття рішень; MCDM; узагальнене середнє

Вступ. У сучасних умовах в багатьох сферах діяльності перед особою яка приймає рішення (ОПР) виникає задача знаходження найкращого рішення в задачах вибору, наприклад в таких галузях як бізнес, медицина, охорона навколишнього середовища, державна політика [1]. Знаходження рішення такої задачі часто ускладнюється зростанням кількості альтернатив та критеріїв, за якими ОПР надає оцінки альтернативам, а також неможливістю порівняти альтернативи в умовах конфлікту критеріїв, коли за деякими критеріями одна альтернатива є кращою, а за іншими гіршою. Виникає задача багатокритеріальної оптимізації [2] і постає проблема вибору методу, який би забезпечував знаходження достовірного рішення, був простим у реалізації, а також характеризувався високою оперативністю та стійкістю. Для знаходження рішення таких задач застосовують багатокритеріальні методи прийняття рішень (Multiple Criteria Decision-Making (MCDM) methods), найбільш відомими та найбільш застосовуваними є: AHP (Analytic Hierarchy Process), TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution), ELECTRE (ELimination Et Choix Traduisant la REalité), SMART (Simple Multi-Attribute Rating Technique) та інші [1, 3-5].

В [1] показано важливість методів MCDM при розв'язанні задач багатокритеріальної оптимізації в умовах складності вибору, що виникає через конфлікт критеріїв, велику кількість та неоднорідність альтернатив. Показані сильні та слабки сторони кожного

розглянутого метода MCDM та обмеження їх застосування. З проведеного в роботі аналізу з'ясовано, що вибір методу MCDM залежить від постановки задач та з урахуванням умов прийняття рішень. Так метод AHP, незважаючи на те, що він є математично обґрунтованим і дозволяє обробляти дані різних типів [5], за різних умов може виявитись неефективним з точки зору оперативності, так як потребує великих часових затрат при порівнянні важливостей альтернатив та критеріїв на різних рівнях ієрархії. Методи групи ELECTRE також виявляються неефективними з точки зору оперативності. Одним з ефективних з точки зору оперативності є метод TOPSIS [6], оскільки він дозволяє швидко провести ранжування альтернатив та обрати найкраще рішення, оцінюючи альтернативи за їх відстанню від ідеального рішення. Метод TOPSIS враховує і ідеальне рішення і анті-ідеальне рішення. Проте метод TOPSIS не позбавлений недоліків, до яких можно віднести наступні: вибір ідеального рішення і анті-ідеального рішення є суб'єктивним; критерії можуть бути нелінійно залежними; нестійкість рішення при зміні вхідних даних. Дані недоліки можуть призвести до прийняття невірних рішень, в залежності від специфіки задач та умов прийняття рішень. Таким чином, актуальним є вдосконалення методу TOPSIS для зменшення суб'єктивності у виборі ідеального і анті-ідеального рішення, а також підвищення оперативності та стійкості методу в умовах великої кількості альтернатив та критеріїв.

Аналіз літературних джерел та постановка проблеми. Наразі, в науковій літературі існує велика кількість різноманітних методів MCDM, що застосовуються для прийняття рішень в задачах з багатьма критеріями [1-6]. Для кожного з цих методів також розроблено багато варіацій, метою яких є адаптація базових методів для вирішення більш широкого класу задач.

Метод AHP (Analytic Hierarchy Process) [3] дозволяє представити складну проблему у вигляді ієрархічної структури, що робить прийняття рішень зрозумілим. Цей метод дозволяє враховувати як кількісні дані, так і експертні оцінки, у тому числі якісні, та дозволяє перевірити узгодженість суджень та уникнути суперечливих оцінок за допомогою розрахунку коефіцієнту узгодженості. Недоліками методу є низька оперативність, залежність від експертних оцінок та чутливість до невеликих змін у даних. Незважаючи на недоліки, метод AHP застосовують до широкого класу задач багатокритеріального прийняття рішень [7]. З метою покращення методу AHP було розроблено метод MAHP [7]. Метод MAHP дозволив знизити суб'єктивність в оцінках експертів, за рахунок запропонованого розрахунку узагальненого критерію і забезпечити стабільність результатів при незначному збільшенні вхідних даних.

Метод Electre [8-10] схожий за недоліками на метод AHP: низька оперативність, робота з кількісними та якісними даними, чутливість до зміни даних. Проте метод Electre також враховує конфліктні критерії.

У роботі [11] детально описано всі переваги, недоліки та обмеження методу SMART, який є оперативним з точки зору витрат часу ОПР під час прийняття рішень. Усі зазначені методи відрізняються тим, що для роботи потребують експерта чи групи експертів, що може бути проблематичним в реальних умовах.

Метод TOPSIS був представлений в роботах [6, 11-13]. На відміну від розглянутих методів MCDM, метод TOPSIS є простим в реалізації та не потребує затрат від ОПР для надання оцінок альтернатив за критеріями, що є його суттєвою перевагою, що робить його достатньо універсальним. Проте ці оцінки є тільки кількісними, що є недоліком методу TOPSIS. У задачах з великою кількістю альтернатив та критеріїв, обсяг часу, який є необхідним для пошуку оптимального рішення, швидко зростає, що знижує оперативність методу при його застосуванні під час вирішення подібних задач. Таким чином, постає задача збільшення оперативності методу.

Однією з переваг методу TOPSIS є оперативність, яка знижується зі збільшенням кількості альтернатив та критеріїв.

Окрім того, існує ряд проблем під час використання методу, наприклад, проблема визначення вагових коефіцієнтів для критеріїв для застосування методу TOPSIS в задачах, де критерії мають різний ступінь важливості, що було досліджено в роботі [11].

Метод TOPSIS має декілька вдосконалень, що допомагають розширити клас задач, до яких можна застосувати метод. Нечіткий TOPSIS [12] застосовується до задач, де дані представлені у вигляді нечітких множин або лінгвістичних змінних. Інтервальний TOPSIS [11] застосовується до задач із інтервальними даними, коли точні значення критеріїв невідомі, але визначені їхні діапазони.

Крім цього, в методі TOPSIS вибір позитивних і негативних ідеалів є суб'єктивним, тому постає задача узагальнення методу TOPSIS для зниження впливу ОПР на оцінки альтернатив за критеріями.

Було проведено порівняння методів MCDM, які є найбільш часто використаними: TOPSIS, SMART, AHP, MAHP, ELECTRE. Результати порівняння наведено в таблиці 1.

Таблиця 1.

Порівняння методів MCDM

Параметр	TOPSIS	SMART	AHP	MAHP	Electre
Взаємозв'язки критеріїв	Немає	Немає	Частково	Частково	Є
Простота реалізації	Простий	Простий	Складний	Складний	Складний
Оперативність	Висока для малих даних. Знижується лінійно зі збільшенням критеріїв і альтернатив	Висока для малих даних	Низька для великих задач (багато парних порівнянь)	Низька через обчислення з нечіткими числами	Низька через кількість обчислень для порогів і матриць
Застосування	Загальні завдання	Загальні завдання	Завдання з ієрархічною структурою	Завдання з невизначеними оцінками	Завдання з якісними оцінками
Тип критеріїв	Точні числові оцінки	Точні числові оцінки	Парні порівняння (якісні та кількісні)	Нечіткі оцінки та парні порівняння	Точні кількісні та якісні оцінки
Потребує втручання ОПР	Немає	Немає	Є	Є	Є

На основі проведеного аналізу з порівняння методів MCDM було виявлено, що методи TOPSIS та SMART є простими в реалізації та характеризуються високою оперативністю.

Мета та задачі роботи. Метою роботи є розробка методики підтримки прийняття рішень для задач багатокритеріального вибору через вдосконалення методу TOPSIS для підвищення оперативності та стійкості методу в умовах великої кількості альтернатив та критеріїв та зменшення суб'єктивності при визначенні ідеального та анті-ідеального вектору. Для досягнення поставленої мети необхідно вирішити наступні задачі: аналіз теоретичних основ методу TOPSIS; вдосконалення методу TOPSIS; експериментальне дослідження розробленої методики.

Основна частина. За результатами дослідження методів MCDM в якості опорного методу для розробки методики підтримки прийняття рішень було обрано метод TOPSIS. Згідно з методом TOPSIS, найкращим рішенням є те, яке знаходиться найближче до ідеального рішення і як найдальше від анті-ідеального рішення. Метод дозволяє вибрати

найкращу альтернативу, оцінюючи відстань до ідеального та анті-ідеального рішень. Однією з переваг методу TOPSIS є оперативність, яка знижується зі збільшенням кількості альтернатив та критеріїв.

Окрім того, існує ряд проблем під час використання методу, наприклад, проблема визначення вагових коефіцієнтів для критеріїв для застосування методу TOPSIS в задачах, де критерії мають різний ступінь важливості, що було досліджено в роботі [11].

Метод TOPSIS має декілька вдосконалень, які допомагають розширити клас задач, до яких можна застосувати метод. Нечіткий TOPSIS [12] застосовується до задач, де дані представлені у вигляді нечітких множин або лінгвістичних змінних. Інтервальний TOPSIS [11] застосовується до задач із інтервальними даними, коли точні значення критеріїв невідомі, але визначені їхні діапазони.

Розглянемо алгоритм методу TOPSIS, який вперше було запропоновано в [6, 14].

1. Створення матриці показників X з m альтернатив та n критеріїв таким чином, що x_{ij} це оцінка i-ої альтернативи ($i = 1,...,m$) та j-ого критерію, ($j = 1,...,n$).
2. Нормалізація матриці показників X за формулою:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}},\ i = 1,...,m,\ j = 1,...,n.$$

Таким чином отримуємо нормалізовану матрицю R.

3. Отримання ідеального та анті-ідеального рішення.

Ідеальне рішення це вектор A^+, який визначається наступним чином:

$A_j^+ = \max_i(r_{ij})$, $j = 1,...,n$, якщо j-ий критерій максимізується,

$A_j^+ = \min_i(r_{ij})$, $j = 1,...,n$, якщо j-ий критерій мінімізується.

Анті-ідеальний рішення це вектор A^-:

$A_j^- = \min_i(r_{ij})$, $j = 1,...,n$, якщо j-ий критерій максимізується,

$A_j^- = \max_i(r_{ij})$, $j = 1,...,n$, якщо j-ий критерій мінімізується.

4. Обчислення евклідової відстані від рішень до ідеального та анті-ідеального рішення:

$$D_i^+ = \sqrt{\sum_{j=1}^{n}(r_{ij} - A_j^+)^2},\ i = 1,...,m,$$

$$D_i^- = \sqrt{\sum_{j=1}^{n}(r_{ij} - A_j^-)^2},\ i = 1,...,m,$$

5. Обчислення відносної близькості до ідеального рішення для кожної i-тої альтернативи за формулою

$$C_i = \frac{D_i^-}{D_i^+ + D_i^-}.$$

6. Вибір найкращого варіанта, як варіанта з найбільшим показником відносної близькості до ідеального рішення.

На етапі 3, розглянутого алгоритму, під час отримання ідеального та анті-ідеального рішень, ОПР обирає напрям оптимальності критеріїв (максимізація або минимизация). Даний етап підкреслює суб'єктивність методу, тобто залежно від знань та опиту ОПР, відбувається обрання напряму оптимальності. В даній роботі запропоновано вдосконалення методу TOPSIS задля розширення класу задач, до яких може бути застосований метод, збільшення оперативності та зменшення суб'єктивності методу. Таким чином, на етапі 3 базового методу пропонується під час отримання ідеального та анті-ідеального рішення спиратись на використання узагальненого середнього, яке

також іноді називають квазі-арифметичним середнім чи f - середнім [1,4]. Нехай $I \subset R$, інтервал дійсних чисел та $f: I \to R$ це неперервна та ін'єктивна функція, тоді для чисел $x_1,...,x_n \in I$ узагальнене середнє [4] визначається формулою:

$$GenM_f(x_1,...,x_n) = f^{-1}(\frac{1}{n}\sum_{i=1}^{n} f(x_i)).$$

У якості середнього може бути обране будь-яке число з інтервалу від найменшого числа до найбільшого числа вибірки, включаючи найменше та найбільше числа, середнє арифметичне, середнє геометричне, середнє гармонійне і подібні. Також, у якості середніх можуть бути використані статистичні показники: мода та медіана [10].

Таким чином отримуємо алгоритм для вдосконаленого методу TOPSIS.

Вдосконалений метод TOPSIS реалізується за наступним алгоритмом:

1. Створення матриці показників X з m альтернатив та n критеріїв таким чином, що x_{ij} це оцінка i-ої альтернативи та j-ого критерію.
2. Нормалізація матриці показників X за формулою:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{m} x_{ij}^2}},\ i = 1,...,m,\ j = 1,...,n.$$

Отримуємо нормалізовану матрицю R.

3. Отримання ідеального та анті-ідеального рішення.

Ідеальний вектор A^+:

$$A_j^+ = GenM_f(r_{1j},...,r_{mj}),\ j = 1,...,n.$$

Анті-ідеальний вектор A^-:

$$A_j^- = GenM_g(r_{1j},...,r_{mj}),\ j = 1,...,n.$$

4. Обчислення евклідової відстані від рішень до ідеального та анті-ідеального рішення:

$$D_i^+ = \sqrt{\sum_{j=1}^{n}(r_{ij} - A_j^+)^2},\ i = 1,...,m,$$

$$D_i^- = \sqrt{\sum_{j=1}^{n}(r_{ij} - A_j^-)^2},\ i = 1,...,m,$$

5. Обчислення відносної близькості до ідеального рішення для кожної i-тої альтернативи за формулою

$$C_i = \frac{D_i^-}{D_i^+ + D_i^-}.$$

6. Вибір найкращого варіанта, як варіант з найбільшим показником відносної близькості до ідеального рішення.

Експериментальне дослідження розробленої методики. Для дослідження оперативності методу було розглянуто базовий метод TOPSIS та запропоноване в работи вдосконалення методу TOPSIS. В ході дослідження було проведено експеримент, а саме: було досліджено яким чином на оперативність методу впливає вибір узагальненого середнього. Було розглянуто три випадки для отримання ідеального та анті-ідеального рішення розглядали три випадки:

1. В якості ідеального рішення обирали A^+:

$A_j^+ = GenM_f(r_{1j},...,r_{mj}),\ j = 1,...,n$, де $GenM_f(r_{1j},...,r_{mj})$ це максимум.

Анті-ідеальний вектор A^-:

$A_j^- = GenM_g(r_{1j},...,r_{mj}),\ j = 1,...,n$, де $GenM_f(r_{1j},...,r_{mj})$ це мінімум.

2. В якості ідеального рішення обирали A^+:
   $A_j^+ = GenM_f(r_{1j},...,r_{mj})$, $j = 1,...,n$, де $GenM_f(r_{1j},...,r_{mj})$ це середнє арифметичне.
   Анті-ідеальний вектор A^-:
   $A_j^- = GenM_g(r_{1j},...,r_{mj})$, $j = 1,...,n$, де $GenM_f(r_{1j},...,r_{mj})$ це середнє геометричне.
3. В якості ідеального рішення обирали A^+:
   $A_j^+ = GenM_f(r_{1j},...,r_{mj})$, $j = 1,...,n$, де $GenM_f(r_{1j},...,r_{mj})$ це середнє арифметичне.
   Анті-ідеальний вектор A^-:
   $A_j^- = GenM_g(r_{1j},...,r_{mj})$, $j = 1,...,n$, де $GenM_f(r_{1j},...,r_{mj})$ це середнє гармонійне.

В ході дослідження було виконано програмна реалізація базового методу та запропонованого вдосконаленого методу TOPSIS та замірено час, за який відповідні методи знаходять найкраще рішення. Для дослідження роботи методів було згенеровано матрицю показників з подальшим знаходженням найкращого рішення базовим методом та його вдосконаленням. Результати представлені на рисунку 1.

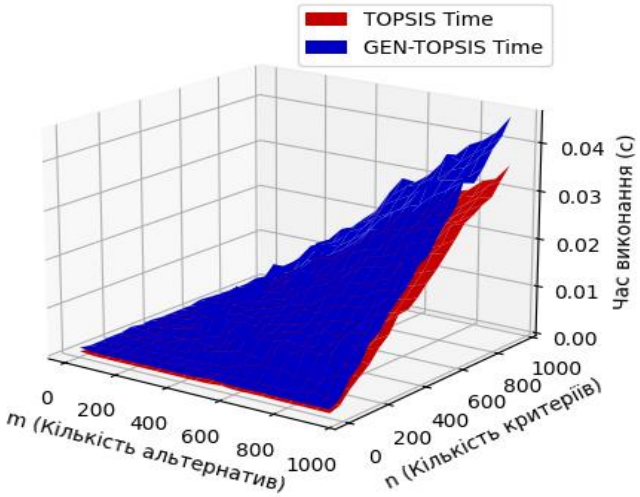


Рис. 1. Порівняння часу пошуку рішення базовим методом TOPSIS та методом TOPSIS з середнім арифметичним та середнім гармонійним

В результаті дослідження отримали, що найшвидше відбувається обчислення максимуму та мінімуму, а далі за зростанням часу обробки: середнє арифметичне, середнє гармонійне, середнє геометричне

Також було проведено дослідження точності отриманого рішення. Для порівняння отриманих рішень в якості контрольного приклада було розглянуто задача з [20-21]. Необхідно обрати найкраще місце для побудови аеропорту, в якості альтернатив розглядалось чотири площадки *A*, *B*, *C*, *D*, в якості критеріїв розглядалось: вартість будови (C_1), відстань від міста (C_2), мінімальний шумовий вплив (C_3).

Вхідні дані і результати обчислень представлені в таблиці 2.

Таблиця 2.

Вхідні дані і результати обчислень

	Вхідні дані			Отримані оцінки альтернатив		
	C_1	C_2	C_3	МАНР	TOPSIS	GEN-TOPSIS
A	180	70	10	0,005	0,4632	0,4903
B	170	40	15	0,076	0,3514	0,4623
C	160	55	20	**0,57**	**0,7093**	**0,5523**
D	150	50	25	0,51	0,5	0,5246

В результаті застосування методів MAHP, TOPSIS і запропонованого вдосконалення методу TOPSIS найкращою виявилася альтернатива, яка відповідає площадці *D* для побудови аеропорту.

Крім того, було досліджено стійкість рішень базового та вдосконаленого методу TOPSIS. В матрицю показників додавали рядок, що дублює існуючий рядок (альтернатива дублює іншу альтернативу). Було встановлено, що найкраще рішення не змінюється незалежно від таких маніпуляцій з матрицею показників, що свідчить про стійкість методів.

Аналогічне дослідження було проведено стосовно критеріїв. У матрицю показників додавали стовпчик, що дублює наявний стовпчик (критерій дублює інший критерій). Експеримент показав, що за всіма методами найкраще рішення не змінюється незалежно від таких маніпуляцій із матрицею показників. Це свідчить про стійкість методів до додавання стовпчика, що дублює інший критерій.

Висновки. Розроблено методику підтримки прийняття рішень для задач багатокритеріального вибору через вдосконалення методу TOPSIS для підвищення оперативності та стійкості методу в умовах великої кількості альтернатив та критеріїв та зменшення суб'єктивності при визначенні ідеального та анті-ідеального рішення.

Було проведено порівняльний аналіз методів багатокритеріального прийняття рішень та виявлено переваги та недоліки методів. Проведений аналіз показав, що метод TOPSIS є простим в реалізації та характеризуються високою оперативністю, тому його було обрано у якості базового для подальшої роботи.

На основі узагальненого середнього було реалізовано вдосконалений метод TOPSIS, який дає можливість застосувати метод до більш широкого класу задач.

У процесі чисельного експерименту було досліджено вплив вибору узагальненого середнього на час знаходження найкращого рішення методом TOPSIS та його вдосконаленням. Отримали, що найшвидшими є обчислення максимуму та мінімуму. Наступними за швидкістю є середнє арифметичне, середнє гармонійне, середнє геометричне.

Для перевірки точності результатів методів MAHP, TOPSIS і розробленого узагальнення TOPSIS було розглянуто задачу вибору найкращої площадки для будівництва аеропорту [19]. Отримали, що узагальнений метод TOPSIS надав таке ж саме рішення, як і розглянуті відомі методи MCDM.

Дослідження показало високу стійкість базового та вдосконаленого методу TOPSIS до маніпуляцій із матрицею показників. Додавання рядка, що дублює існуючу альтернативу, не вплинуло на результат, що свідчить про стабільність методів. Аналогічно, додавання стовпчика, який дублює наявний критерій, не призвело до змін у результатах, що підтверджує стійкість методів до таких маніпуляцій.

Отже, можна зробити висновок, що класичний метод TOPSIS залишається ефективним інструментом для багатокритеріального аналізу, але вдосконалий метод TOPSIS надав можливість зменшити суб'єктивність при визначенні ідеального та анті-ідеального вектору, при цьому метод забезпечує високу точність та оперативність..

Перспективи подальших досліджень можуть бути зосереджені на вдосконаленні та комбінуванні різних методів для підвищення точності та оперативності методу.

Список літератури

1. Sahoo S.K., Goswami S.S. A Comprehensive Review of Multiple Criteria Decision-Making (MCDM) Methods: Advancements, Applications, and Future Directions. *Decision Making Advances.* 2023. V.1(1). P. 25–48. URL: https://doi.org/10.31181/dma1120237
2. Ambroziak T., Malesa A., Kostrzewski M. Analysis of multicriteria transportation problem connected to minimization of means of transport number applied in a selected example. *WUT J. Transp.* Eng, 2018. V.123. P.5-20. DOI: 10.5604/01.3001.0013.7349.

3. Akmaludin A. Comparison of Selection for Employee Position Recommended MCDM-AHP, SMART and MAUT Method. *Sinkron: jurnal dan penelitian teknik informatika.* 2023. V. 7. No. 2. P. 603-616.
4. Dhurkari R. K. MCDM methods: Practical difficulties and future directions for improvement. *RAIRO-Operations Research*. 2022. V.56(4). P. 2221-2233. URL: https://doi.org/10.1051/ro/2022060.
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; Vol. 5 No. 4. P. 275–286. DOI: https://doi.org/10.15276/hait.05.2022.20.
6. Bullen P. S. Handbook of means and their inequalities. *Springer Science & Business Media*. 2013.
7. Cardozo F. Application of Monte Carlo Analytic Hierarchy Process (MAHP) in Underground Mining Access Selection. *Mining.* 2023. V. 3. No. 4. P. 773-785.
8. De Carvalho M. Mean, what do you Mean? *The American Statistician.* 2016. V. 70. No. 3. P. 270-274.
9. Jahanshahloo G. R., Lotfi F. H., Izadikhah M. An algorithmic method to extend TOPSIS for decision-making problems with interval data. *Applied mathematics and computation.* 2006. V. 175. No. 2. P. 1375-1384.
10. Hwang C.L,, Lai Y.J., Liu T.Y. A new approach for multiple objective decision making. *Computers & operations research.* 1993. V. 20. No. 8. P. 889-899.
11. Figueira J.R. ELECTRE methods: Main features and recent developments. Handbook of multicriteria analysis. 2010. P.51-89.
12. Figueira J.R., Mousseau V., Roy B. ELECTRE methods. *Multiple criteria decision analysis: State of the art surveys.* 2016. P.155-185.
13. Olson D.L. Comparison of weights in TOPSIS models. *Mathematical and Computer Modelling*. 2004. V. 40. No. 7-8. P. 721-727.
14. Von Hippel P.T. Mean, median, and skew: Correcting a textbook rule. *Journal of statistics Education.* 2005.V. 13. No. 2.
15. Podvezko V.. Application of AHP technique. *Journal of Business Economics and management*. 2009. V. 2. P. 181-189.
16. Sorin N., Dzitac S, Dzitac I. Fuzzy TOPSIS: a general view. *Procedia computer science.* 2016. V. 91. P. 823-831.
17. Taherdoost H., Mohebi A. Using SMART Method for Multi-Criteria Decision Making: Applications, Advantages and Limitations. *Archives of Advanced Engineering Science.* 2024. P. 1-10.
18. Yoon K. P., Hwang C.L. Multiple attribute decision making: an introduction. Sage publications. 1995.
19. Kwangsun Y. A reconciliation among discrete compromise solutions. *Journal of the Operational Research Society*. 1987. V. 38. No. 3. P. 277-286.
20. Power D. J. Web-based and model-driven decision support systems: concepts and issues. *Americas Conference on Information Systems*, California. 2000.
21. Kozina Y., Volkova N., Horpenko D., Mobile Application for Decision Support in Multi-Criteria Problems. *IEEE Second International Conference on Data Stream Mining & Processing (DSMP)*, Lviv, Ukraine. 2018. P. 56-59, DOI: 10.1109/DSMP.2018.8478499.

DECISION-MAKING METHODOLOGY FOR MULTI-CRITERIA SELECTION PROBLEMS THROUGH ENHANCEMENT OF THE TOPSIS

Y.M. Yurchenko[1], N.P.Volkova[2]

National Odesa Polytechnic University
1, Shevchenko Ave., Odesa, 65044, Ukraine
Emails: yuriy090202@gmail.com[1], volkova.n.p@op.edu.ua[2]

This paper proposes a decision-making methodology for solving multi-criteria selection problems by enhancing the TOPSIS method. A review and comparison of multi-criteria decision-making (MCDM) methods commonly used for identifying the best solution in multi-criteria environments were conducted. The analyzed methods include AHP, MAHP, ELECTRE, TOPSIS, and SMART. The methods were compared in terms of operational efficiency, reliability, stability, and ease of implementation. The analysis revealed the advantages and disadvantages of these methods, leading to the conclusion that TOPSIS is simple to implement, exhibits high efficiency and reliability, and is widely applied across various fields for decision support. The key steps of the TOPSIS method include data normalization, constructing ideal and anti-ideal vectors, calculating distances to these vectors, and computing proximity coefficients to determine the optimal solution. However, the TOPSIS method has limitations. To address the challenge of determining the ideal and anti-ideal solutions, this paper proposes an enhancement of the method based on generalized means. A numerical experiment demonstrated that the computation time for determining the best solution depends on the type of generalized mean. The most efficient calculations were observed with the maximum and minimum means, followed by arithmetic, harmonic, and geometric means in descending order of efficiency. The developed enhanced TOPSIS method is more flexible in its configuration and can be applied to a broader range of problems, making it valuable for both researchers and practitioners in various fields.
Keywords: multi-criteria decision-making methods; MCDM, ELECTRE, TOPSIS, SMART, AHP

ІНФОРМАТИКА ТА МАТЕМАТИЧНІ МЕТОДИ В МОДЕЛЮВАННІ

Том 14, номер 3, 2024. Одеса – 268 с., іл.

INFORMATICS AND MATHEMATICAL METHODS IN SIMULATION

Volume 14, No. 3, 2024. Odesa – 268 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 №3 від 30.10.2024р.)

Адреса редакції: Національний університет «Одеська політехніка»

1, Шевченка проспект, Одеса 65044 Україна

Web: www.immm.op.edu.ua (immm.opu.ua)

Email: immm.ukraine@gmail.com