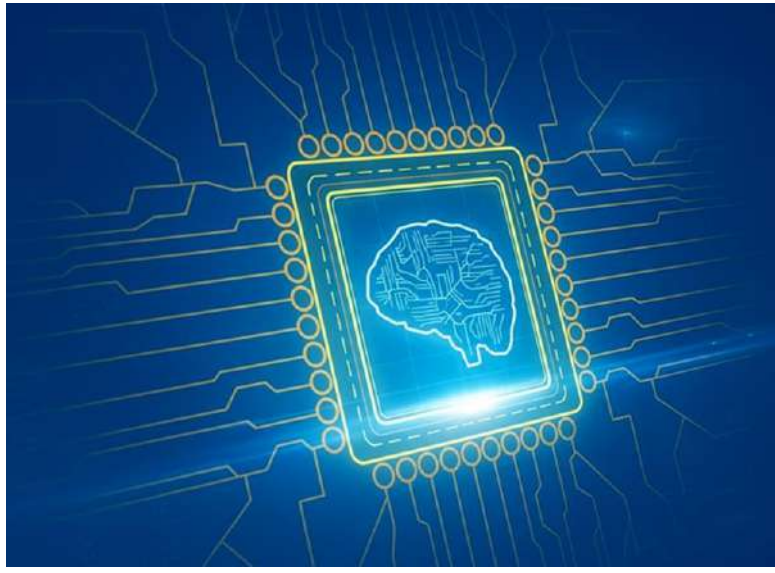
Національний університет «Одеська політехніка»



СИСТЕМИ КОНТРОЛЮ ІНФОРМАЦІЇ ТА ІНТЕЛЕКТУАЛЬНІ ТЕХНОЛОГІЇ. ДОСЯГНЕННЯ ТА ЗАСТОСУВАННЯ

МОНОГРАФІЯ

Львів-Торунь
Liha-Pres
2025

УДК 004:37:001:62
Р64

Авторський колектив:
*Н.Аксак, С.Антощук**.** О.Арсірій, К.Бєляєв, І.Бобок, А.Вичужанін, В.Вичужанін, О.Вовк, А.Глушко, Ж.Дейнеко, Д.Джонс, Д.Жужнєв, О.Залуцька, О.Іванов**,** Ю.Кавац, А.Кобозєва, О.Ковров, О.Коршун, В.Краснобаєв, М.Кушнарьов, Д.Кошутіна**,** А.Лабіб, С.Литвиненко, О.Лисенко, Ю.Маріяш, О.Мазурець, М.Мендієлєва, М.Мизюра, С.Науменко, Ю.Нетребін,О.Овчарук, А.Олійник, С.Орлов, Т.Отрадська, К.Паничук, І.Петров, С.Положаєнко, С.Пономаренко, І.Розломій, М.Рудніченко, А.Савєльєв, В.Саваневич,К.Сергєєва, С.Смик, О.Степанець, О.Степаненко, І.Табакова, О.Татарин, Ю.Тесленко, О.Тачиніна, Т.Трунова, Є.Федорченко, О.Фомін, А.Фролов, С.Хламов, Н.Цудецка-Пуріна, Д.Чумичов, Ю.Шеліхов, Д.Шведов, Н.Шибаєва, І.Шпінарева, Г.Щербакова**,** А.Янко, А.Ярмілко.*

Рекомендовано до друку вченою порадою
Національного університету «Одеська політехніка»
(протокол №2 від 18.09.2025 р.)

Рецензенти:
Професор **Є Чженмао**, Південний університет (США);
Професор **С. Овермайер**, Університет Південного Нью-Гемпширу (США)

Р64 **Системи** контролю інформації та інтелектуальні технології. Досягнення та застосування : монографія / під наук. ред. проф. В.Вичужаніна. – Львів-Торунь : Liha-Pres, 2025 – 402 с. Укр., англ. мовами.
ISBN 978-966-397-538-2
DOI https://doi.org/10.36059/978-966-397-538-2

У монографії відображені результати наукових досліджень у галузі інформаційних, інтелектуальних систем та аналізу даних, моделювання. Матеріали монографії будуть корисними для аспірантів, магістрантів, викладачів вищих навчальних закладів, що спеціалізуються на галузі ІТ-технологій.

УДК 004:37:001:62

ISBN 978-966-397-538-2

CONTENTS

PREFACE

The present monograph compiles materials derived from articles submitted to the Program Committee and presented at the XIII International Scientific and Practical Conference *"Information Systems and Technology Management (ICST-Odessa-2025)."*

This collective volume reflects the outcomes of scientific investigations in the domains of information systems and technologies, intelligent systems, data analysis, modeling, and software engineering.

The monograph is organized in the form of research article-based chapters, grouped according to thematic sections that comprehensively represent the scope of studies conducted in the following areas: Information Control Systems, Intelligent Systems and Data Analysis, and Modeling and Software Engineering.

Special emphasis is placed on addressing issues such as: Implementation of Cryptographic Transformations Based on the Residue Number System to Enhance Digital Security; Development of an Intelligent Hybrid System for Big Data Volumes Health Harm Risks; Integrated Modeling of Reliability and Maintenance of Ship Power Plant Equipment Considering Degradation and Operational Conditions Analysis.

Considerable attention is also devoted to:

Intelligent Hybrid System Design for Big Data Health Risk Analysis, Modeling of Nonlinear Dynamics Using the Support Model Method, and Performance Metrics Evaluation for Load Testing Tools.

The results presented provide readers with valuable insights required for a deeper understanding of the challenges and solutions in the field of information systems and technologies.

All articles included in this monograph correspond to the original versions submitted by the authors. Full responsibility for the content of individual contributions rests solely with their respective authors.

The materials of the monograph will serve as a useful resource for postgraduate and master's students, as well as academic faculty engaged in research and education in the field of information systems and technologies.

The preparation of this collective work involved the contribution of scholars, including 14 Doctors of Science, 26 Candidates of Science, and 21 PhD applicants.

Section 1. Information control systems

UDC 004.056.5:519.725
DOI https://doi.org/10.36059/978-966-397-538-2-1

IMPLEMENTATION OF CRYPTOGRAPHIC TRANSFORMATIONS BASED ON THE RESIDUE NUMBER SYSTEM TO ENHANCE DIGITAL SECURITY

Ph.D. A. Yanko[0000-0003-2876-9316], **Dr.Sci. V. Krasnobayev**[0000-0001-5192-9918], **Ph.D. A. Hlushko**[0000-0002-4086-1513], **M. Myziura**[0009-0009-9301-2054]
National University «Yuri Kondratyuk Poltava Polytechnic», Ukraine
EMAIL:al9_yanko@ukr.net

Abstract. *The paper addresses the critical issues of strengthening business security during digital transformation. The authors demonstrate that the expansion of digitalization processes necessitates a reevaluation of the economic security concept. It is substantiated that in order to strengthen business resilience to risks and threats to digital security, it is necessary to implement a number of measures aimed at protecting the confidentiality, integrity and availability of information. A study of cyber threats to national economic entities and citizens was conducted, including with the use of artificial intelligence tools. This made it possible to identify a priority area of data protection – improving the RSA cryptosystem. This research details the development of efficient information processing strategies for reducing the latency of RSA cryptographic functions. To accelerate RSA cryptographic transformations, this study introduces methods for high-speed information processing. The core of suggested method involves the realization of a cyclic shift mechanism utilizing modular arithmetic, entirely implemented by the residue number system (RNS). The application of RNS demonstrates its effectiveness in structuring the process of implementing modular integer arithmetic operations for accelerating public-key cryptographic transformations.*

Keywords: *binary remainder representation technique, cryptographic information protection, cryptography algorithm, cyclic shift arrays, digital transformation, high-speed crypto accelerators, modular arithmetic codes, residue number system, ring shift mechanism.*

Problem Statement. In today's world of rapid digitalization of society and globalization of economic processes, digital security is a fundamental prerequisite for the stable functioning of all sectors – from public administration and finance to industrial production and critical infrastructure. The constant growth in the volume of data processed and the introduction of Internet of Things (IoT), artificial intelligence (AI), and cloud computing technologies create not only new opportunities for the development of business and socio-economic systems, but also

new vectors of vulnerability [1]. In these conditions, the risks of unauthorized access, distortion, or destruction of information become a critical factor that can lead to large-scale economic losses, disruption of business processes, and loss of trust from consumers and partners.

Ensuring digital security goes far beyond technical issues and has become a complex interdisciplinary task that combines scientific research in the fields of cryptography, mathematical modeling, cybernetics, and risk management [2]. At the same time, in practical terms, it involves the development and implementation of innovative solutions capable of countering increasingly complex cyber threats, in particular by adapting modern cryptographic transformations and optimizing computing processes.

In this context, research aimed at improving the effectiveness of information protection systems by using residual number systems (RNS) as a tool for increasing the resistance of cryptographic algorithms to attacks and reducing computational costs is of particular importance [3]. Such research responds both to significant scientific challenges – the search for new mathematical models and encryption methods – and to the practical tasks of digital transformation of the economy, the formation of a secure cyberspace, and the reduction of threats to strategically important objects.

Analysis of Previous Studies. The digital transformation, which now encompasses virtually all sectors of the economy and society, is bringing about a fundamental change in approaches to data processing, storage, and transmission. The introduction of intelligent information systems, cloud services, IoT, AI, and blockchain technologies opens up new opportunities for improving the efficiency of business processes, optimizing management decisions, and increasing the competitiveness of organizations [4]. However, along with these advantages, the range of cyber threats is also growing, putting the issue of ensuring reliable protection of digital assets on the agenda.

The issue of digital security, particularly through cryptographic mechanisms, is key to ensuring trust in transformed digital processes. Research shows that the degree of integration of information security into business processes directly affects the stability and performance of organizations. One example is the analysis of the impact of digital transformation on business security and recommendations for strengthening its cyber resilience [5].

In scientific discourse, digital security is being rethought as a growing component of digital transformation that requires comprehensive solutions – from technological to managerial [6]. In addition, the growth of computing capabilities, especially with the approach of the era of quantum computing, is forcing a rethinking of traditional cryptographic approaches. In this context, post-quantum cryptography (PQC) is becoming critically important for protecting data today – to avoid “collect now, decrypt later” attacks – and to minimize the risks of future code breakthroughs [7].

From a scientific point of view, digital security challenges require innovative scientific solutions, such as combining cryptographic transformations with

blockchain technologies that ensure data resilience, transparency, and authenticity in decentralized systems [8].

Contemporary public-key cryptosystems widely utilize algebraic curve-based transformations, including elliptic curves (EC), hyperelliptic curves (HEC) [9, 10], Picard curves (PC), and superelliptic curves (SEC) [11], in addition to the traditional RSA scheme. The practical implementation of these systems relies on various scalar multiplication algorithms, such as the Kantor divisor addition method, the Koblitz method, arithmetic transformation techniques for HEC Jacobian divisors, weighted divisor addition methods, the Karatsuba algorithm for modular multiplication, polynomial function field reduction, and approaches based on the Chinese Remainder Theorem. However, many of these methods do not fully meet the high efficiency requirements of modern cryptographic applications. In contrast, recent studies [12, 13] demonstrate that modular arithmetic codes, particularly those based on the Residue Number System (RNS), offer significant advantages in accelerating digital information processing, including tasks such as digital filtering, Fast Fourier Transform (FFT), and Discrete Fourier Transform (DFT) computations.

This context underscores the critical importance and timeliness of developing novel approaches to improve the performance of cryptographic transformations, particularly RSA, through the utilization of RNS. The RSA system, initially proposed in 1977, remains the most prevalent public-key cryptosystem in use today [14, 15, 16].

The primary goal of the studies documented in [17, 18] is to formulate a method for rapid execution of public-key cryptographic transformations and to design a structural model for the operating unit (OU) of a high-speed cryptographic coprocessor, leveraging the capabilities of RNS. The research [19] presents a modified stream cipher cryptographic processor equipped with specialized instructions based on the VLIW architecture. The proposed system utilizes a distributed (clustered) memory structure and is designed for efficient execution of stream cipher operations. Such architecture ensures high performance in processing stream cryptographic algorithms.

Research in [20] investigates the impact of fundamental properties of the modular number system (MNS), such as remainder independence, equality, and the presence of low-order digits, on the architecture and operational principles of crypto accelerator systems utilizing MNS. Specifically, it highlights that the presence of low-order digits in modular representations allows for a wide array of system and technical design choices when implementing integer modular arithmetic operations.

There are four primary methodologies for performing arithmetic operations within RNS: the summation method (utilizing low-order bits of binary adders modulo RNS); the table lookup method (employing read-only memory); the direct logical method, which involves defining and implementing modular operations at the switching function level to generate result values (systolic arrays, programmable logic matrices, and programmable logic devices (PLDs) are suitable hardware platforms for this approach) [21]; and the ring shift mechanism (RSM), which leverages cyclic shift arrays (CSA).

A significant and highly advantageous characteristic of RNS, when based on modular multiplication algorithms, is the absence of inter-remainder carry propagation during cryptographic transformations within the cryptographic coprocessors employing the ring shift mechanism (RSM). While intra-remainder carries exist between binary digits within each modulus p_n, the elimination of carry propagation between remainders during modular operations [22] presents a key benefit.

Unresolved Issues. A prevailing direction in cryptographic information processing research focuses on extending key lengths. However, this approach inherently leads to a reduction in the processing speed of public-key cryptosystems. This slowdown is particularly problematic when implementing EC-based cryptosystems in resource-constrained environments, such as specialized systems and devices where the use of high-performance, multi-precision computers is not feasible. Consequently, there is a pressing need for the development of techniques that enhance the efficiency, reliability, and security of cryptographic transformations.

Objective of the Article. The objective of this article is to investigate methods for enhancing the performance and security of public-key cryptographic systems. The work focuses on the RSA cryptosystem, which remains a fundamental mechanism for data protection. A primary goal is to address the issue of computational latency that arises from the need for increasingly long key lengths, which in turn diminishes the processing speed of cryptographic transformations.

To achieve this, the article aims to develop and substantiate a novel approach based on the RNS. The proposed method involves the implementation of a cyclic shift mechanism utilizing modular arithmetic, entirely within the RNS framework. This approach is designed to accelerate core cryptographic operations, such as modular multiplication and exponentiation, which are the most computationally intensive components of the RSA algorithm.

Ultimately, the research seeks to demonstrate that the application of RNS can provide a more efficient and reliable solution for high-speed crypto accelerators. The findings are intended to offer a practical and academically sound contribution to the field of information security, paving the way for the development of more robust and responsive digital security infrastructure.

Main Content. In a positional number system (PNS), arithmetic operations necessitate sequential digit processing due to operation-specific rules, preventing completion until all intermediate results, reflecting inter-digit dependencies, are determined. Consequently, PNS, prevalent in contemporary high-speed crypto accelerators (HSCA), suffers from inherent inter-digit connections that complicate arithmetic operation implementation, demand complex hardware, compromise computational reliability, and limit cryptographic transformation speed [23]. Therefore, a number system devoid of inter-digit dependencies is desirable. The RNS offers this advantage, possessing a unique property: the independence of remainders based on the chosen base [24]. This independence facilitates the

development of novel machine arithmetic and fundamentally new HSCA architectures, thereby expanding the applicability of machine arithmetic. Numerous studies [25, 26, 27] suggest that adopting non-traditional data representation and parallel processing in digital systems enhances computational efficiency, particularly in modular arithmetic, which exhibit maximum internal parallelism during information processing. RNS falls within this category.

The primary bottleneck in high-speed digital systems, including crypto accelerators, is the "carry propagation problem" inherent in positional number systems. In PNS, the calculation of each digit in an arithmetic operation, such as addition or multiplication, depends on the carry from the preceding position. This sequential dependency creates a critical path that directly limits the maximum operating frequency and overall processing speed. This issue is particularly pronounced when dealing with the large bit-length numbers required for modern, secure cryptographic algorithms like RSA. As key sizes increase, the carry propagation delay scales almost linearly, creating a fundamental barrier to achieving real-time performance in resource-constrained environments. This fundamental limitation of PNS makes alternative number systems, such as the RNS, highly attractive for high-performance applications where parallel processing can be leveraged.

To further illustrate the effectiveness of the proposed approach, let's consider a simple example of adding two large numbers in a traditional binary system (PNS) versus in the Residue Number System (RNS). Suppose we need to compute the sum of two 64-bit numbers, for example, *Y* and *U*.

1. The Traditional Approach (PNS): In the binary system, addition is performed sequentially, bit by bit. Each subsequent bit's value depends on the carry from the previous position. For a 64-bit number, the result in the 64th bit cannot be computed until all intermediate carries from the 1st to the 63rd bit are known. This dependency on carry propagation is the primary factor limiting the computation speed. The addition time is the sum of the time required to process each bit, which can be notionally represented as $T_{PNS} = 64 \times T_{gate}$, where T_{gate} – is the switching time of a single logic gate.

2. The Proposed Approach (RNS): In contrast, let's use the RNS with a set of relatively prime moduli. For a 64-bit number, this could be a set of 8-bit moduli. Adding the numbers Y and U in RNS is performed in parallel for each modulus, with no carry propagation between them. The calculations are performed as follows $\left((y_1 + u_1) (\mathrm{mod}\, m_1), (y_2 + u_2) (\mathrm{mod}\, m_2), ..., (y_n + u_n) (\mathrm{mod}\, m_n)\right)$. Each of these calculations is executed independently in a separate arithmetic processing unit (APU). The proposed ring shift mechanism (RSM) allows each remainder to be computed in a time that depends only on the size of its respective modulus, not on the total length of the number.

This example demonstrates that while PNS speed is limited by the number's length, RNS calculations occur in parallel. This parallelism eliminates the delay

caused by carry propagation, achieving a significant speedup that is critical for high-performance cryptographic applications.

Several factors support the effective utilization of RNS in HSCA: HSCA, like RNS, processes only integer data; HSCA primarily performs modular arithmetic operations; RNS excels in executing modular multiplication and squaring operations, which constitute over 95% of RSA cryptosystem operations, particularly in modulus p_n; as the word length (W) of HSCA processors increases, a trend in modern RSA system development, RNS application efficiency improves; the widespread use of CSA in HSCA for RSA transformations; the limitations of PNS in achieving significant HSCA efficiency and reliability gains; and promising preliminary results demonstrating RNS's effectiveness in enhancing real-time HSCA performance and reliability [28].

Research presented in [29] elucidates the operational principle of integer residual arithmetic, specifically the ring shift mechanism (RSM). This mechanism is distinguished by its ability to determine the result of arithmetic operations, such as $(y_n \pm u_n) \bmod p_n$, for any modulus p_n within the RNS base set $\{p_n\}\ (n=\overline{1,\ q})$, without necessitating the computation of partial sums S_n or carry values C_n from binary adders in PNS. Instead, the result is derived through cyclic shifts of a predefined digital structure. This approach is grounded in Cayley's theorem, which establishes an isomorphism between the elements of a finite abelian group and those of a permutation group [30].

From Cayley's theorem, it can be inferred that the action of abelian group elements on the group of integers is homomorphic [36]. This property enables the organization of arithmetic operation result determination in RNS through the application of RSM. Thus, an operand in RNS is represented as a set of q remainders $\{y_n\}\ (n=\overline{1,\ q})$, obtained by successively dividing an initial number Y by n pairwise prime numbers $\{p_n\}$. In this context, the collection of remainders $\{y_n\}$ directly corresponds to the sum of q simple Galois fields $GF(p_n)$ [32].

An algebraic system (A) consists of a plural (P) and a set of operations (F) defined on this set. This system is denoted as $A=(P,F)$, where P is a non-empty plural of integers (Z); F is a set of binary operations (specifically, in RNS implementation, the operations executed in a single clock cycle are the arithmetic operations: $+,-,\times$) [33]. That is, F is the set of operations addition (+), subtraction ($-$), multiplication ($\times$) for any $y_n, u_n \in Z$, y_n+u_n, y_n-u_n, $y_n \times u_n$ also belong to Z. It is important that the operations be closed on the plural P, that is, the result of the operation on elements from P also belongs to P. Therefore, it is very important that the range of representation of numbers in the

$D = \prod_{n=1}^{q} p_n$

MSN overlaps the set P, that is, that the elements a and b themselves, and the result of the arithmetic operations $+, -, \times$, lie in this range. In cryptography, where information security is a key aspect, the use of large numbers becomes necessary to ensure the reliability and robustness of cryptographic systems. The larger the number of bits, the more difficult it is to break a cryptographic algorithm, as the number of possible combinations grows exponentially. Asymmetric cryptography algorithms, such as RSA, DSA, and ECC, are based on the use of large prime numbers to generate cryptographic keys [34]. The key operations in these algorithms are modular multiplication and exponentiation, which are performed on large-bit numbers. Given the increasing requirements for the speed of cryptographic systems, the optimization of these operations is a relevant area of research. In this context, the goal of our research is to develop and analyze a method for ultrafast execution of the modular addition operation in RNS, which can serve as an effective replacement for the modular multiplication and exponentiation operations, ensuring increased performance of cryptographic transformations [35].

Algebraic systems A is a plural P with operations F forming an algebraic system, for example, a group, ring, or field. Groups, rings, and fields are fundamental structures in abstract algebra, each defined by a set of axioms that specify the properties of operations. These structures are used to model a variety of mathematical objects and processes, from simple arithmetic operations to complex cryptographic algorithms.

One of the important directions in the study of algebraic systems is the study of factor structures, which allow us to build new algebraic objects based on existing ones. In particular, in the case of rings, we can construct a ring of subtraction classes, or a factor ring, which is a powerful tool for analyzing the structure of rings and their properties.

Let us consider in more detail the process of constructing a ring of subtraction classes. Let R be a ring with the operations of addition (+) and multiplication ($\times$) defined on it, and J be an ideal of the ring R. The ideal J is a subset of R that satisfies certain conditions that allow us to partition R into subtraction classes. The subtraction class containing an element $y_n \in R$ is defined as the set $y_n + J = \{y_n + j \mid j \in J\}$. The set of all subtraction classes forms a new ring, called the subtraction class ring or factor ring, and is denoted by R / J. The operations of addition and multiplication in R / J are defined in terms of the operations in R, allowing us to inherit many properties from the original ring.

Subtraction class rings are an important tool for studying the structure of rings and their applications in various fields of mathematics and computer science, including cryptography, number theory, and algebraic geometry.

The factor ring R/J can be expressed as Z/p_n, where V represents the set of integers. When p_n, the base of the RNS, is a prime number, Z/p_n forms a finite field. Given the methodology for performing arithmetic operations within the RNS, it is advantageous to focus on an arbitrary finite Galois field $GF(p_n)$, where n remains constant, corresponding to a specific defined residue system. Leveraging the aforementioned properties, modular addition and subtraction operations in RNS can be implemented without inter-digit carry propagation using the RSM through q CSAs with a range of with a range of elements representation D, effectively achieved through ring shifts of digit representations utilizing bit shift registers [36].

Based on the RSM proposed in the research, a method for performing arithmetic operations within the RNS is introduced, namely the binary remainder representation technique (BRRT). This approach, grounded in the principles of RNS, which originates from the Chinese remainder theorem [37], facilitates efficient execution of arithmetic operations, including addition, subtraction, and multiplication, on large-bit numbers. A key feature of BRRT is the utilization of binary representations for remainders [38], which allows for the substitution of complex multiplication and exponentiation operations with simpler shift and addition operations. This significantly enhances the speed of arithmetic computations, a critical factor for cryptographic algorithms where computational efficiency is paramount. Furthermore, BRRT enables parallel processing, further accelerating operation execution. These advantages render the proposed method highly promising for cryptographic systems that demand high performance and reliability [39]. Utilizing this approach, the primary (foundational) digital structure of the CSA for each modulus p_n of RNS is represented by the initial row (column) of the Cayley addition table, specifically $(y_n + u_n) \bmod p_n$, as illustrated in Fig. 1.

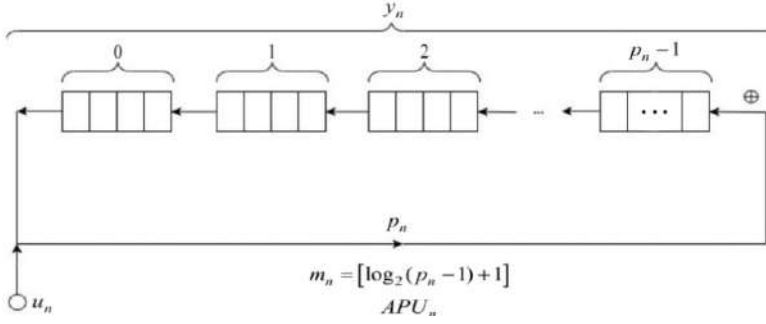


Figure 1. Primary digital structure of the CSA for modulus p_n in RNS

The primary digital structure of the CSA content for each modulus p_n can be expressed as:

$$B\_p_n = \left(B\_y_0 \,||\, B\_y_1 \,||\, \ldots \,||\, B\_y_{p_n-1}\right), \quad (1)$$

where symbol $||$ denotes the concatenation operation (combining, merging); B_y_j is a *m*-bit binary representation of the number y_j (while y_j iterates from 0 to $p_n - 1$) for modulus p_n.

The bit width *m* of the binary code of the primary digital structure of the CSA is determined by:

$$m_n = \left[\log_2(p_n - 1) + 1\right], \quad (2)$$

where square brackets [x] denotes the integer part of x, discarding the fractional part.

Given a specific modulus $p_n = 7$, the primary digital structure of the CSA content, derived from mathematical expression (1), is as follows:

$$B\_7 = \left(000 \,||\, 001 \,||\, 010 \,||\, 011 \,||\, 100 \,||\, 101 \,||\, 110\right).$$

Therefore, leveraging CSA, which are prevalent in binary PNS, especially within cryptography, facilitates the straightforward implementation of addition operations in the RNS. The degree *k* of cyclic displacements (shift) is established through the following expression, as per structure (1):

$$\left\lfloor B\_y_0 \,||\, B\_y_1 \,||\, \ldots \,||\, B\_y_{p_n-1} \right\rfloor = \\ = \left[B\_y_k \,||\, B\_y_{k+1} \,||\, \ldots \,||\, B\_y_0 \,||\, \ldots \,||\, B\_y_{p_n-1} \right]^k, \quad (3)$$

$$\left[B\_y_0 \,||\, B\_y_1 \,||\, \ldots \,||\, B\_y_{p_n-1} \right]^{-k} = \\ = \left[B\_y_{p_n-1-k} \,||\, B\_y_{p_n-k} \,||\, \ldots \,||\, B\_y_0 \,||\, B\_y_1 \,||\, \ldots \,||\, B\_y_{p_n-k-2} \right]. \quad (4)$$

It is noteworthy that $\left[B\_y_0 \,||\, B\_y_1 \,||\, \ldots \,||\, B\_y_{p_n-1} \right]^{p_n}$, implying that when $k = p_n$, all elements of the ordered set $\{B\_y_j\}$ remain in their original positions.

For the practical realization of this approach, the first term y_n indicates the quantity of CSA digit positions that hold the result of the modular operation $(y_n + u_n) \bmod p_n$, while the second term u_n indicates the number *k* shifts CSA applied to the primary CSA content (1), as defined by expressions (3)-(4). The number of shifts equals the product of the second term u_n and the bit width m_n of the CSA's primary digital structure binary code, i.e. $u_n \cdot m_n$ – the total binary digit displacement in a positive direction within the CSA Figure 2 depicts a potential operational architecture for the HSCA operating unit (OU) within the RNS.

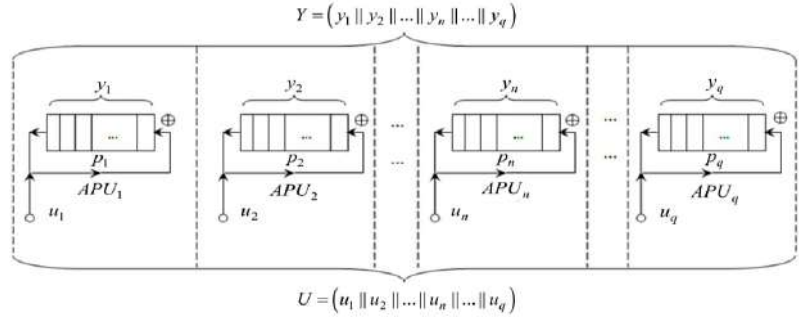


Figure 2. HSCA OU operation scheme for arbitrary RNS

For a comparative analysis of the execution time of integer addition in binary PNS and the RNS, it is necessary to determine the time required to add two numbers $Y=(y_1\|y_2\|...\|y_n\|...\|y_q)$ and $U=(u_1\|u_2\|...\|u_n\|...\|u_q)$, within the SRC utilizing the RSM. In the RSM, the time θ for modular addition of two remainders y_n and u_n, specifically in the circuit that calculates $(y_n+u_n)\bmod p_n$ $(n=\overline{1,q})$, is primarily governed by the time $\underset{\rightarrow}{\theta}$ needed to shift the primary contents of CSA digit positions (hereafter, we assume $\theta=\underset{\rightarrow}{\theta}$). The time of a single bit shift (trigger activation time) of the digital contents of CSA digit positions is given by the expression:

$$\underset{\rightarrow}{\theta}=3\cdot t', \tag{5}$$

where t' – switching time of a single logic gate (an AND, NOT, or OR gate).

Building upon prior research [40], the processing time for the modular addition of remainders y_n and u_n, specifically $(y_n+u_n)\bmod p_n$, within the RNS can be expressed by the ensuing expression:

$$\theta_{RNS}=V_n\cdot m_n\cdot\underset{\rightarrow}{\theta}, \tag{6}$$

where V_n – the second term u_n in the modular addition $(y_n+u_n)\bmod p_n$, which indicating the quantity of CSA digits cyclically shifted counterclockwise from the CSA's initial state, i.e. $V_n=\overline{0,p_n-1}$.

Thus, based on expressions (5) and (6), for an arbitrary modulus p_n of RNS, the addition time of two remainders y_n and u_n modulo p_n is defined by:

$$\theta_{RNS}=V_n\cdot\left[\log_2(p_n-1)+1\right]\cdot 3\cdot t'. \tag{7}$$

In this case, the maximum possible value of expression (7) for the arbitrary modulus m_i of RNS is defined by:

$$\theta_{RNS\_\max}=(p_n-1)\left[\log_2(p_n-1)+1\right]\cdot 3\cdot t'. \tag{8}$$

However, for the specified RNS, the maximum addition time of two numbers $Y=\left(y_1\|y_2\|...\|y_n\|...\|y_q\right)$ and $U=\left(u_1\|u_2\|...\|u_n\|...\|u_q\right)$ is determined by the maximum value of modulus p_q:

$$\theta'_{RNS\_\max}=(p_q-1)\left[\log_2(p_q-1)+1\right]\cdot 3\cdot t'. \quad (9)$$

In general, the addition time of two numbers $Y=\left(y_1\|y_2\|...\|y_n\|...\|y_q\right)$ and $U=\left(u_1\|u_2\|...\|u_n\|...\|u_q\right)$ in RNS is determined by the time (8) of realization of module operation $(y_n+u_n)\bmod p_n$ in *n*-th arithmetic processing unit (*APU*$_n$), i.e. in HSCA, in which instance $V_n\cdot m_n$ is reaches its peak $(V_n\cdot m_n=\max)$ across all $APU_e(e=\overline{1,q};\ n\neq e)$.

Previous studies [24, 40], focused on the optimization of the RSA cryptographic algorithm through the utilization of the RNS, have thoroughly examined the implementation of modular addition for one- and two-byte digit numbers. A simplified OD scheme for a one-byte HSCA processor in RNS is presented in Fig. 3.

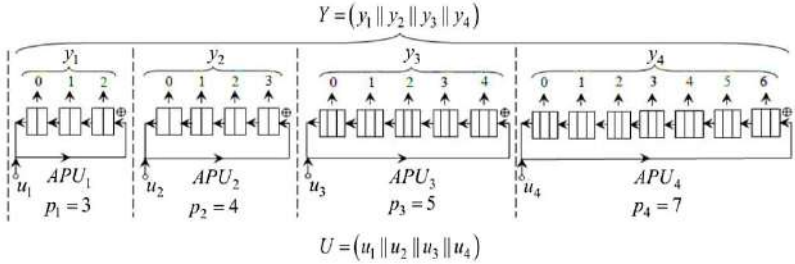


Figure 3. Simplified HSCA OU scheme of low-bit representation of numbers in RNS [40]

However, given the substantial range of number representation required for ensuring the robustness of the RSA cryptographic algorithm, there arises a necessity to investigate the effectiveness of RNS in processing large data arrays. A comprehensive analysis and illustrative examples demonstrating the advantages of employing RNS for modular addition of large-digit numbers will be presented. Cases where operand sizes reach values typical for contemporary cryptographic applications will be considered, and results will be compared with conventional computational methods. This will enable the evaluation of the practical value of RNS for enhancing the performance of cryptographic systems.

Concrete example of implementing the addition operation for two numbers within the RNS are presented, utilizing the following set of moduli: $p_1=11$,

$p_2 = 13$, $p_3 = 15$, and $p_4 = p_q = 19$, which provides a number representation range from 0 to $D = \prod_{n=1}^{q} p_n = 11 \cdot 13 \cdot 15 \cdot 19 = 40755$ in the RNS. According to equation (8), the modular addition operation's execution time depends on the second addend and the modulus p_n of the respective APU_n, under the condition that $V_n \cdot m_n = \max$.

Example 1. If the second number $(U_{10} = 95)$ is equal to $U_{RNS} = (111||100||101||000)_2 = = (7||4||5||0)_{10}$, then it is necessary to find the APU with the largest product value $V_n \cdot m_n$, therefore:

In the APU_1 with modulus $p_1 = 11$, the following values are obtained: $V_1 = 7$, $m_1 = [\log_2(p_1 - 1) + 1] = [\log_2(11-1) + 1] = 4$ and $V_1 \cdot m_1 = 7 \cdot 4 = 28.$

In the APU_2 with modulus $p_2 = 13$, the following values are obtained: $V_2 = 4$, $m_2 = [\log_2(p_2 - 1) + 1] = [\log_2(13-1) + 1] = 4$ and $V_2 \cdot m_2 = 4 \cdot 4 = 16.$

In the APU_3 with modulus $p_3 = 15$, the following values are obtained: $V_3 = 5$, $m_3 = [\log_2(p_3 - 1) + 1] = [\log_2(15-1) + 1] = 4$ and $V_3 \cdot m_3 = 5 \cdot 4 = 20.$

In the APU_4 with modulus $p_4 = 19$, the following values are obtained: $V_4 = 0$, $m_4 = [\log_2(p_4 - 1) + 1] = [\log_2(19-1) + 1] = 5$ and $V_4 \cdot m_4 = 0 \cdot 5 = 0.$

It is evident that the maximum binary digit shift, amounting to 28, is observed within the first arithmetic processing unit (APU_1). Consequently, the execution time for the addition of two numbers Y and U, represented in the RNS utilizing the ring shift mechanism, is determined by the value of the second term U and is equivalent to:

$$\theta_{RNS} = V_1 \cdot [\log_2(p_1 - 1) + 1] \cdot 3 \cdot t' = 7 \cdot 4 \cdot 3 \cdot t' = 84 \cdot t'.$$

Example 2. If the second number $(U_{10} = 78)$ is equal to $U_{RNS} = (001||000||011||010)_2 = = (1||0||3||2)_{10}$, then it is necessary to find the APU with the largest product value $V_n \cdot m_n$, therefore:

In the APU_1 with modulus $p_1 = 11$, the following values are obtained: $V_1 = 1$, $m_1 = [\log_2(p_1 - 1) + 1] = [\log_2(11-1) + 1] = 4$ and $V_1 \cdot m_1 = 1 \cdot 4 = 4.$

In the APU_2 with modulus $p_2 = 13$, the following values are obtained: $V_2 = 0$, $m_2 = [\log_2(p_2 - 1) + 1] = [\log_2(13-1) + 1] = 4$ and $V_2 \cdot m_2 = 0 \cdot 4 = 0.$

In the APU_3 with modulus $p_3 = 15$, the following values are obtained: $V_3 = 3$, $m_3 = [\log_2(p_3 - 1) + 1] = [\log_2(15-1) + 1] = 4$ and $V_3 \cdot m_3 = 3 \cdot 4 = 12.$

In the APU_4 with modulus $p_4 = 19$, the following values are obtained: $V_4 = 2$, $m_4 = \left[\log_2(p_4 - 1) + 1\right] = \left[\log_2(19-1) + 1\right] = 5$ and $V_4 \cdot m_4 = 2 \cdot 5 = 10$.

It is evident that the maximum binary digit shift, amounting to 12, is observed within the third arithmetic processing unit (*APU*3). The execution time for the addition of two numbers *Y* and *U*, represented in the RNS utilizing the RSM, is equivalent to:

$$\theta_{RNS} = V_3 \cdot \left[\log_2(p_3 - 1) + 1\right] \cdot 3 \cdot t' = 3 \cdot 4 \cdot 3 \cdot t' = 36 \cdot t'.$$

An analysis comparing the time required to perform the addition of two numbers *Y* and *U* between PNS and RNS is provided. The addition time of numbers *Y* and *U* in PNS is:

$$\theta_{PNS} = \underset{\rightarrow}{\theta} \cdot (2 \cdot r - 1) = 3 \cdot t'(16 \cdot l - 1), \tag{10}$$

where $r = 8 \cdot l$ – the number of bits for an *l*-byte data unit; $\underset{\rightarrow}{\theta} = 3 \cdot t'$ – the summation time in the (*n*+1)th binary place of the positional adder for partial sum values S_{n+1} and carry values C_{n+1}.

Recognizing that an existing method achieves a two-fold shortening of the maximum operation time for modular addition in RNS, the following applies to RSM:

$$\theta''_{RNS\_\max} = \theta'_{RNS\_\max} / 2. \tag{11}$$

The ratio of addition operation execution times in PNS and RNS will be represented by a coefficient, namely:

$$\begin{aligned} \gamma &= \theta_{PNS} / \theta''_{RNS\_\max} = \\ &= \frac{(16 \cdot l - 1) \cdot 3 \cdot \underset{\rightarrow}{\theta} \cdot 2}{(p_q - 1) \cdot \left[\log_2(p_q - 1) + 1\right] \cdot 3 \cdot \underset{\rightarrow}{\theta}} = \\ &= \frac{2 \cdot (16 \cdot l - 1)}{(p_q - 1) \cdot \left[\log_2(p_q - 1) + 1\right]}. \end{aligned} \tag{12}$$

The computational assessment and comparative evaluation of arithmetic operation execution times during cryptographic transformations demonstrated the significant effectiveness of the BRRT method, which utilizes the RSM within the RNS, when contrasted with a method employed in PNS (see Table 1). It is important to note that Table 1 specifically presents a comparative analysis of the modular addition operation within the RNS versus the PNS. While these results highlight the efficiency gains at the fundamental arithmetic level, a direct comparative analysis of the overall RSA cryptosystem's performance using the proposed RNS-based acceleration against other established RSA acceleration methods (e.g., Montgomery multiplication, Karatsuba algorithm, or dedicated hardware implementations) is a complex task that requires specific experimental setups and is beyond the scope of this initial theoretical and methodological paper.

Table 1

Data of comparative analysis of time of addition operation

l (r)	PNS	RNS			%
	$\theta_{PNS} / 3 \cdot t'$	p_q	m_q	$\theta^{*}_{RNS\_\max} / 3 \cdot t'$	
4 (32)	63	19	5	48	31.25
8 (64)	127	30	5	75	69.33

The presented data are derived without the inclusion of supplementary algorithms, which, if implemented, could expedite the execution of modular arithmetic operations. The resulting mathematical expressions (7)-(9) and (12), along with the determined operational times for arithmetic operations in RNS, can be utilized for evaluating and comparing the computational complexity of RSA cryptographic transformation algorithms.

Conclusions and Prospects. Digital transformation necessitates a rethinking of existing approaches to digital security. To strengthen the ability of entities to counter dynamic and growing risks in the field of cyber security, existing tools and technologies need to be improved. Of particular importance in this context is the modernization of the RSA cryptosystem as one of the basic mechanisms for data protection, which ensures resistance to modern cyber threats and increases the level of trust in digital services.

This paper introduced a novel method for accelerating cryptographic transformations within Galois fields, focusing on improving the efficiency of RSA cryptosystems with public keys. The proposed method leverages the RNS. By exploiting the fundamental theoretical properties of RNS, we have effectively streamlined the execution of modular operations essential for cryptographic tasks. The core advantage of this approach lies in its inherent parallelism, which fundamentally bypasses the carry propagation bottleneck that limits the speed of traditional positional number systems. This enables modular operations to be executed in constant time, regardless of the operand's bit length, a critical achievement for modern, high-bit-length cryptographic keys.

Furthermore, we have presented a practical method for realizing arithmetic operations in RNS based on a ring shift mechanism, namely the binary remainder representation technique. The efficiency analysis and concrete technical implementation examples of modular arithmetic operations substantiate the practical feasibility of this approach. This method of information processing is highly recommended for crypto accelerators enabling real-time security surveillance and secure authentication.

The application of the proposed method significantly reduces the execution time of operations, which is critical for ensuring real-time security. The obtained results confirm the practical value of RNS in enhancing the performance of cryptographic systems, particularly when processing large data arrays, which is typical for modern cryptographic applications.

The research findings offer significant potential for application in systems and devices designed for high-throughput, real-time digital information processing. Practical examples confirm its feasibility for real-time applications, strengthening digital security infrastructure, especially in dynamic environments. The implementation of this method not only improves the speed of critical cryptographic processes, but also enhances the overall security posture of digital systems. Moreover, while this study specifically focuses on RSA, the core principles of RNS-based modular arithmetic and the cyclic shift mechanism are inherently adaptable to other cryptographic algorithms that heavily rely on modular exponentiation and multiplication, such as ElGamal, Diffie-Hellman, and various elliptic curve cryptography (ECC) schemes. The parallel processing capabilities offered by RNS make it a versatile foundation for accelerating a broad spectrum of public-key cryptographic operations beyond RSA. As such, it represents a substantial advancement in the field of secure computation.

The research findings offer significant potential for application in resource-constrained systems, such as embedded devices, Internet of Things (IoT), and industrial control systems, where high performance must be achieved with limited computational power. By enhancing the speed and efficiency of cryptographic operations, this method contributes to strengthening the digital security infrastructure, particularly critical in dynamic and challenging environments.

Future work will focus on a comprehensive experimental evaluation of the proposed RNS-based RSA acceleration method against state-of-the-art hardware and software implementations of RSA, including detailed comparative performance indicators such as throughput, latency, and resource utilization. This will provide a more objective and complete assessment of its practical advantages and potential for real-world deployment. Moreover, the core principles of RNS-based modular arithmetic and the cyclic shift mechanism are inherently adaptable to other public-key cryptographic algorithms that heavily rely on modular exponentiation and multiplication, such as ElGamal and Diffie-Hellman. Further research will also investigate the method's applicability to post-quantum cryptography (PQC), which demands exceptional computational efficiency for its large-integer-based algorithms.

References

1. Onyshchenko, S., Yanko, A., Hlushko, A., & Sivitska, S. (2020). Conceptual principles of providing the information security of the national economy of Ukraine in the conditions of digitalization. *International Journal of Management, 11*(12), 1709–1726. https://doi.org/10.34218/IJM.11.12.2020.157

2. Kharchenko, V., et al. (2022). Combining Markov and semi-Markov modelling for assessing availability and cybersecurity of cloud and IoT systems. *Cryptography, 6*(3), 44. https://doi.org/10.3390/cryptography6030044

3. Hlushko, A. D., & Yanko, A. S. (2019). Optimal reservation of data in the system of residual classes in the direction of ensuring information security of the

national economy. *Economics and Region, (4)75*, 35–44. https://doi.org/10.26906/EiR.2019.4(75).1814

4. Hlushko, A., Yanko, A., & Bilko, S. (2025). Digital transformation of business processes: Security aspect. *Digital Economy and Economic Security, 2*(17). https://doi.org/10.32782/dees.17-26

5. Saeed, S., et al. (2023). Digital transformation and cybersecurity challenges for businesses resilience: Issues and recommendations. *Sensors, 23*(15), 6666. https://doi.org/10.3390/s23156666

6. Shahim, A. (2021). Security of the digital transformation. *Computers & Security, 108*, 102345. https://doi.org/10.1016/j.cose.2021.102345

7. Monroe, D. (2023). Post-quantum cryptography. *Communications of the ACM, 66*(2), 15–17. https://doi.org/10.1145/3575664

8. Kuznetsov, A., et al. (2022). Cryptographic transformations in a decentralized blockchain environment. In *Information Security Technologies in the Decentralized Distributed Networks* (pp. 89–113). Cham. https://doi.org/10.1007/978-3-030-95161-0_4

9. Pasumponpandian, D. (2020). Development of secure cloud-based storage using the Elgamal hyper elliptic curve cryptography with fuzzy logic based integer selection. *Journal of Soft Computing Paradigm, 2*(1), 24–35. https://doi.org/10.36548/jscp.2020.1.003

10. Ullah, S., & Din, N. (2021). Blind signcryption scheme based on hyper elliptic curves cryptosystem. *Peer-to-Peer Networking and Applications, 14*(2), 917–932. https://doi.org/10.1007/s12083-020-01044-8

11. Berini, A. D. e., et al. (2023). HCALA: Hyperelliptic curve-based anonymous lightweight authentication scheme for Internet of Drones. *Pervasive and Mobile Computing, 101798.* https://doi.org/10.1016/j.pmcj.2023.101798

12. Onyshchenko, S., Yanko, A., Hlushko, A., Maslii, O., & Cherviak, A. (2023). Cybersecurity and improvement of the information security system. *Journal of the Balkan Tribological Association, 29*(5), 818–835. https://scibulcom.net/en/article/L8nV7It2dVTBPX09mzWB

13. Albusifi, S., Mugassab, S., & Elferjani, A. (2021). RSA cryptography algorithm and its applications to security system by using linear congruence. *International Journal of Multidisciplinary Sciences and Advanced Technology, Special 1*, 558–564. https://www.researchgate.net/publication/380320192_RSA_Cryptography_Algorithm_and_its_Applications_to_Security_System_by_Using_Linear_Congruence/stats

14. Al-Hamami, A. H., & Aldariseh, I. A. (2012, November 26–28). Enhanced method for RSA cryptosystem algorithm. *2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT).* Kuala Lumpur, Malaysia. https://doi.org/10.1109/acsat.2012.102

15. Mammadov, F. K. (2023). New approach to book cipher: Web pages as a cryptographic key. *Advanced Information Systems, 7*(1), 59–65. https://doi.org/10.20998/2522-9052.2023.1.10

16. Shang, Y., et al. (2008). Study on the scheme for RSA iterative encryption system. *Computer and Information Science*, *1*(3). https://doi.org/ 10.5539/cis.v1n3p120

17. Barker, E., & Roginsky, A. (2019). Transitioning the use of cryptographic algorithms and key lengths. Gaithersburg, MD: National Institute of Standards and Technology. https://doi.org/10.6028/nist.sp.800-131ar2

18. Vorobets, H., Vorobets, O., Luchyk, O., & Rusyn, V. (2023). Information technology and software for simulation, synthesis and research of data crypto protection methods. *Security of Infocommunication Systems and Internet of Things*, *1*(2), 02011. https://doi.org/10.31861/sisiot2023.2.02011

19. Nan, L., et al. (2019). A VLIW architecture stream cryptographic processor for information security. *China Communications*, *16*(6), 185–199. https://doi.org/10.23919/jcc.2019.06.015

20. Krasnobayev, V., Yanko, A., Martynenko, A., & Kovalchuk, D. (2023). Method for computing exponentiation modulo the positive and negative integers. In *XI International Scientific and Practical Conference on Information Control Systems & Technologies (ICST-2023)* (pp. 374–383). Odessa, Ukraine: CEUR. https://ceur-ws.org/Vol-3513/paper31.pdf

21. Zhang, J., Liu, J., & He, J. (2023). Composing parameter-efficient modules with arithmetic operation. *Advances in Neural Information Processing Systems*, *36*, 12589–12610. https://doi.org/10.48550/arXiv.2306.14870

22. Vollala, S., et al. (2014, February 21–22). Efficient modular multiplication algorithms for public key cryptography. *2014 IEEE International Advance Computing Conference (IACC).* Gurgaon, India. https://doi.org/ 10.1109/iadcc.2014.6779297

23. Kochan, R., et al. (2020, September 17–18). Development of methods for improving crypto transformations in the block-symmetric code. *2020 IEEE 5th International Symposium on Smart and Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS).* Dortmund. https://doi.org/10.1109/idaacs-sws50031.2020.9297102

24. Onyshchenko, S., Yanko, A., & Hlushko, A. (2023). Improving the efficiency of diagnosing errors in computer devices for processing economic data functioning in the class of residuals. *Eastern-European Journal of Enterprise Technologies*, *5*(4(125)), 63–73. https://doi.org/10.15587/1729-4061.2023.289185

25. Wang, X., et al. (2021). Processing methods for digital image data based on the geographic information system. *Co* [26] A comparative investigation for flatness and parallelism measurement uncertainty evaluation using laser interferometry and image processing. *Indian Journal of Engineering and Materials Sciences*, 31(1). (2024). https://doi.org/10.56042/ijems.v31i1.4887

27. Hofheinz, D., Hövelmanns, K., & Kiltz, E. (2017). A modular analysis of the Fujisaki-Okamoto transformation. In Y. Kalai & L. Reyzin (Eds.), *Proceedings of the Theory of Cryptography, TCC 2017* (Lecture Notes in Computer Science, Vol. 10677, pp. 341–371). Cham: Springer. https://doi.org/10.1007/978-3-319-70500-2_12

28. Nguyen, D. C., & Pershin, Y. (2024). Fully parallel implementation of digital memcomputing on FPGA. In *2024 IEEE 67th International Midwest Symposium on Circuits and Systems (MWSCAS)* (pp. 263–266). Springfield, MA, USA: IEEE. https://doi.org/10.1109/MWSCAS60917.2024.10658882

29. Krasnobayev, V., Kuznetsov, A., Yanko, A., Akhmetov, B., & Kuznetsova, T. (2021). Processing of the residuals of numbers in real and complex numerical domains. In T. Radivilova, D. Ageyev, & N. Kryvinska (Eds.), *Data-Centric Business and Applications* (Lecture Notes on Data Engineering and Communications Technologies, Vol. 48, pp. 529–555). Cham: Springer. https://doi.org/10.1007/978-3-030-43070-2_24

30. Liu, N., & Wang, H. (2021). The characterizations of WG matrix and its generalized Cayley–Hamilton theorem. *Journal of Mathematics*, 2, 1–10. https://doi.org/10.1155/2021/4952943

31. Kovács, I., & Kwon, Y. (2021). Regular Cayley maps for dihedral groups. *Journal of Combinatorial Theory, Series B*, 148, 84–124. https://doi.org/10.1016/j.jctb.2020.12.002

32. Brown, S. (2024). Distance between consecutive elements of the multiplicative group of integers modulo n. *Notes on Number Theory and Discrete Mathematics*, 30(1), 81–99. https://doi.org/10.7546/nntdm.2024.30.1.81-99

33. Kress, J., Schöbel, K., & Vollmer, A. (2023). An algebraic geometric foundation for a classification of second-order superintegrable systems in arbitrary dimension. *The Journal of Geometric Analysis*, 33(11). https://doi.org/10.1007/s12220-023-01413-8

34. Singh, P., Pranav, P., Anwar, S., & Dutta, S. (2024). Leveraging generative adversarial networks for enhanced cryptographic key generation. *Concurrency and Computation: Practice and Experience*, 36(22). https://doi.org/10.1002/cpe.8226

35. Antão, S., & Sousa, L. (2013). An RNS-based architecture targeting hardware accelerators for modular arithmetic. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 2572–2576). Vancouver, BC, Canada: IEEE. https://doi.org/10.1109/ICASSP.2013.6638120

36. Krasnobayev, V., Kuznetsov, A., Yanko, A., & Kuznetsova, T. (2020). The data errors control in the modular number system based on the nullification procedure. In *3rd International Workshop on Computer Modeling and Intelligent Systems (CMIS-2020)* (pp. 580–593). Zaporizhzhia, Ukraine: CEUR. https://doi.org/10.32782/cmis/2608-45

37. Adigun, A. A., Abolarinwa, M. O., Ojo, O. E., Oladimeji, A. I., & Bakare, O. S. (2024). Enhanced local binary pattern algorithm for facial recognition using Chinese remainder theorem. *Dutse Journal of Pure and Applied Sciences*, 10(1c), 255–262. https://doi.org/10.4314/dujopas.v10i1c.24

38. Jackiewicz, W., & Jackiewicz, K. (2024). Algorithmic sequencing of remainders for hyperbolic functions in discrete space (pp. 1–16). https://doi.org/10.13140/RG.2.2.15943.92328

39. Liu, R., Li, Z., Zhang, X., Li, W., Shen, L., Tang, R., Luo, Z., Chen, X., Han, Y., & Tang, M. (2024). Crypto-DSEDA: A domain-specific EDA flow for

CiM-based cryptographic accelerators. *IEEE Design & Test*, 4(5), 46–54. https://doi.org/10.1109/MDAT.2024.3395987

40. Krasnobayev, V., Yanko, A., & Koshman, S. (2016). Conception of realization of cryptographic RSA transformations with using of the residue number system. *Computer Science and Cybersecurity*, 2, 5–12. Retrieved from https://periodicals.karazin.ua/cscs/article/download/6207/5745/*mplexity, 2021*, 1–12. https://doi.org/10.1155/2021/2319314

ВПРОВАДЖЕННЯ КРИПТОГРАФІЧНИХ ПЕРЕТВОРЕНЬ НА ОСНОВІ СИСТЕМИ ЗАЛИШКОВИХ КЛАСІВ ДЛЯ ПІДВИЩЕННЯ ЦИФРОВОЇ БЕЗПЕКИ

Ph.D. А. Янко[0000-0003-2876-9316], **Dr.Sci. В. Краснобаєв**[0000-0001-5192-9918], **Ph.D. А. Глушко**[0000-0002-4086-1513], **М. Мизюра** [0009-0009-9301-2054]
Національний університет «Полтавська політехніка імені Юрія Кондратюка», Україна
EMAIL: al9_yanko@ukr.net

Анотація. *У роботі розглядаються критичні питання посилення безпеки бізнесу в умовах цифрової трансформації. Автори демонструють, що розширення процесів цифровізації вимагає перегляду концепції економічної безпеки. Обґрунтовано, що для зміцнення стійкості бізнесу до ризиків та загроз цифровій безпеці необхідно впроваджувати низку заходів, спрямованих на захист конфіденційності, цілісності та доступності інформації. Було проведено дослідження кіберзагроз для суб'єктів національної економіки та громадян, у тому числі з використанням інструментів штучного інтелекту. Це дало змогу визначити пріоритетну сферу захисту даних – удосконалення RSA-криптосистеми. У дослідженні деталізовано розробку ефективних стратегій обробки інформації для зменшення затримки криптографічних функцій RSA. Для прискорення криптографічних перетворень RSA в цьому дослідженні запропоновано методи високошвидкісної обробки інформації. Основа запропонованого методу включає реалізацію механізму циклічного (кільцевого) зсуву з використанням модулярної арифметики, повністю реалізованого системою залишкових класів (СЗК). Застосування СЗК демонструє її ефективність у структуруванні процесу реалізації модулярних цілочислових арифметичних операцій для прискорення криптографічних перетворень з відкритим ключем.*

Ключові слова: *техніка представлення двійкового залишку, криптографічний захист інформації, алгоритм криптографії, масиви циклічного зсуву, цифрова трансформація, високошвидкісні криптографічні прискорювачі, модулярні арифметичні коди, система залишкових класів, механізм кільцевого зсуву.*

УДК 519.85
DOI https://doi.org/10.36059/978-966-397-538-2-2

МЕТОД НАЛАШТУВАННЯ РЕГУЛЯТОРІВ ІЗ СУТТЄВИМИ НЕЛІНІЙНОСТЯМИ

Dr.Sci. О. Лисенко[1][0000-0002-7276-9279], **Dr.Sci. О.Тачиніна** [2][0000-0001-7081-0576], **Ph.D. С. Пономаренко**[1][0000-0001-5512-3778]

[1]Національний технічний університет України «КПІ ім. Ігоря Сікорського», Україна,

[2]Державний університет «Київський авіаційний інститут», Україна
EMAIL:lysenko.a.i.1952@gmail.com, tachinina5@gmail.com, sol_@ukr.net

Анотація. *В статті розроблено інженерний метод налаштування (або переналаштування в процесі експлуатації) регуляторів електроприводів вузлів рухомості маніпулятора, який враховує наявність суттєвих нелінійностей. Цей метод дозволяє запобігти появі «первинних автоколиваннь» в системі автоматичного керування електроприводами вузлів рухомості маніпулятора, що стимулюють виникнення резонансних пружних вібрацій та автоколивань (ефект автопружності). Запропонований метод дозволяє не тільки усунути причину виникнення ефекту автопружності, алей зробити це на інженерному рівні володіння математичним апаратом, системами комп'ютерної математики та навичками програмування. Прояв ефекту автопружності пов'язаний із наявністю таких факторів, як: динамічні властивості приводу вузлів рухомості; пружна гнучкість маніпуляторів; суттєві нелінійності конструктивно-технологічного характеру або такі, що виникають в процесі експлуатації в механічних та електричних пристроях. Інженерна простота та зручність методу виражається в тому, що налаштування регуляторів електроприводів вузлів рухомості при виробництві маніпулятора або їх переналаштування в процесі експлуатації не потребує спеціалізованого наукового дослідження, а може бути виконано фахівцем із інженерним рівнем математичної підготовки в інтерактивному режимі за короткий час.*

Ключові слова: *система автоматичного керування, ПІД – регулятор, суттєва нелінійність, чисельні методи оптимізації, комп'ютерна математична модель, антена спрямованої дії, маніпулятор.*

1 Вступ

В аерокосмічній галузі застосовуються багатоланцюгові маніпулятори, які повинні забезпечувати плавне та нечутливе до дії зовнішніх збурень переміщення у тривимірному просторі інерційних об'єктів із значною «парусністю» [1]. Маніпулятори аерокосмічної галузі, за звичай, повинні бути легкими, мати, як мінімум, три степені рухомості (тобто, вузлів рухомості) і охоплювати значну робочу зону [2]. При цьому регулятори електроприводів вузлів рухомості маніпулятора повинні забезпечувати можливість виконання

швидкої переорієнтації та прецизійного стеження за вхідною керуючою дією [3].

Пружність маніпулятора разом із динамічними властивостями приводів, парусністю об'єктів, які переміщує маніпулятор, та об'єктивною наявністю в механічній конструкції маніпулятора суттєвих нелінійностей типу гістерезис, зона нечуттєвості та насичення призводять до виникнення в процесі експлуатації ефекту так званої «автопружності» (ефект автопружності (ЕАП)). Ефект автопружності починається із «первинних автоколивань» (ПА), які пов'язані із неврахуванням дії суттєвих нелінійностей при налаштуванні регуляторів електроприводів вузлів рухомості і далі розвивається навіть до резонансних режимів, що пов'язані із пружністю конструкції маніпулятора [4, 5, 6].

Зрозуміло, що постає актуальна проблема стосовно такого налаштування (або переналаштування в процесі експлуатації) регуляторів електроприводів вузлів рухомості маніпулятора, яке враховує наявність суттєвих нелінійностей і подавляє причину виникнення ЕАП, тобто подавляє «первинні автоколивання».

Ще раз підкреслимо, що за фізичним змістом суттєва нелінійність або притаманна конструктивному-технологічному виконанню маніпулятора, або виникає в процесі нормальної експлуатації в наслідок зносу та биття в підшипниках електроприводу, або спричинена зовнішніми механічними (фізичними) ударами, або спеціально задається алгоритмом керування вузлом рухомості маніпулятора.

2 Постановка проблем

В теперішній час існують добре розроблені методи аналізу та синтезу режимів роботи нелінійних систем автоматичного керування (САК) в загалі і, зокрема, суттєво нелінійних САК, які спрямовані: по-перше, на з'ясування умов виникнення автоколивань або нестійкості; по-друге, на пошук шляхів стабілізації та (або) усунення автоколивань: частотні методи (частотний метод аналізу стійкості Попова; метод гармонічної лінеарізації Гольдфарба-Попова); метод фазової площини (фазових траєкторій); метод припасовування (метод точкових перетворень Андронова); метод статистичної лінеарізації; методи теорії катастроф; метод функцій Ляпунова [7].

Існуючі методи аналізу та синтезу нелінійних САК базуються на фундаментальних теоретичних математичних роботах А.М. Ляпунова, де викладено необхідні і достатні умови стійкості нелінійних систем. Ці методи орієнтовані на математиків науковців, а не на інженерів експлуатаційників.

В аерокосмічній техніці маніпулятори виготовляються і використовуються масово. Проблема ЕАП має масовий характер. Налаштовувати та переналаштовувати в процесі експлуатації регулятори потрібно масово. Це означає, що потрібен метод, який доступний по теоретичному рівню для інженерів експлуатаційників та орієнтований на застосування сучасних і перспективних систем комп'ютерної математики із розвинутим

спеціалізованим програмним забезпеченням, яке спрямовано на розв'язання статичних та динамічних задач оптимізації. Тобто потрібен інженерний метод для інженерної практики.

В системі комп'ютерної математики *MATLAB* існує так званий пакет *Nonlinear Control Design* (*NCD Blockset*), який реалізує метод динамічної оптимізації із врахуванням заданих користувачем часових обмежень [8]. Цей метод можна вважати напівінженерним з тієї причини, що евристичний спосіб завдання часових обмежень (бажаної моделі зміни в часі вихідних координат САК) – це скоріше мистецтво, аніж інженерний спосіб.

Постає науково-технічна задача: необхідно розробити інженерний метод доступний для масового використання в інженерної практики для налаштування (або переналаштування в процесі експлуатації) регуляторів електроприводів вузлів рухомості маніпулятора, який враховує наявність суттєвих нелінійностей.

В статті пропонується інженерний метод налаштування регуляторів електроприводів вузлів рухомості маніпулятора із суттєвими нелінійностями, що базується на використанні методів чисельної оптимізації алгоритмічно заданого критерія.

3 Інженерний метод налаштування регуляторів електроприводів вузлів рухомості маніпулятора із суттєвими нелінійностями (в подальшому будемо скорочено позначати цей метод ІМ)

Оскільки в ІМ мова йде про «налаштування», то фактично ІМ – це метод параметричного синтезу регуляторів систем автоматичного керування із заданою структурою. Структура САК електроприводів вузлів рухомості маніпуляторів в інженерній практиці достатньо відпрацьована і у більшості випадків вона є двоконтурною [9].

Інженерний метод базується:

- на використанні чисельних методів оптимізації критерія, який залежить від вектору параметрів регулятора і, при цьому, в явному аналітичному вигляді цю залежність отримати або неможливо, або дуже складно;
- на використанні комп'ютерних математичних моделей (алгоритмів моделювання) динамічних ланок систем автоматичного керування (САК) та статичних характеристик нелінійних елементів як із суттєвими, так і гладкими (несуттєвими) нелінійностями, завдяки яким і відбувається обчислення кількісного значення критерію (тобто чисельне значення критерію знаходиться завдяки застосуванню алгоритмів моделювання: звідси і походить назва – алгоритмічно заданий критерій).

Інженерний метод дозволяє: алгоритмічно обчислювати кількісні значення критерію для заданих значень вектору параметрів регулятора з метою отримання кількісної інтегральної оцінки ефективності функціонування САК; візуально спостерігати зміну в часі змінних, які характеризують якість функціонування САК і в інтерактивному режимі аналізувати та синтезувати алгоритм роботи регулятора. При використанні ІМ нелінійний елемент (НЕ),

який є нелінійною ланкою спрямованої дії, моделюється як статична ланка із статичною характеристикою, яка відповідає фізичному змісту дії нелінійного елементу у складі САК. Інженерний метод може застосовуватися для параметричного синтезу регулятора (ІМ ПСР) та для переналаштування регулятора (ІМ ПНР) в процесі експлуатації.

Вихідними даними для застосування ІМ ПСР є математична модель САК – прототипу, тобто структурна схема САК – прототипу із відомими математичними моделями усіх ланок спрямованої дії, а також, ланцюгів, які включають до свого складу нелінійні елементи.

Вихідними даними для застосування ІМ ПНР є реальна САК [9].

Етапи ІМ ПСР:

1. Побудова структури модернізованої САК. Виконується вибір типу регулятора із відомою структурою законів керування та підключення регулятора до структурної схеми комп'ютерної математичної моделі САК – прототипу. Параметри регулятора підлягають визначенню завдяки чисельній оптимізації алгоритмічно заданого критерія.
2. Пошук першого наближення до оптимальних значень параметрів регулятора, які в подальшому будуть обчислені завдяки чисельній оптимізації алгоритмічно заданого критерія.
3. Вибір еталонної математичної моделі для моделювання бажаної зміни в часі тих змінних, які в математичній моделі САК вибрані для налаштування.
4. Вибір критерія оцінювання розбіжності (критерія налаштування) в часі між відповідними вихідними змінними (координатами) еталонної математичної моделі (бажана зміна в часі) та математичної моделі САК (зміна в часі змінних при заданих значеннях параметрів регулятора).
5. Вибір методу чисельної оптимізації параметрів регулятора.
6. Підключення еталонної моделі та алгоритму обчислення кількісного значення критерія оцінювання розбіжності (критерія налаштування) до структурної схеми модернізованої САК.
7. Застосування чисельного методу для параметричного синтезу (налаштування), тобто пошуку оптимальних значень параметрів регулятора.

Етапи ІМ ПНР складаються з 3 – 8 етапів ІМ ПСР, де математичні моделі елементів САК – прототипу та модернізованої САК замінюються натурними моделями (тобто реальними обладнанням та пристроями).

Розглянемо застосування ІМ ПСР для налаштування регулятора САК електроприводу вузла рухомості ланки, на якій розташовано робочий орган із закріпленою антеною спрямованої дії. Ця антена призначена для передачі зондуючих сигналів в режимі ідентифікації «рою» малорозмірних об'єктів (тобто, сканування виділеної зони при пошуку активно маневруючих групових об'єктів та їх групового супроводження (захват руху групового об'єкту). Вхідний сигнал $u(t)$ задає кутову швидкість робочого органу маніпулятора $x(t)$, який є вихідним сигналом [10].

Вихідні дані представлені на Рис. 1 у вигляді структурної схеми комп'ютерної математичної моделі САК – прототипу електроприводу вузла рухомості ланки, на якій розташовано робочий орган із закріпленою антеною спрямованої дії. Всі змінні представлені у безрозмірній формі (відносні значення, що обчислені по відношенню до номінальних значень).

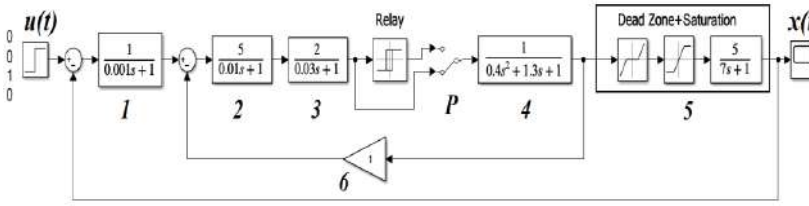


Рисунок 1. Структурна схема комп'ютерної математичної моделі САК – прототипу: САК – прототипу *L* (перемикач *P* у нижньому положенні); САК – прототипу *R* (перемикач *P* у верхньому положенні)

Математична модель САК-прототипу складається із математичних моделей: 1 – сенсора кутової швидкості робочого органу; 2, 3, *Relay*, 4 – електронного перетворювача (ЕП) та електричної частини електроприводу; 5 – механічної частини електроприводу із робочим органом; 6 – сенсора зусиль.

Структурна схема комп'ютерної математичної моделі САК – прототипу *L* дозволяє моделювати САК-прототипу із лінійною математичною моделлю ЕП (перемикач *P* у нижньому положенні), а САК – прототипу *R* моделює ЕП з підсилювачем потужності із релейною характеристикою (-10;10), перемикач *P* у верхньому положенні). Підкреслимо, що в обох комп'ютерних математичних моделях САК – прототипу *L* та *R* моделюється суттєва нелінійність механічної частини електроприводу із ланкою, на якій розташований робочий орган, у вигляді зони нечуттєвості (-1;1) та насичення (-5; 5).

Перейдемо до виконання етапів ІМ ПСР

Етап 1.

В якості регуляторів для внутрішнього та зовнішнього контурів обираємо пропорційно -, інтегрально -, диференціюючі регулятори (PID – регулятори) (Рис. 2). В математичних моделях регуляторів врахована фізична реалізуємось цих регуляторів (реальна диференціююча ланка та обмеження типу зони насичення (-10;10) для вихідного сигналу ПІД – регуляторів) [1, 7, 9].

Етап 2.

Параметричному синтезу (налаштуванню) підлягають параметри (*Kp*1, *Ki*1, *Kd*1) та (*Kp*2, *Ki*2, *Kd*2) регуляторів відповідно PID-1 та PID-2. Згідно рекомендаціям [4] покладаємо, що $taudj = 0.15 \cdot \frac{Kdj}{Kpj}(j = 1,2)\cdot$

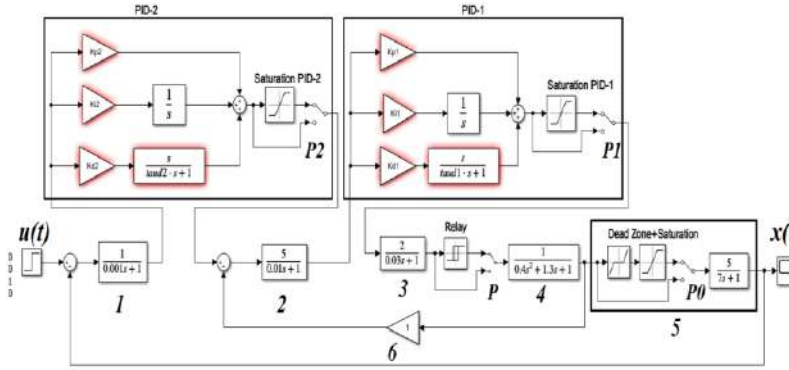


Рисунок 2. Структурна схема комп'ютерної математичної моделі модернізованої САК *L* (перемикач *P* у нижньому положенні) та модернізованої САК *R* (перемикач *P* у верхньому положенні) електроприводу вузла рухомості ланки, на якій розташовано робочий орган із закріпленою антеною спрямованої дії: регулятори PID-1 та PID-2 підключено відповідно у внутрішній контур та зовнішній контури; положення перемикачів *P*0, *P*1 та *P*2 задається на етапах виконання методу

Пошук першого наближення (первинне налаштування) до оптимальних значень параметрів регуляторів PID-1 та PID-2 обох схем модернізованої САК (*L* та *R*) (див. рис. 2) виконаємо із використанням методу коливань Зіглера-Ніколса [7] без врахування усіх суттєвих нелінійностей (тобто усі перемикачі на рис.2 перебувають у нижньому положенні).

В результаті отримаємо: PID-1 (*Kp*1 =1.2, *Ki*1 = 3, *Kd*1 = 0.12, *taud*1 = 0.015) та PID-2 (*Kp*2 = 2.1, *K*i2 = 3.5, *Kd*2 = 0.315, *taud*2 = 0.0225).

Перехідну функцію комп'ютерної математичної моделі модернізованих САК *L* та *R* (див. рис. 2), яка побудована для значень параметрів регуляторів обчислених без врахування дії суттєвих нелінійностей, представлено на Рис. 3.

Етап 3.

Для налаштування обираємо вихідну координату модернізованих САК *L* та САК *R*, тобто кутову швидкість робочого органу маніпулятора $x(t)$. Приймаємо, що тривалість перехідного процесу треба зменшити до 2 с, а перерегулювання приблизно у двічі.

Враховуючи вигляд перехідної функції (див. рис. 3) та аналізуючи фізичний зміст процесів, які відбуваються в модернізованих САК *L* та САК *R*, в якості еталонної моделі зміни в часі вихідної координати модернізованих САК *L* та САК *R* обираємо стандартну форму Баттерворта 5-го порядку [7, 9].

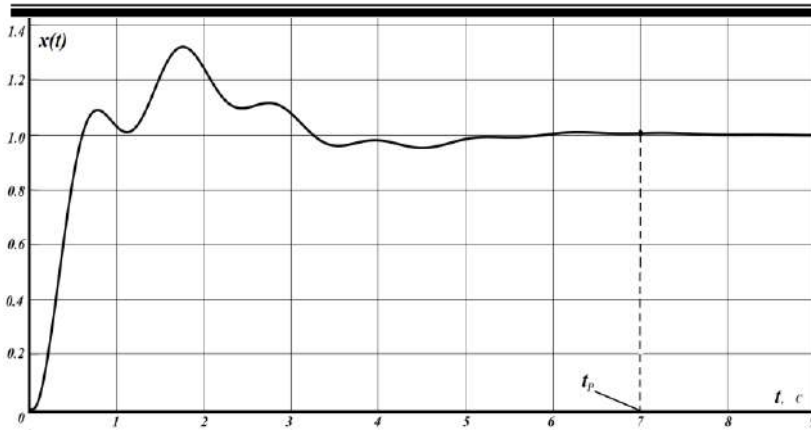


Рисунок 3. Перехідна функція SISO LTI математичної моделі модернізованих САК *L* та *R* (усі перемикачі на рис. 2 перебувають у нижньому положенні): час тривалості перехідного процесу $t_p = 7c;$ перерегулювання 35 %

Неперервна передавальна функція системи, що дозволяє генерувати неперервний перехідний процес, який відповідає стандартній формі Баттерворта має вигляд:

$$W_{\text{СФБ}}(S) = \frac{\nu_0^k}{P_k(S)},$$

де $P_k(S)$ – характеристичний поліном *k*-ого порядку. Характеристичний поліном 5-ого порядку стандартної форми Баттерворта має вигляд:

$$P_k(S) = S^5 + 3.24 \cdot S^4 \cdot \nu_0 + 5.24 \cdot S^3 \cdot \nu_0^2 + 5.24 \cdot S^2 \cdot \nu_0^3 + 3.24 \cdot S \cdot \nu_0^4 + \nu_0^5.$$

Використовуючи передавальну функцію еталонної моделі [7, 9]

$$W(S) = \frac{\nu_0^5}{S^5 + 3.24 \cdot S^4 \cdot \nu_0 + 5.24 \cdot S^3 \cdot \nu_0^2 + 5.24 \cdot S^2 \cdot \nu_0^3 + 3.24 \cdot S \cdot \nu_0^4 + \nu_0^5}$$

будуємо нормовану еталонну передавальну функцію при $\nu_0 = 1$ (Рис. 4) і знаходимо значення параметра ν_0, при якому час тривалості бажаного еталонного перехідного процесу t_{PE} не перевищує 2 с:

$$\nu_0 = \frac{t_{PN}}{t_{PE}} = \frac{14}{2} = 7,$$

де t_{PN} – час тривалості нормованого еталонного перехідного процесу.

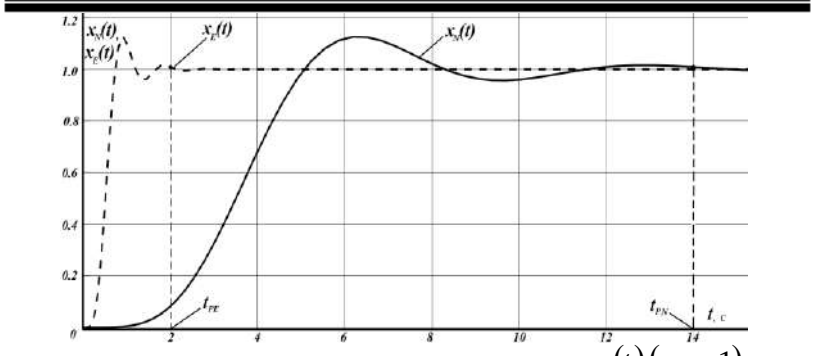


Рисунок 4. Перехідні функції на виході нормованої $x_N(t)(\nu_0 = 1)$ та бажаної $x_E(t)(\nu_0 = 7)$ еталонних моделей: t_{PN}, t_{PE} – час тривалості перехідного процесу відповідно у нормованій та бажаній еталонних моделях

Етап 4.

В якості критерія оцінювання розбіжності в часі між відповідними вихідними змінними (координатами) еталонної математичної моделі (бажана зміна в часі) та математичної моделі САК (зміна в часі змінних при заданих значеннях параметрів регуляторів PID-1 та PID-2), тобто критерія налаштування, обираємо інтеграл зваженого квадрату помилки:

$$I = \int_{t_0}^{t_f} \alpha(t) \cdot (x_{\text{em}}(t) - x(t))^2 dt,$$

де приймаємо, що ваговий коефіцієнт $\alpha(t) = 1, t_0 = 0, t_f = 0, t_f = 10c.$

Кількісне значення критерія відображається на дисплеї *D* (рис. 5).

Етап 5.

В якості методу чисельної оптимізації параметрів регуляторів PID-1 та PID-2 пропонується обрати чисельний метод серед методів, які викладено у [7, 9].

Як приклад, на етапі 7 розглянемо використання чисельного методу Гаусса-Зейделя.

Етап 6.

На рис. 5 показано підключення еталонної моделі 7 та алгоритму обчислення кількісного значення критерія I (інтеграл зваженого квадрату помилки, значення якого відображається на дисплеї *D*) до комп'ютерної математичної моделі модернізованої САК.

Структурна схема, що представлена на Рис. 5, дозволяє моделювати різновиди модернізованої САК в залежності від положення перемикачів. Якщо всі перемикачі перебувають у положенні, яке вказано на схемі (див. рис. 5), окрім перемикачів *P* та *PG*, які переведені у нижнє положення, то

моделюється модернізована САК *L*. Якщо всі перемикачі перебувають у положенні, яке вказано на схемі, окрім перемикача *PG*, який переведено у нижнє положення, то моделюється модернізована САК *R*1. Якщо усі перемикачі у верхньому положенні (як це показано на схемі Рис. 5), то моделюється модернізована САК *R*2.

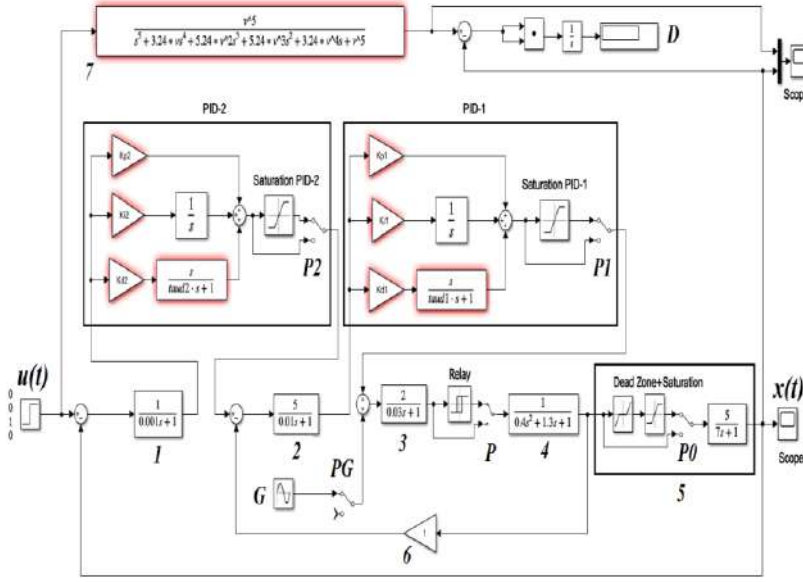


Рисунок 5. Структурна схема комп'ютерної математичної моделі модернізованої САК із підключеними еталонною моделлю 7 та алгоритмом обчислення кількісного значення критерія налаштування, яке відображається на дисплеї *D*: *G* – генератор вібрацій (коливань) для вібраційної лінеаризації

Особливістю модернізованих САК *R*1 та *R*2 порівняно із модернізованою САК *L* є наявність у їх складі суттєво нелінійного елементу релейного типу.

Суттєва нелінійність релейного типу притаманна деяким різновидам електронних перетворювачів, що використовуються в електроприводах вузлів рухомості маніпуляторів. Доцільність використання таких перетворювачів визначається відповідними конструктивно-технологічними особливостями електроприводу та маніпулятора.

Особливістю модернізованої САК *R*2 порівняно із *R*1 є застосування вібраційної лінеарізації для зменшення впливу нелінійного елементу релейного типу на ефективне функціонування САК.

Фізичний зміст впливу вібраційної лінеарізації на покращення статичної характеристики електронного перетворювача полягає у наступному. При наявності плавного корисного сигналу на вході ланки із суттєвою нелінійністю релейного типу (корисний сигнал та сигнал з генератора вібрацій *G* додаються) буде забезпечена плавна залежність середнього значення вихідного сигналу цієї ланки за період роботи генератора вібрацій *G* від середнього значення вхідного сигналу.

Етап 7.

Застосуємо чисельний метод Гаусса-Зейделя для оптимізації (налаштування) параметрів регуляторів PID-1 та PID-2. Скористаємося комп'ютерною математичною моделлю (див. рис. 5) із врахуванням різного набору суттєвих нелінійностей для кожного варіанту модернізації САК. Варіант модернізації САК позначається буквами *L* або *R*1, або *R*2. Положення перемикачів (див. рис. 5) для варіанта модернізованої САК: варіанту *L* відповідає верхнє положення перемикачів *P*0, *P*1, *P*2 та нижнє положення перемикачів *P* та *PG*; варіанту *R*1 відповідає верхнє положення перемикачів *P*, *P*0, *P*1, *P*2 та нижнє положення перемикача *PG* (тобто, вібраційна лінеарізація не застосовується); варіанту *R*2 відповідає верхнє положення усіх перемикачів, тобто застосовується вібраційна лінеарізація).

Критерій налаштування (оптимізації) залежить від шести параметрів регуляторів PID-1 та PID-2:

$$I(r) \to \min_{r \in \text{ОДР}},$$

де $r = [r_1, r_2, r_3, r_4, r_5, r_6]$ – вектор, що складений із параметрів регуляторів PID-1 та PID-2;

$$r_1 = k_{p1},\ r_2 = k_{i1},\ r_3 = k_{d1},\ r_4 = k_{p2},\ r_5 = k_{i2},\ r_6 = k_{d1}.$$

ОДР – область допустимих розв'язків для пошуку оптимальних значень параметрів регуляторів PID-1 та PID-2, яка задається фізичним змістом задачі (зокрема особливостями реалізації регуляторів PID-1 та PID-2 в конкретній задачі).

Критерій налаштування для відповідного варіанту модернізації САК будемо позначати буквами *L* або *R*1, або *R*2, тобто: $IL(r)$, $IR1(r)$, $IR2(r)$.

Параметри регуляторів taud1 та *taud*2 вважаємо незмінними параметрами. Ці параметри дорівнюють тим значенням, які були отримані при виконанні етапу 2 методики, тобто при первинному налаштуванні регуляторів PID-1 та PID-2: *taud*1 = 0.0150 та *taud*2 = 0.0225.

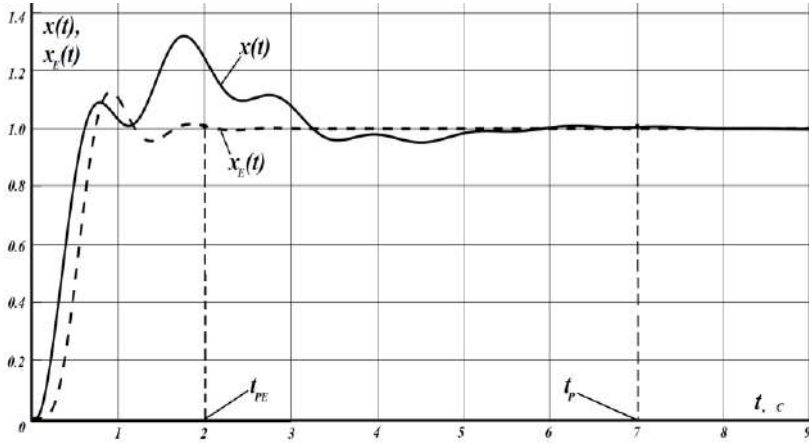
Ще раз підкреслимо, що для початку застосування чисельного методу Гаусса-Зейделя використовується перше наближення до оптимальних значень параметрів регуляторів PID-1 та PID-2, яке отримано на етапі 2 методом коливань Зіглера-Ніколса для *SISO LTI* математичної моделі модернізованих САК *L* та *R*1, *R*2 (усі перемикачі на рис. 2 перебувають у нижньому положенні).

4 Результати та їх обговорення*Результат першого наближення до оптимальних значень параметрів регуляторів PID-1 та PID-2*

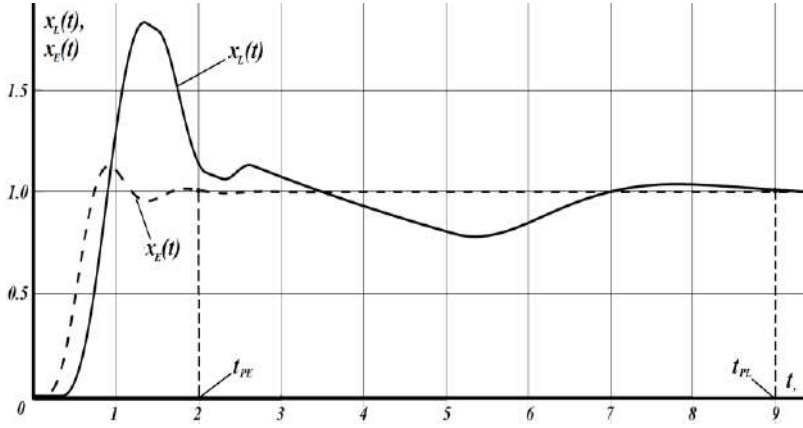
Порівняльне моделювання перехідних функцій еталонної моделі 7 та математичної моделі модернізованої САК *L*, *R*1 та *R*2 (див. рис. 5) для першого наближення

$$r1 = [Kp1 = 1.2,\ Ki1 = 3,\ Kd = 0.12,\ Kp2 = 2.1,\ Ki2 = 3.5,\ Kd2 = 0.315]$$

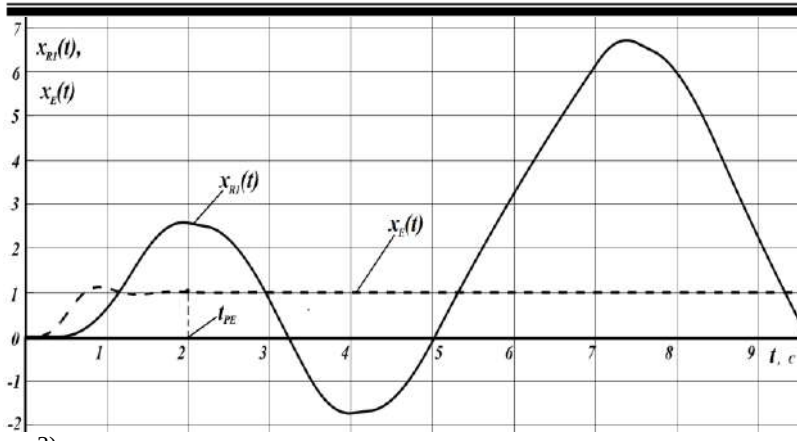
до оптимальних значень параметрів регуляторів PID-1 та PID-2, яке отримано на етапі 2, представлено на Рис. 6 (1, 2, 3, 4).



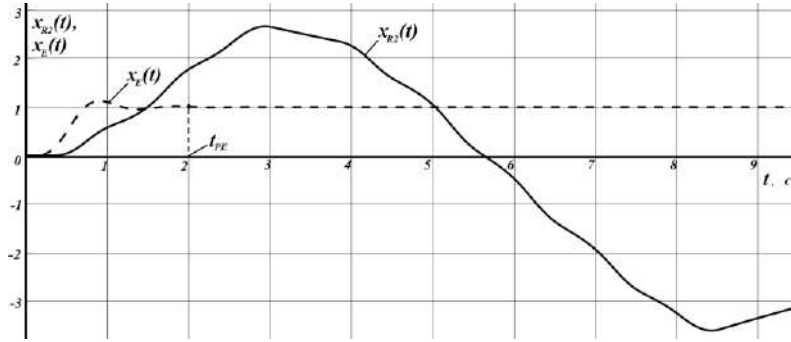
1)



2)



3)



4)

Рисунок 6. Перехідні функції еталонної моделі 7 $(x_E(t))$ та математичної моделі модернізованої САК *L* $(x_L(t))$ або $R1(x_{R1}(t))$, або $R2(x_{R2}(t))$ для першого наближення до оптимальних значень параметрів регуляторів PID-1 та PID-2, яке отримано на етапі 2

При використанні першого наближення *r1* до оптимальних значень параметрів регуляторів PID-1 та PID-2, які було отримано на етапі 2, на дисплеї *D* відобразилися наступні значення критерію налаштування:

$$IL(r1) = 0.5871,\ IR1(r1) = 61.21,\ IR2(r1) = 60.51$$

відповідно до комп'ютерних математичних моделей модернізованої САК *L* або *R*1, або *R*2 (рис.5).

Результат налаштування (оптимізації) параметрів регуляторів PID-1 та PID-2

Завдяки використанню методу Гаусса-Зейделя було налаштовано шість параметрів регуляторів PID-1 та PID-2 для кожної з модернізованих САК *L*, *R*1, *R*2. Результат налаштування параметрів позначено відповідно $rCL, rCR1, rCR2.$

В методі Гаусса-Зейделя використовується «покоординатний спуск»[7, 9]. В процесі оптимізації по кожній координаті (параметру регуляторів PID-1 та PID-2) було використано прямий інтерактивний метод покоординатного спуску (тобто оптимізація виконувалася практично «в ручну»). Один повний цикл складається із проходження по шести координатах, тобто від $r_1 = k_{p1}$ до $r_6 = k_{d2}$ включно (див. етап 7). Після оптимізації за окремою координатою запам'ятовувалася точка мінімуму по цій координаті. В процесі оптимізації не ставилося за мету досягти глобального мінімуму. Оскільки мова йде про інженерний метод налаштування, то ставилося за менту показати, що навіть без «автоматизації» (застосування спеціального програмного забезпечення для пошуку мінімуму алгоритмічно заданого критерію), а навіть у ручному режимі, можна досягти бажаного результату налаштування суттєво нелінійної САК. Підкреслимо, що в процесі оптимізації виконувалося візуальне порівняння якості перехідного процесу модернізованої САК із якістю перехідного процесу еталонної моделі за допомогою осцилографа Scope 2 (рис. 5).

За результатом виконання одного повного циклу в інтерактивному режимі покоординатного спуску (час одного повного циклу склав менше 30 хвилин) отримано наступні кількісні значення параметрів регуляторів PID-1, PID-2 та критерія налаштування для модернізованих САК:

$$L: CL = [Kp1 = 2.5;\ Ki1 = 3.2;\ Kd1 = 0.14;\ Kp2 = 2.1;\ Ki2 = 2;\ Kd2 = 0],$$
$$IL(rCL) = 0.06487;$$
$$R1: rCR1 = [Kp1 = 10;\ Ki1 = 4;\ Kd1 = 0.4;\ Kp2 = 2.1;\ Ki2 = 1.5;\ Kd2 = 0],$$
$$IR1(rCR1) = 0.2387;$$
$$R2: rCR2 = [Kp1 = 5;\ Ki1 = 4.5;\ Kd1 = 0;\ Kp2 = 2.1;\ Ki2 = 1;\ Kd2 = 0],$$
$$IR2(rCR2) = 0.2895.$$

Перехідні функції для комп'ютерних математичних моделей модернізованих САК *L*, *R*1 та *R*2 для вектора параметрів регуляторів PID-1 та PID-2, який дорівнює відповідному $rCL, rCR1, rCR2,$ представлені на рис. 7, 8, 9.

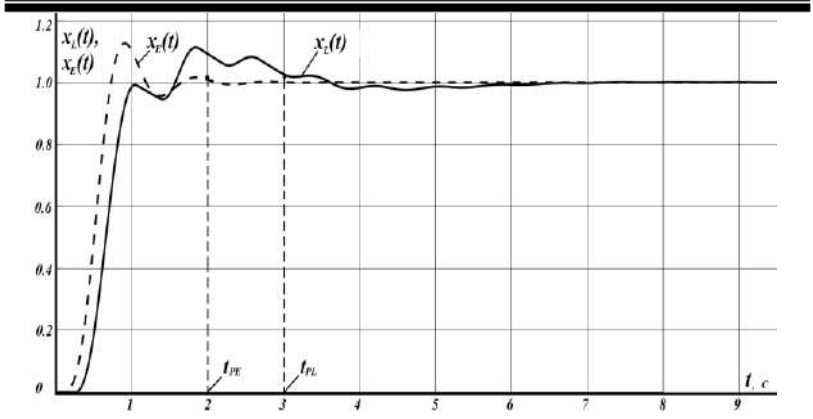


Рисунок 7. Перехідні функції еталонної моделі 7 $(x_E(t))$ та математичної моделі модернізованої САК *L* $(x_L(t))$ (див.рис. 5) для одного повного циклу наближення до оптимальних значень параметрів регуляторів PID-1 та PID-2: $t_{PL} = 3c$ – час тривалості перехідного процесу в модернізованій САК *L*

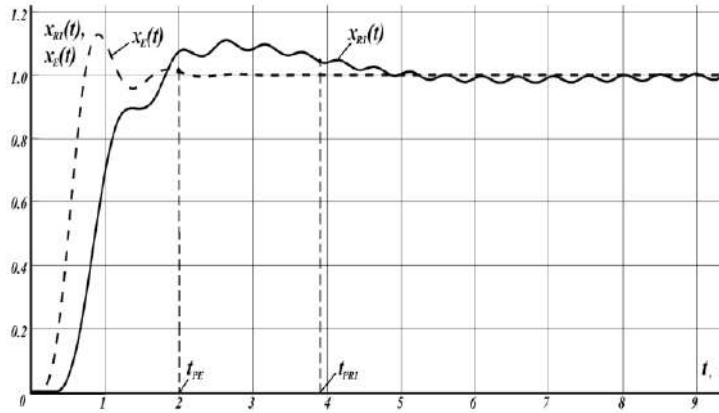


Рисунок 8. Перехідні функції еталонної моделі 7 $(x_E(t))$ та математичної моделі модернізованої САК *R*1 $(x_{R1}(t))$ (див. рис. 5) для одного повного циклу наближення до оптимальних значень параметрів регуляторів PID-1 та PID-2: $t_{PR1} = 3.8c$ – час тривалості перехідного процесу в модернізованій САК *R*1

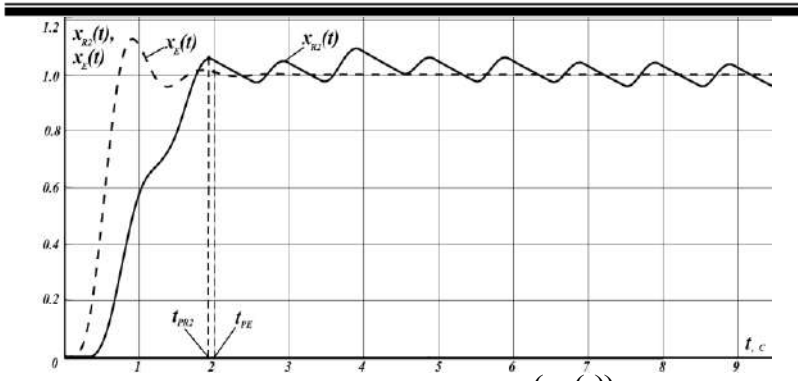


Рисунок 9. Перехідні функції еталонної моделі 7 $(x_E(t))$ та математичної моделі модернізованої САК *R*2 $(x_{R2}(t))$ (див. рис. 5) для одного повного циклу наближення до оптимальних значень параметрів регуляторів PID-1 та PID-2: $t_{PR2} = 1.9c$ – час установлення сталого режиму вібраційної лінеарізації в модернізованій САК *R*2; амплітуда коливань генератора вібрацій *G* (рис. 5) складала 20 відносних одиниць, а кутова частота 6.28 рад/с

Висновки

1. В статті розроблено інженерний метод налаштування (або переналаштування в процесі експлуатації) регуляторів електроприводів вузлів рухомості маніпулятора, який враховує наявність суттєвих нелінійностей. Цей метод дозволяє усунути причину виникнення ефекту автопружності (виникнення резонансних пружних вібрацій та автоколивань, що викликані динамічними властивостями приводу вузлів рухомості пружних гнучких маніпуляторів та суттєвими нелінійностями конструктивно-технологічного характеру або такими, що виникають в процесі експлуатації), тобто подавляє «первинні автоколивання».

2. Інженерна простота та зручність методу виражається в тому, що налаштування регуляторів електроприводів вузлів рухомості при виробництві маніпулятора або їх переналаштування в процесі експлуатації не потребує спеціалізованого наукового дослідження, а може бути виконано фахівцем із інженерним рівнем математичної підготовки в інтерактивному режимі за короткий час.

3. Позитивна відмінність запропонованого методу порівняно з методом, який реалізує пакет *Nonlinear Control Design* (*NCD Blockset*) системи комп'ютерної математики MATLAB полягає в тому, що суттєво спрощується спосіб завдання бажаної зміни в часі змінних, що характеризують стан системи автоматичного керування (часових обмежень). В розробленому методі ці обмеження задаються вибором добре відомих в інженерній практиці стандартних форм *SISO LTI* – моделей (еталонних моделей). При цьому

використовуються квадратичні інтегральні критерії наближення перехідного процесу на виході САК до перехідного процесу еталонної моделі, які теж найширше використовуються в інженерній практиці.

4. В комп'ютерних математичних моделях, які розглянуті в статті, було виконано моделювання типових суттєвих нелінійностей, які обумовлені конструктивно–технологічними особливостями механічної частини маніпуляторів аерокосмічних об'єктів та їх електроприводів: зона нечуттєвості, насичення, релейність.

5. Лише за один цикл оптимізації завдяки розробленому інженерному методу вдалося досягти порівняно із першим наближенням до бажаного (еталонного) перехідного процесу: покращення кількісного значення критерія налаштування приблизно на два порядки; зменшити час тривалості перехідного процесу фактично від нескінченності (якщо при першому наближенні система автоматичного керування була нестійка) приблизно до 3 - 4 с (без використання вібраційної лінеарізації); досягти такого значення перерегулюваня, яке складає від 25 % до 75 % еталонного перерегулювання.

6. Застосування добре відомого інженерного методу вібраційної лінеарізації у складі запропонованого інженерного методу дозволило додатково зменшити час тривалості перехідного процесу (час виходу на сталий режим вібраційної лінеарізації) з 3 – 4 с до 1.9 с при бажаному часі тривалості перехідного процесу 2 с.

7. В подальших дослідженнях буде розглянуто:

- застосування нелінійних регуляторів (нелінійних коригуючих пристроїв) електроприводів вузлів рухомості маніпуляторів із суттєвими нелінійностями;
- розширено вектор параметрів, що налаштовуються, з шести

$$(r = [r_1, r_2\ r_3, r_4, r_5, r_6]),$$

де $r_1 = k_{p1}, r_2 = k_{i1}, r_3 = k_{d1}, r_4 = k_{p2}, r_5 = k_{i2}, r_5 = k_{i2}, r_6 = k_{d1}$ (див. етап 7) до десяти компонент завдяки включенню до його складу параметрів математичної моделі реальної диференціюючої ланки *taud*1, *taud*2 та параметрів генератора вібрацій (рис.5), якими є амплітуда та кутова частота вібрацій.

8. Запропонований у статті інженерний метод налаштування регуляторів електроприводів вузлів рухомості маніпулятора із суттєвими нелінійностями дозволяє:

- отримати кількісні значення параметрів регулятора для САК із заданою структурою та відомим типом суттєвих нелінійностей на основі застосування відомих, апробованих на практиці та доступних програмних засобів сучасних систем комп'ютерної математики;
- бути повністю запрограмованим із використанням будь-якого чисельного методу оптимізації, що дозволить повністю автоматизувати параметричну оптимізацію САК із суттєвими нелінійностями.

Література

1. Tachinina, O., Lysenko, O., Alekseeva, I., et al. (2020). Mathematical modeling of motion of iron bird target node of security data management system sensors. *CEUR Workshop Proceedings*, 2711, 482–491.
2. Cabás Ormaechea, L. M. (2011). Humanoid robots. In *Advanced mechanics in robotic systems* (pp. 1–18). London. https://doi.org/10.1007/978-0-85729-588-0_1
3. Aswini, S., et al. (2023, December 1–3). Biomechanics-inspired control strategies for humanoid robots. *IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, Gautam Buddha Nagar, India. https://doi.org/10.1109/upcon59197.2023.10434421
4. Goodwin, G. C., Graebe, S. F., & Salgado, M. E. (2001). *Control system design* (1st ed.). Prentice Hall.
5. Dorf, R. C., & Bishop, R. H. (1998). *Modern control systems* (4th ed.). Prentice Hall.
6. Phillips, C. L., & Harbor, R. D. (2000). *Feedback control systems* (4th ed.). Prentice Hall.
7. Lysenko, O., Tachinina, O., Guida, O., et al. (2024). Generalized integro-differentiating controller form mechatronic devices of mobility nodes of humanoid robots. *CEUR Workshop Proceedings*, 3790, 87–98.
8. Solomentsev, O. V., Zaliskyi, M. Yu., Zuiev, O. V., & Asanov, M. M. (2013). Data processing in exploitation system of unmanned aerial vehicles radioelectronic equipment. *IEEE International Conference Actual Problems of Unmanned Air Vehicles Developments (APUAVD)*, Kyiv, Ukraine, 77–80. https://doi.org/10.1109/APUAVD.2013.6705288
9. Lysenko, O., Tachinina, O., Novikov, V., et al. (2023). Methodology of synthesizing digital regulators in precision electric drives for orientation and stabilization target tracking system of mobile robot's directional sensors. *CEUR Workshop Proceedings*, 3513, 51–63. https://ceur-ws.org/Vol-3513/
10. Solomentsev, O. V., Zaliskyi, M. Yu., Asanov, M. M., & Melkumyan, V. H. (2015). UAV operation system designing. *IEEE 3rd International Conference on Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, Kyiv, Ukraine, 95–98. https://doi.org/10.1109/APUAVD.2015.7346570

METHOD FOR ADJUSTING REGULATORS WITH SIGNIFICANT NONLINEARITIES

Dr.Sci. O. Lysenko[1][0000-0002-7276-9279], **Dr.Sci. O. Tatchynina**[2][0000-0001-7081-0576], **Ph.D. S. Ponomarenko**[1][0000-0001-5512-3778]
[1]National Technical University of Ukraine

"Igor Sikorsky Kyiv Polytechnic Institute", Ukraine,
[2]State University "Kyiv Aviation Institute", Ukraine
EMAIL:lysenko.a.i.1952@gmail.com, tachinina5@gmail.com, sol_@ukr.net

Abstract. *The article develops an engineering method for adjusting (or readjusting during operation) the controllers of electric drives of manipulator mobility units, which takes into account the presence of significant nonlinearities. This method prevents the occurrence of "primary self-oscillations" in the automatic control system of electric drives of manipulator mobility units, which stimulate the occurrence of resonant elastic vibrations and self-oscillations (self-oscillation effect). The proposed method allows not only to eliminate the cause of the autoelasticity effect, but also to do so at the engineering level of mastery of mathematical apparatus, computer mathematics systems, and programming skills. The manifestation of the autoelasticity effect is associated with the presence of factors such as: the dynamic properties of the drive of the mobility nodes; the elastic flexibility of manipulators; significant nonlinearities of a structural and technological nature or those that arise during operation in mechanical and electrical devices. The engineering simplicity and convenience of the method is expressed in the fact that the adjustment of the electric drive controllers of the mobility nodes during the manufacture of the manipulator or their readjustment during operation does not require specialized scientific research, but can be performed by a specialist with an engineering level of mathematical training in interactive mode in a short time.*

Keywords: *automatic control system, PID controller, significant nonlinearity, numerical optimization methods, computer mathematical model, directional antenna, manipulato*

UDC 004.8
DOI https://doi.org/10.36059/978-966-397-538-2-3

METHODOLOGY FOR IDENTIFYING POST-TRAUMATIC STRESS DISORDER INDICATORS IN TEXT DATA

Ph.D. O. Mazurets[1][0000-0002-8900-0650], **O. Ovcharuk**[2][0009-0008-3815-0035]
Khmelnytskyi National University, Ukraine
EMAIL:[1]exe.chong@gmail.com, [2]off4aruk@gmail.com

Abstract. *This study introduces the methodology for detecting indicators of post-traumatic stress disorder (PTSD) within textual data produced by users. Unlike traditional approaches, the proposed solution emphasizes two critical aspects: the refinement of contextual dependencies that are characteristic of PTSD, and the ability to more clearly distinguish PTSD-related signals from those of other mental*

health conditions. By concentrating on PTSD-specific linguistic patterns, the method provides a more accurate and contextually grounded analysis.The evaluation of the approach demonstrated robust performance, yielding Accuracy of 0.934, Precision of 0.948, F_1-score of 0.841, and AUC of 0.872. When compared against existing analogues, the proposed method showed consistent improvements across key metrics: Accuracy increased by 0.130%, the F_1-score improved by 0.031, and AUC rose by 0.132. A key technical innovation lies in the use of relative rather than absolute positional embeddings within the neural architecture. This enables the model to capture the nuanced relationships between words depending on their contextual displacement, thereby enhancing its sensitivity to subtle textual cues that often signal PTSD manifestations. In parallel, improved differentiation from other mental disorders was ensured by expanding the dataset to include samples reflecting manifestations of alternative psychological conditions, which were placed in an orthogonal reference category. This design decision helped the model to learn sharper boundaries between PTSD-specific content and symptoms indicative of other mental illnesses.

Keywords. *Post-traumatic stress disorder, NLP, neural network, text data*

Problem Statement

Post-traumatic stress disorder (PTSD) represents a severe and often long-lasting psychological condition that affects a significant portion of the global population, with prevalence rates reaching approximately 10% [1, 2]. The growing influence of digital technologies and the rapid expansion of social media platforms have created new avenues for studying PTSD manifestations through the lens of user-generated textual content [3]. This research direction gains particular urgency in the context of armed conflicts, large-scale natural disasters, and other traumatic events, all of which leave profound and lasting marks on human mental health [4, 5].

The analysis of user-generated materials, such as social media posts, blog entries, online forum discussions, and comment sections, offers unique opportunities to gain insight into the psychological state and emotional well-being of individuals affected by traumatic experiences [6, 7]. By applying advanced machine learning techniques and natural language processing (NLP) methods, it becomes possible to automatically detect subtle indicators of PTSD within text data [8, 9]. Such technological applications are not limited to identifying isolated cases of PTSD but also hold the potential to support large-scale monitoring, the design of preventive strategies, and the development of targeted assistance programs for populations at risk [10].

In contemporary society, which is increasingly exposed to wars, global pandemics, and socio-political crises, the timely identification of PTSD manifestations from textual data is of critical importance [11]. This type of research contributes not only to scientific understanding but also provides practical tools for psychologists, social workers, and healthcare professionals who are directly engaged in supporting vulnerable groups [12, 13]. The integration of automated

PTSD detection methods into professional practice thus offers a pathway toward more responsive and adaptive forms of psychological care [14].

The central objective of this paper is to present a methodology for identifying indicators of PTSD in user-generated text content. The proposed approach differentiates itself from existing solutions by placing special emphasis on context dependencies unique to PTSD, thereby reducing misclassification with other mental health disorders [15]. This is achieved by constructing a custom training dataset enriched with examples of alternative psychological conditions, which are placed in an orthogonal category to guide the model toward sharper diagnostic boundaries [16, 17]. Additionally, the methodology employs specialized positional embeddings designed to more accurately account for the location of words in a sequence, which significantly improves the accuracy and sensitivity of the neural network model [18].

The key contributions of this study can be outlined as follows. First, a novel method for identifying PTSD manifestations in text data has been developed, advancing the computational tools available for mental health analysis. Second, a specialized dataset was constructed to emphasize PTSD-related linguistic patterns while simultaneously reducing confusion with textual signals of other mental disorders by incorporating them into a separate category. Finally, the effectiveness of the methodology has been empirically validated, demonstrating not only its ability to detect PTSD with high accuracy but also to provide a visual interpretation of results, making the system more transparent and practically useful compared to existing analogues.

Analysis of Previous Studies

Over the years, the scientific community has made considerable efforts to design monitoring strategies aimed at detecting a wide spectrum of mental health conditions and high-risk behaviors [19]. Among these, depression, eating disorders, gambling addiction, and suicidal ideation have been of particular interest, as their timely identification enables the activation of preventive or mitigating interventions, and in severe cases, the initiation of clinical treatment [20]. In recent years, the prevalence of post-traumatic stress disorder has risen sharply, especially in the wake of the COVID-19 pandemic [21]. The global health crisis not only generated new traumatic experiences but also severely restricted access to traditional psychological support due to isolation measures and the limited availability of therapeutic interventions. These challenges have emphasized the urgent need for supplementary screening tools capable of enhancing PTSD identification and diagnosis within virtual or remote environments.

Recent advancements in artificial intelligence, and particularly in large language models (LLMs), have opened promising new opportunities for PTSD detection. Preliminary studies have suggested that ChatGPT possesses a degree of feasibility for mental health assessment, although its ability to provide accurate and reliable diagnoses remains under investigation [22]. One notable study compared the performance of ChatGPT with the text-embedding-ada-002 (ADA) model in the

detection of childbirth-related PTSD (CB-PTSD), a condition that affects millions of new mothers worldwide and frequently goes undiagnosed due to the lack of standardized screening procedures. The research, conducted on a cohort of 1,295 women within six months postpartum and recruited from hospitals, social media platforms, and professional organizations, evaluated the effectiveness of these models in classifying maternal birth narratives. Ground truth was established using the PTSD Checklist for DSM-5, ensuring methodological rigor. The ADA model, leveraging numerical vector representations for narrative classification, demonstrated superior performance (F_1 score = 0.81) compared to ChatGPT and six previously published text embedding models trained on mental health or clinical data. This finding suggests that ADA-based approaches may serve as a reliable framework for identifying CB-PTSD and could potentially be extended to the detection of other psychiatric conditions.

Further research has also investigated the role of language itself as a diagnostic biomarker for PTSD. One study [23] analyzed a dataset of 148 individuals who survived the Paris terrorist attacks of November 13, 2015. The participants, all of similar socioeconomic background and exposed to the same traumatic event, underwent interviews 5–11 months after the incident, which were later transcribed and subjected to analysis. An interdisciplinary methodology was employed, bringing together psychiatry, linguistics, and NLP to examine the connection between linguistic patterns and PTSD symptoms. The methodology comprised three stages. First, a clinical psychiatrist attempted to diagnose PTSD using only the transcribed interviews, achieving an area under the curve (AUC) of 0.72, which was comparable to the gold standard questionnaire (AUC $\approx$ 0.80). In the second stage, statistical and machine learning methods were applied to extract psycholinguistic features and evaluate their predictive power, achieving an AUC of 0.69. Finally, deep learning methods were tested in a hypothesis-free setting, with performance reaching an AUC of 0.64. The results confirmed that language carries clinically relevant information, while also demonstrating the limitations of current automated approaches. Importantly, the study controlled for potential confounding factors, established clear links between language and DSM-5 subsymptoms, and demonstrated how automated approaches can be complemented by qualitative analysis. Similar research conducted with military personnel deployed in Afghanistan reported a model AUC of 0.74 [24], further highlighting the potential of language-based methods for PTSD detection.

Unresolved Issues

The review of existing work in this domain indicates a persistent problem: relatively low detection accuracy. In most cases, the performance of current PTSD detection systems has not exceeded 81% accuracy, which is insufficient for reliable deployment in real-world psychological or clinical practice. This limitation underscores the need for more advanced artificial intelligence models that combine several essential properties. Such models should demonstrate a high capacity for generalization, enabling them to handle diverse expressions of trauma in text, as individuals often describe their experiences in highly variable linguistic forms.

Furthermore, they should be explicitly context-oriented, allowing the identification of subtle linguistic markers that signal the presence of PTSD while avoiding confusion with other mental health conditions. Finally, the models must support visual interpretability of their results, ensuring that the decision-making process can be explained and validated by human experts. Together, these requirements form the foundation for the methodology proposed in this paper, aimed at enhancing the accuracy, reliability, and transparency of PTSD detection from user-generated text data.

Objective of the Article

The object of research is the process of for identifying post-traumatic stress disorder indicators in text data. The subject of research is methods, algorithms, information technologies, models, and tools for identifying post-traumatic stress disorder indicators in text data.

Methodology for identifying post-traumatic stress disorder indicators

The methodology proposed for detecting manifestations of post-traumatic stress disorder in user-generated textual content is schematically illustrated in Figure 1. The general workflow demonstrates the transformation of input data, represented by raw text content, into structured analytical output through the use of a trained, context-oriented neural network of transformer architecture combined with a tokenizer. The outcome of this process is expressed as a probabilistic percentage indicating the likelihood of PTSD-related signals within user content.

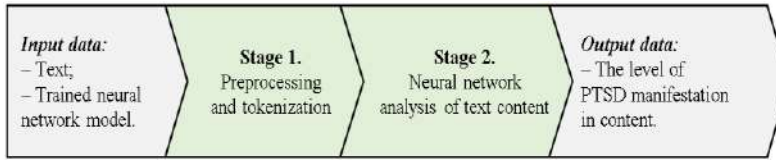


Figure 1. Approach to identifying post-traumatic stress disorder indicators

At the initial stage, preprocessing and tokenization of text data are performed. Each text entry is examined to ensure that it is non-empty and of sufficient length for analysis. Unlike conventional preprocessing pipelines, punctuation marks, emoticons, and other subtle textual features are deliberately retained, since they may contain valuable contextual information associated with PTSD indicators. Tokenization is then carried out using the same tokenizer that was employed during neural network training, ensuring compatibility and consistency between training and inference stages.

The second stage involves neural network–based analysis of PTSD indicators in the processed user content. For this purpose, a context-oriented model of transformer architecture is applied, specifically the DeBERTa model. The structure of this model, depicted in Figure 2, is organized into five primary component

groups: *Embedding*, *Encoder*, *Pooler*, *Classifier*, and *Dropout*. The *Embedding* layer incorporates word embeddings along with normalization operations. The *Encoder* section contains six stacked DeBERTaV3Layer blocks, each including attention, intermediate, and output modules. These layers form the core of text processing, capturing hierarchical representations of words and contexts. Unlike traditional transformer mechanisms, DeBERTa employs a disentangled attention mechanism, where semantic content and positional information are processed separately. This modification enhances the model's ability to capture nuanced interactions between words and contextual dependencies. Furthermore, instead of absolute positional encoding, DeBERTa applies relative positional displacement, enabling more robust modeling of long-range dependencies through variance estimation. The *Pooler* component aggregates contextual embeddings to produce compact summary representations, while the *Classifier* layer performs the final decision-making task. The *Dropout* layer ensures model regularization, reducing the risk of overfitting during training.

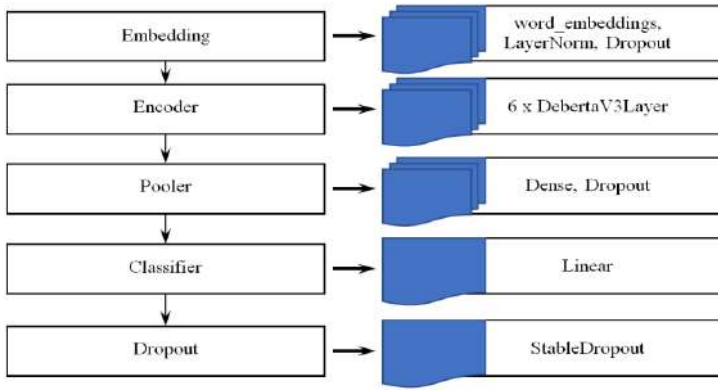


Figure 2. Architecture of DeBERTa model

The complete methodological scheme is shown in Figure 3. According to this workflow, the process begins with the preparation of training data. For effective PTSD detection, a composite dataset is constructed from multiple sources and subjected to balancing and preprocessing procedures. This ensures sufficient representation of positive and negative cases, thereby improving training stability and reducing bias. The dataset is then vectorized and divided into training and validation subsets in 80:20 ratio. At this stage, training hyperparameters such as batch size and number of epochs are also defined.

Subsequently, the transformer-based model undergoes retraining on the prepared dataset. Upon completion of the training process, the performance of the model is rigorously evaluated using a set of standard classification metrics,

including Accuracy, Precision, F_1-score, and AUC. Only models achieving satisfactory results – defined as performance above 80% across all metrics – are preserved. The trained model and tokenizer are then stored for integration into downstream applications.

The final step of the methodology addresses the interpretability of the model's decisions. To this end, explainability frameworks designed for transformer-based architectures are employed. These allow for the visualization and explanation of how specific words, phrases, or contextual structures contributed to the classification decision.

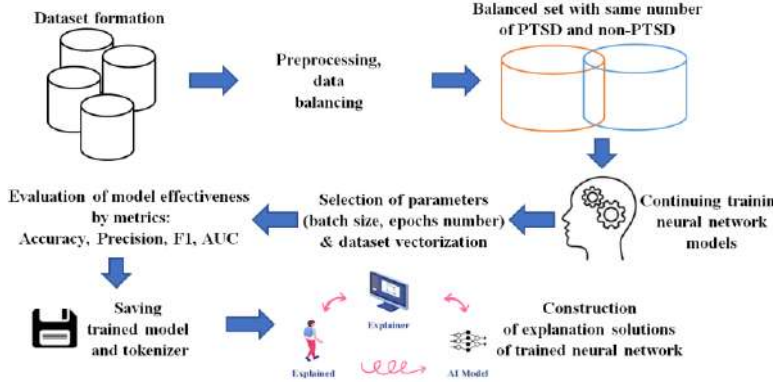


Figure 3. Steps for identifying of PTSD stress disorder indicators in text data

The methodological contributions are multifold. By implementing relative positional shifts instead of absolute encodings, the model achieves enhanced sensitivity to PTSD-specific contextual dependencies. Dispersion estimation techniques further improve the capacity to analyze relationships at varying textual distances. Additionally, the disentangled treatment of semantic and positional attention components enables the model to capture complex, layered dependencies that are characteristic of PTSD-related linguistic patterns. Finally, the inclusion of textual examples of other mental health conditions in the dataset strengthens the separation between PTSD and non-PTSD categories, reducing misclassification and improving overall robustness of detection.

Experiment

The scheme for constructing the dataset is presented in Figure 4. Since publicly available datasets with a sufficient number of records for training a neural network are lacking, a composite dataset was created by combining data from the "Human Stress Prediction" and "Aya PTSD" datasets.

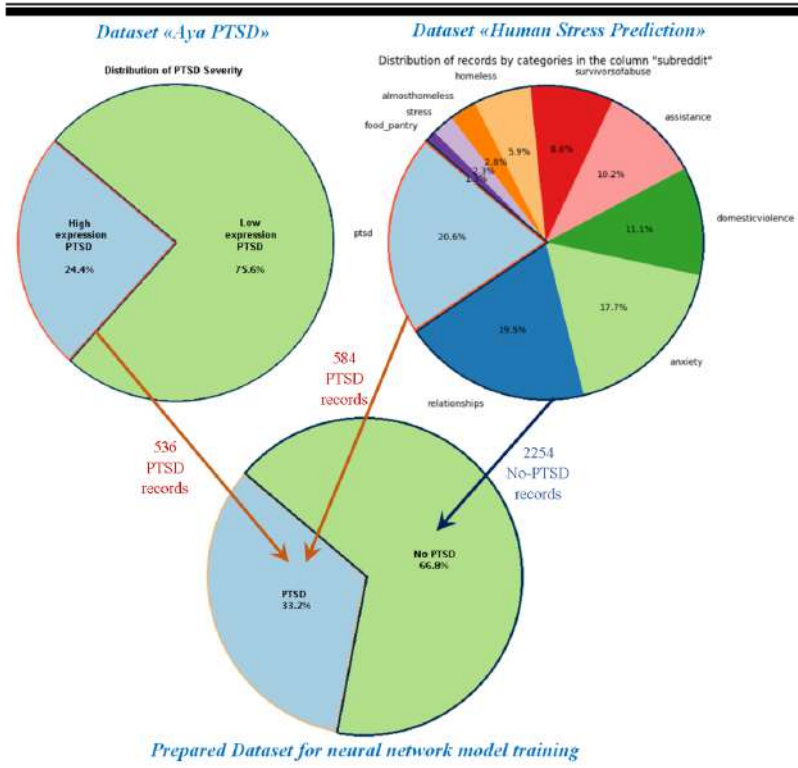


Figure 4. Dataset formation scheme for neural network training

The “Human Stress Prediction” dataset [25] contains user posts collected from subreddits focused on mental health discussions. It encompasses a wide range of issues people share regarding their personal experiences. The dataset includes fields such as “subreddit,” “post_id,” “sentence_range,” “text,” “label,” “confidence,” and “social_timestamp,” which form the structure of the “Stress.csv” file. For this study, only the entries labeled “PTSD” in the “subreddit” column were selected, which correspond to posts explicitly associated with post-traumatic stress disorder. All other entries, not marked with PTSD, were used as the negative class. Within this dataset, there are 584 posts labeled as “PTSD” and 2,254 posts without such indications.

Although the negative class includes not only records from healthy individuals but also posts linked to various other psychological disorders, this diversity is beneficial for learning contextual dependencies unique to PTSD. At the same time, it may lead to a slight reduction in training accuracy due to overlap between mental conditions.

The dataset derived from this source appeared to be imbalanced, with the PTSD class being nearly four times smaller than its counterpart. To address this, additional data were integrated from the "Aya PTSD" dataset [26], available via Kaggle. This dataset specifically targets PTSD and provides a richer description, including attributes such as clinical assessments, socio-demographic information, and related textual data ("ID," "PTSD Severity," "label," "text," "file_path," and "response"). For the purposes of this research, only entries with "PTSD Severity" values exceeding 50% were selected. Incorporating these records expanded the PTSD class to 1,120 entries. As a result, a balanced dataset of 3,374 samples was formed, with 1,120 labeled as "PTSD" and 2,254 labeled as "No PTSD." For training purposes, 2,200 records were utilized – 1,100 belonging to each category.

To evaluate the proposed method for analyzing PTSD manifestations in textual content, dedicated software was developed and implemented as a Notebook in the Google Colab cloud environment. As the pretrained backbone, the "microsoft/deberta-v3-small" transformer model from the HuggingFace library was adopted. The DeBERTaV3 model builds on the original DeBERTa architecture, replacing the masked language modeling (MLM) task with replaced token detection (RTD), which provides more efficient pretraining. The model was fine-tuned on the constructed dataset for 3 epochs. Performance assessment was carried out on a validation set. The trained model achieved Accuracy of 0.934, Precision of 0.948, F_1-score of 0.841, and AUC of 0.872. The ROC curve, presented in Figure 5, illustrates the classification quality. The Precision value of 0.948 indicates that among all samples classified as positive (PTSD), 94.8% were indeed correct, which is particularly significant for PTSD detection since minimizing false positives prevents unnecessary interventions. The F_1-score of 0.841 demonstrates a balanced trade-off between Precision and Recall, confirming the model's reliability in identifying true PTSD cases. The ROC curve further validates the effectiveness of the classifier by comparing the true positive rate and false positive rate at various thresholds, with an AUC of 0.872 highlighting the model's ability to correctly distinguish PTSD cases with a low incidence of false alarms.

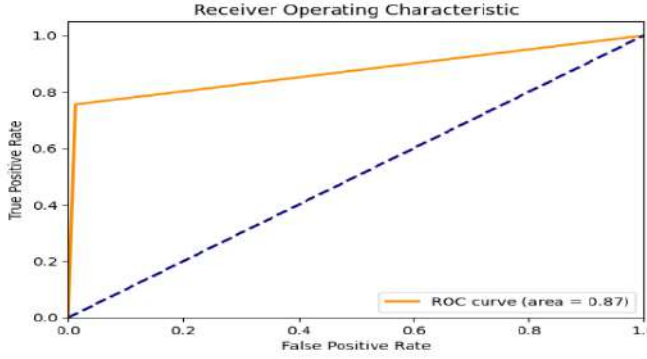


Figure 5. ROC curve for neural network performance evaluation

Additionally, the interpretability of the classification process was examined by employing the Transformers Interpret tool, designed for the transformers library. This approach allowed for visualization and explanation of the neural network's decision-making, providing a more transparent understanding of how PTSD-related textual cues are captured and analyzed.

Results and discussion

The F_1-score, which was previously reported at 0.81 in [22], increased to 0.841 in this study. The diagnostic AUC of the machine learning model described in [23] was 0.69, and the developed approach in [24] reached 0.74. In contrast, the current method achieved an AUC of 0.872, marking a substantial improvement over existing solutions. Overall, compared with established analogues, the efficiency of PTSD detection increased noticeably: Accuracy improved by 13 percentage points (from 80.4% to 93.4%), the F_1-score rose by 0.031 (from 0.810 to 0.841), and AUC improved by 0.132 (from 0.740 to 0.872). These enhancements demonstrate the robustness of the developed method, which effectively reduces confusion with other mental health conditions. This robustness stems from the fact that the negative class in the dataset was not limited to psychologically healthy individuals, but also included cases of other psychological disorders, thereby strengthening the model's ability to distinguish PTSD-specific patterns.Despite these promising outcomes, further optimization remains possible. The metrics suggest that additional epochs could improve the results, as the training loss continued to decline, indicating potential for better generalization. However, due to computational resource constraints and the 12-hour session limit in Google Colab, the training process was restricted to 10 hours. The confusion matrix obtained for the validation set is shown in Figure 6. According to these results, only 10 out of 220 PTSD-labeled entries were incorrectly classified as "No PTSD." Conversely, 24 out of 220 non-PTSD records were mistakenly identified as PTSD. Although false positives were observed, these values remain competitive when compared to prior studies, confirming the effectiveness of the developed model.

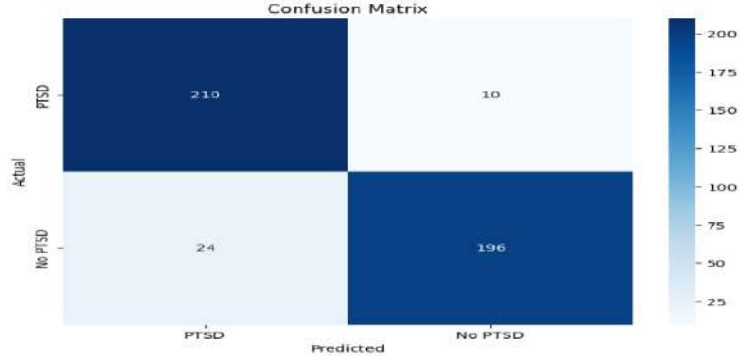


Figure 6. Confusion matrix in neural network detection of PTSD manifestations

Beyond classification, the method also provides interpretability of decisions, enabling a deeper understanding of model reasoning. An example is illustrated in Figure 7, where the input text was: *"He said he had not felt that way before, suggested I go rest and so I decide to look up feelings of doom in hopes of maybe getting sucked into some rabbit hole of ludicrous conspiracy, a stupid are you psychic test or new age b.s., something I could even laugh at down the road. No, I ended up reading that this sense of doom can be indicative of various health ailments; one of which I am prone to.. So on top of my doom to my gloom..I am now f'n worried about my heart. I do happen to have a physical in 48 hours."* This text was classified as PTSD-related, as it contains strong indicators such as "sense of doom" and the phrase "doom to my gloom," reflecting heightened anxiety and stress. Figure 8 highlights that tokens such as *"doom"* and *"gloom"* had the greatest influence on the model's decision, while additional words like *"conspiracy," "psychic,"* and *"rabbit"* also contributed significantly.

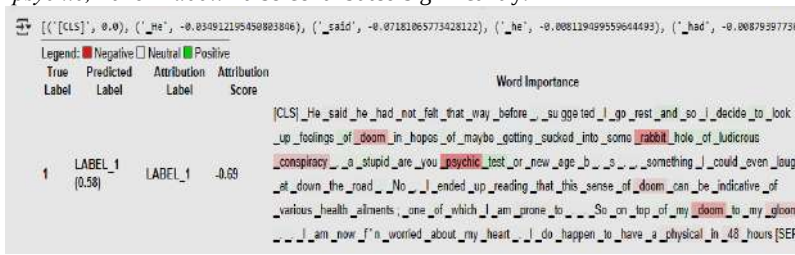


Figure 7. Interpretation of transformer model solution

In parallel, the study explored how the model distributes attention across different tokens to capture contextual dependencies. The visualization tool BertViz was applied to illustrate attention weights across all layers and heads of the transformer. Figure 8 shows an example where brighter colors represent stronger attention contributions to the final decision. Each attention head forms a distinct representation of token-to-token dependencies, visualized through arrows whose thickness and color denote the intensity and direction of attention.

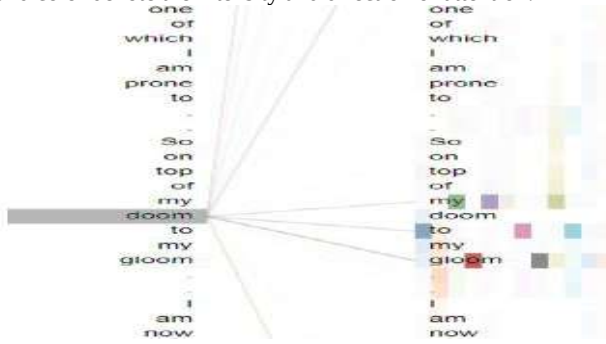


Figure 8. Visualization of attention weights using the "BertViz" model interpreter

Thus, the combined use of *Transformers Interpret* and *BertViz* not only enabled accurate PTSD detection in user-generated content but also allowed detailed interpretation of the neural network's internal decision-making process. This interpretability adds significant value, as it makes the model's predictions more transparent and reliable for practical applications in mental health diagnostics.

Conclusions and Prospects

The conducted research has resulted in the development of a methodology for detecting indicators of post-traumatic stress disorder in user-generated text data. Unlike traditional approaches, the proposed method emphasizes deeper consideration of PTSD-specific contextual dependencies and enhances the ability to separate these patterns from manifestations of other mental health conditions.

The improved focus on contextual dependencies was achieved through several architectural and algorithmic solutions. In particular, instead of the conventional single attention mechanism used in transformer networks, the proposed approach employs separate attention components for words and positional embeddings. This modification increases the granularity of interaction captured between lexical units and their context, which is critical for identifying subtle linguistic markers of PTSD. Additionally, the model implements relative positional displacement rather than absolute, which allows it to capture long-range semantic dependencies more effectively by estimating variance across different distances. These adjustments collectively improved the accuracy and robustness of PTSD-specific feature recognition.

To further reduce false associations with other psychological conditions, the training dataset was deliberately structured so that the non-PTSD category contained not only samples from psychologically healthy individuals but also examples representing other mental illnesses. This strategy created a more orthogonal dataset structure, improving the model's ability to distinguish PTSD-specific linguistic cues. As a result, a custom training dataset consisting of 3,374 records was formed, which provided a solid foundation for model learning.

The developed solution transforms raw text input into structured output, presenting the probability of PTSD manifestation in percentage terms. Importantly, the methodology supports interpretability of the neural network's decisions, enabling researchers and practitioners to trace which textual elements contributed most to classification outcomes. This transparency significantly strengthens the potential application of the method in clinical and preventive scenarios.

Experimental evaluation demonstrated that the trained transformer-based neural network achieved Accuracy of 0.934, Precision of 0.948, F_1-score of 0.841, and AUC of 0.872. Compared with existing studies, this represents a marked improvement: Accuracy increased by 13.0% (from 80.4% to 93.4%), the F_1-score rose by 0.031 (from 0.810 to 0.841), and AUC improved by 0.132 (from 0.740 to 0.872). These results confirm the effectiveness of the proposed methodology in reliably identifying PTSD manifestations in text.

Future research directions include expanding the dataset with additional labeled examples to enrich the spectrum of PTSD-related linguistic markers and further improve classification reliability. Another promising direction involves increasing the number of training epochs, since the observed decrease in loss during experiments indicates that the model has not yet reached its full generalization capacity. Continued work in these areas is expected to yield even higher performance metrics and enhance the interpretability of the model's decisions, thereby contributing to more accurate and transparent detection of PTSD manifestations in textual content.

References

1. Malgaroli, M., & Schultebraucks, K. (2020). Artificial intelligence and posttraumatic stress disorder (PTSD). *European Psychologist, 25*(4), 272–282.

2. Mazurets, O., & Ovcharuk, O. (2024, November 21). Efficiency research of method for detecting mental disorders by analysis of user content. In *Proceedings of the 11th International Conference on Information Technology Implementation (Satellite)* (pp. 46–47). Kyiv, Ukraine.

3. Bartal, A., Jagodnik, K. M., Chan, M. S. J., Babu, M. M. S., & Dekel, S. (2022). Identifying women with post-delivery posttraumatic stress disorder using natural language processing of personal childbirth narratives. *American Journal of Obstetrics & Gynecology MFM, 100834*.

4. Molchanova, M., Didur, V., Sobko, O., & Mazurets, O. (2025). Detection of web propaganda patterns by transformer neural networks: Improving efficiency via dataset balancing. *CEUR Workshop Proceedings, 3988,* 112–126.

5. Kuhn, E., & Owen, J. E. (2020). Advances in PTSD treatment delivery: The role of digital technology in PTSD treatment. *Current Treatment Options in Psychiatry, 7*(2), 88–102.

6. Mazurets, O., Sobko, O., Dydo, R., Zalutska, O., & Molchanova, M. (2025). Transformer-based multilabel classification for identifying hidden psychological conditions in online posts. *CEUR Workshop Proceedings, 4004,* 347–361.

7. Murava, V., Zalutska, O., Didur, V., & Mazurets, O. (2025, June 25–27). Software architecture of information system for exchanging LLM thematic prompts. In *Proceedings of the IV International Scientific and Practical Conference Global Trends in the Development of Information Technology and Science* (pp. 121–127). Stockholm, Sweden.

8. Tymofiiev, I., Mazurets, O., Hardysh, D., & Molchanova, M. (2024, November 6–8). Neural network dual architecture for depression detection using cloud services. In *Proceedings of the XLVI International Scientific and Practical Conference Scientific Research in the Era of Digital Technologies: Challenges and Opportunities* (pp. 84–88). Barcelona, Spain.

9. Yurchenko, D., Mazurets, O., Didur, V., & Molchanova, M. (2024, November 13–15). Approach to using cloud services for visual analytics of neural network analysis of texts emotional tonality. In *Proceedings of the XLVII*

International Scientific and Practical Conference The Future of Scientific Discoveries: New Trends and Technologies (pp. 108–113). Marseille, France.

10. Park, A. H., et al. (2023). Machine learning models predict PTSD severity and functional impairment: A personalized medicine approach for uncovering complex associations among heterogeneous symptom profiles. *Psychological Trauma: Theory, Research, Practice, and Policy.*

11. Mazurets, O., & Vit, R. (2024, November 21). Practical application of method of thematic classification of text information using LDA. In *Proceedings of the 11th International Conference on Information Technology Implementation (Satellite)* (pp. 151–152). Kyiv, Ukraine.

12. Hladun, O., Mazurets, O., Molchanova, M., & Sobko, O. (2024, November 25–27). Real time detection of the person emotion state using neural network. In *Proceedings of the II International Scientific and Practical Conference Scientific Research: Modern Innovations and Future Perspectives* (pp. 119–123). Montreal, Canada.

13. Andrushchenko, D., Klimenko, V., & Mazurets, O. (2025, May 28–30). Vector databases search for adaptive filtering of scientific articles. In *Proceedings of the XXII International Scientific and Practical Conference Scientific Trends in the Development of Modern Technology* (pp. 189–195). Krakow, Poland.

14. Mazurets, O., Tymofiiev, I., & Dydo, R. (2024, November 15). Approach for using neural network BERT-GPT2 dual transformer architecture for detecting persons depressive state. In *Proceedings of the VI International Scientific and Practical Conference Ricerche scientifiche e metodi della loro realizzazione* (pp. 147–151). Bologna, Italy.

15. Krak, I., Ovcharuk, O., Mazurets, O., Molchanova, M., Sobko, O., Tyschenko, O., & Barmak, O. (2025). Method for post-traumatic stress disorder manifestation analyzing in text content. *CEUR Workshop Proceedings, 3976,* 206–218.

16. Mazurets, O., Molchanova, M., Klimenko, V., & Prosvitliuk, M. (2024, June 12–14). Practice implementation of neural network model BART-Large-CNN for text annotation. In *Proceedings of the XXVII International Scientific and Practical Conference Prospects of Scientific Research in the Conditions of the Modern World* (pp. 97–102). Rotterdam, Netherlands.

17. Sobko, O., Mazurets, O., Didur, V., & Chervonchuk, I. (2024, June 5–7). Recurrent neural network model architecture for detecting a tendency to atypical behavior of individuals by text posts. In *Proceedings of the XXVI International Scientific and Practical Conference Theoretical and Practical Aspects of Modern Research* (pp. 113–117). Ottawa, Canada.

18. Mazurets, O., Sobko, O., Molchanova, M., Zalutska, O., & Yurchak, A. (2024, May 31). Practical implementation of neural network method for stress features detection by social internet networks posts. In *Proceedings of the II International Scientific Theoretical Conference Global Science: Prospects and Innovations* (pp. 160–167). Berlin, Germany.

19. Browning, L., Rashid, I., & Javanbakht, A. (2024). The current state of digital technologies for the treatment and management of PTSD. *A Look into the Future of Psychiatry*.

20. Blazhuk, V., Mazurets, O., & Zalutska, O. (2024, October 23–25). An approach to using the mBERT deep learning neural network model for identifying emotional components and communication intentions. In *Proceedings of the XLIV International Scientific and Practical Conference The Impact of Scientific Research on the Development of the Modern World* (pp. 79–84). Dubrovnik, Croatia.

21. Krak, I., Mazurets, O., Ovcharuk, O., Molchanova, M., Barmak, O., & Azarova, L. (2025). Transformer-based multilabel classification for identifying hidden psychological conditions in online posts. *CEUR Workshop Proceedings, 4004*, 86–97.

22. Bartal, A., Jagodnik, K. M., Chan, S. J., & Dekel, S. (2024). AI and narrative embeddings detect PTSD following childbirth via birth stories. *Scientific Reports, 14*(1).

23. Quillivic, R., et al. (2024). Interdisciplinary approach to identify language markers for post-traumatic stress disorder using machine learning and deep learning. *Scientific Reports, 14*(1).

24. Papini, S., et al. (2023). Development and validation of a machine learning prediction model of posttraumatic stress disorder after military deployment. *JAMA Network Open, 6*(6), e2321273.

25. Kaggle. (2025, September 15). Human stress prediction [Data set]. Kaggle. https://www.kaggle.com/datasets/kreeshrajani/human-stress-prediction/data

26. Kaggle. (2025, September 15). Aya PTSD [Data set]. Kaggle. https://www.kaggle.com/datasets/abdelrahmanahmed3/aya-ptsd

МЕТОДОЛОГІЯ ВИЯВЛЕННЯ ОЗНАК ПОСТТРАВМАТИЧНОГО СТРЕСОВОГО РОЗЛАДУ В ТЕКСТОВИХ ДАНИХ

Ph.D. О. Мазурець [1][0000-0002-8900-0650], **О. Овчарук** [2][0009-0008-3815-0035]
Хмельницький національний університет, Україна
EMAIL: [1]exe.chong@gmail.com, [2]off4aruk@gmail.com

Анотація. *Дослідження присвячене розробці методології виявлення ознак посттравматичного стресового розладу (ПТСР) у текстових даних, що створюються користувачами. На відміну від традиційних підходів, запропоноване рішення підкреслює два критичні аспекти: уточнення контекстних залежностей, характерних для ПТСР, та здатність чіткіше розрізняти сигнали, пов'язані з ПТСР, від сигналів інших психічних розладів. Зосереджуючись на специфічних для ПТСР лінгвістичних шаблонах, метод забезпечує більш точний та контекстуально обґрунтований аналіз.*

Оцінка підходу продемонструвала надійну продуктивність, забезпечивши точність 0,934, повноту 0,948, F_1 0,841 та AUC 0,872. У порівнянні з існуючими аналогами, запропонований метод показав послідовні покращення за ключовими показниками: точність збільшилася на 0,130%, F_1-оцінка покращилася на 0,031, а AUC зросла на 0,132.

Ключова технічна інновація полягає у використанні відносних, а не абсолютних позиційних вбудовувань у нейронну архітектуру. Це дозволяє моделі фіксувати нюансовані зв'язки між словами залежно від їх контекстного зміщення, тим самим підвищуючи її чутливість до тонких текстових сигналів, які часто сигналізують про прояви ПТСР. Також, покращена диференціація від інших психічних розладів була забезпечена шляхом розширення набору даних, щоб включити зразки, що відображають прояви альтернативних психологічних станів, які були розміщені в ортогональній референтній категорії. Це рішення допомогло моделі вивчити чіткіші межі між специфічним для ПТСР контентом та симптомами, що вказують на інші психічні захворювання.

Ключові слова. *Посттравматичний стресовий розлад, NLP, нейромережа, текстові дані*

УДК 004.056.5
DOI https://doi.org/10.36059/978-966-397-538-2-4

НОРМОВАНА ВІДОКРЕМЛЕНІСТЬ МАКСИМАЛЬНОГО СИНГУЛЯРНОГО ЧИСЛА БЛОКУ ЦИФРОВОГО ЗОБРАЖЕННЯ ЯК ПОКАЗНИК ДОЦІЛЬНОСТІ ЙОГО СТЕГАНОПЕРЕТВОРЕННЯ

Dr.Sci. І. Бобок[1][0000-0003-4548-0709], **Dr.Sci. А. Кобозєва**[2][0000-0001-7888-0499]

[1]Національний університет «Одеська політехніка», Україна,
[2]Одеський національний морський університет, Україна,
EMAIL: [1]onu_metal@ukr.net, [2]alla_kobozeva@ukr.net

Анотація. *Одним із найбільш перспективних та ефективних напрямів захисту інформації сьогодні є стеганографія. Серед низки вимог до стеганосистеми одною з основних є забезпечення надійності сприйняття стеганоповідомлення. Однак багато існуючих стеганометодів, зокрема стійких до атак проти вбудованого повідомлення, не в змозі систематично забезпечувати ацю вимогу та не розраховані на роботу з випадковим контейнером, який на практиці є найчастіше застосовуваним. Метою роботи є підвищення ефективності стеганосистеми при використанні довільного стеганометоду і випадкового контейнеру шляхом розробки методу вибору блоків контейнера, в якості якого використовується цифрове*

зображення, для вбудови в них додаткової інформації. Критерієм ефективності виступає надійність сприйняття формованого стеганоповідомлення, що кількісно оцінюється за допомогою різницевого показника - пікового відношення «сигнал-шум». Мета була досягнута шляхом дослідження властивостей формальних параметрів блоків цифрового зображення – сингулярних чисел з урахуванням відповідності сингулярних трійок і частотних складових блоку. Найбільш важливим результатом роботи є теоретично обґрунтоване визначення параметра блоку, що дає інтегральну кількісну характеристику відносного розподілу його частотних складових – нормованої відокремленості максимального сингулярного числа блоку. Практична значимість отриманих результатів полягає в розробці методу вибору блоків зображення і методу визначення небажаних для використання в якості контейнера зображень, алгоритмічні реалізації яких дозволили підвищити ефективність стеганосистеми при їх застосуванні.

***Ключові слова**: стеганографія, цифрове зображення, надійність сприйняття стеганоповідомлення, вибір блоку контейнера, вибір контейнера, нормована відокремленість сингулярного числа.*

Вступ

Стеганографічна система сьогодні є одною з напотужніших систем захисту інформації, яка дає можливість приховати сам факт наявності секретного повідомлення шляхом його попереднього кодування, в результаті чого отримується додаткова інформація (ДІ), і вбудови в деякий інформаційний об’єкт – контейнер, що не привертає уваги, в якості якого в сучасних умовах найчастіше використовується цифрове зображення (ЦЗ) [1], що і розглядається в роботі. Результатом вбудови ДІ є стеганоповідомлення (СП).

До стеганографічної системи висувається низка вимог, серед яких: стійкість до виявлення (стеганоаналізу [2,3]); стійкість до атак проти вбудованого повідомлення [4] – збурних дій (ЗД), які змінюють матрицю СП; забезпечення надійності сприйняття СП (зміни, що вносяться при вбудові ДІ не повинні бути помітними); значна пропускна спроможність прихованого каналу зв’язку; можливість забезпечення автентифікації [5]; незначної обчислювальної складності тощо [6]. Одночасне задоволення всіх цих вимог викликає труднощі, адже задоволення деяких з них вимагає виконання взаємовиключних умов. Так відомо [7], що модифікація високочастотних складових веде до найменших візуальних спотворень вихідного зображення. Однак застосування даного факту на практиці в області стеганографії не завжди виправдано в силу того, що при використанні для вбудови ДІ високочастотної складової зображення одержуване СП виявиться чутливим до будь-яких атак проти вбудованого повідомлення, що змушує шукати інші, компромісні можливості забезпечення необхідних властивостей. Ще одним прикладом взаємовиключних вимог на практиці можна вважати забезпечення якнайбільшої пропускної спроможності прихованого каналу і надійності сприйняття СП, оскільки зі збільшенням пропускної спроможності, як

правило, ймовірність непомітності спотворень в результаті стеганоперетворення (СПр) зменшується [1].

Враховуючи вищезазначене, на практиці до уваги беруться конкретні, передбачувані, умови використання конкретної стеганографічної системи, з оглядом на які формуються пріоритети вимог до стеганосистеми [8].

На сьогодні актуальною, такою, що не має остаточного рішення, залишається задача систематичного забезпечення надійності сприйняття СП, зокрема при використанні стеганографічних методів, стійких до атак проти вбудованого повідомлення. Відомо [9], що для того, щоб існувала потенційна можливість декодування ДІ зі СП, що піддалось ЗД, збурення контейнера в результаті вбудови ДІ повинно бути не менше, ніж збурення, що є результатом атаки на СП. Враховуючи це, очевидно, що для забезпечення стійкості СП до значних ЗД спотворення контейнера при СПр, як правило, будуть більшими, ніж коли задача забезпечення стійкості до атак проти вбудованого повідомлення не є критичною. Це означає, що для стійких стеганоалгоритмів задача забезпечення надійності сприйняття СП є більш актуальною і потребує більших зусиль для її рішення, ніж для стеганоалгоритмів, від яких стійкість не вимагається. Саме цій задачі і присвячена дана робота.

Аналіз проблеми

Питання розробки достатніх умов забезпечення надійності сприйняття, стійкості СП до ЗД не раз піднімалися вченими-стеганографами. Так в [9] на основі загального підходу до аналізу стану інформаційних систем (ЗПАІС) було доведено, що з метою забезпечення значної ймовірності надійності сприйняття СП вбудову ДІ в ЦЗ-контейнер доцільно робити таким чином, щоб збурені в результаті СПр сингулярні вектори (СНВ) матриці контейнера відповідали малим за значенням сингулярним числам (СНЧ); збурення будь-яких СНЧ були незначними. Для забезпечення стійкості стеганосистеми до атак проти вбудованого повідомлення достатньо, щоб в процесі вбудови ДІ збурення отримали нечутливі до ЗД параметри повного набору матриці контейнера, який складається з СНВ і СНЧ, що визначаються однозначно для матриці ЦЗ шляхом застосування нормального сингулярного розкладання [10]. Отримані достатні умови є загальними, можуть бути застосованими незалежно від того, в якій області ЦЗ-контейнера (просторовій, перетворення) відбувається СПр, але їх одночасне застосування є утрудненим, оскільки, зокрема, нечутливі до збурних дій СНВ відповідають найбільшим за значенням СНЧ, що суперечить достатній умові надійності сприйняття. Аналогічне зауваження може бути висунутим до достатніх умов, отриманих в [11] в області перетворення Уолша-Адамара шляхом використання ЗПАІС, на основі якого були встановлені зв’язки між трансформантами Уолша-Адамара, коефіцієнтами дискретного косинусного перетворення і складовими сингулярного розкладання відповідних матриць.

Однак усі достатні умови вимагають їхнього врахування під час розробки стеганографического методу. Для наявних методів задоволення існуючим достатнім умовам вимагає, як правило, їхньої певної модифікації або взагалі не може бути реалізованим, хоча надійність сприйняття, стійкість до атак такими методами може і не забезпечуватися систематично, як, наприклад, методом, запропонованим в [4], який до цього моменту залишається одним з найбільш стійких до атаки стиском з малими коефіцієнтами якості, але не забезпечує систематично надійність сприйняття формованого СП.

Модифікація методу не є тривіальним питанням. Вона вимагає змін математичного базису, змін в практичній реалізації, при цьому будь-яка модифікація методу, спрямована на поліпшення конкретної властивості, може погіршити інші показники методу. З урахуванням цього бажаним є пошук можливості поліпшення результатів застосування стеганометоду, розширення області його застосування без будь-якої модифікації.

Усі контейнери, що використовуються в стеганографії, можна розділити на три класи: обрані, випадкові та нав'язані [12]. Саме вибір контейнера дозволяє найкраще забезпечити необхідні властивості СП, але на практиці найчастіше контейнер є випадковим, наслідком чого є бажаність незалежності властивостей стеганоалгоритму від властивостей контейнера. Однак багато наявних стеганографічних методів не розраховані на випадковий контейнер, маючи обмеження на область застосування, що очевидно є їх недоліком [8,13]. Так в [14] запропонований стійкий до атак проти вбудованого повідомлення метод, що робить вбудову ДІ в просторовій області ЦЗ-контейнера шляхом певних збурень яскравості пікселів. Цей метод не гарантує забезпечення надійності сприйняття СП, що зазначається одним з авторів в [15], де пропонується його модифікація, яка теж не вирішую проблему остаточно.

Більшість сучасних стеганометодів, зокрема розглянуті вище, є блоковими [16,17], що дозволяє: краще забезпечити їх стійкість до атаки стиском; порівняно незначну обчислювальну складність, оскільки тут, незалежно від конкретики способу обробки окремого блоку, для матриці розміром $n \times n$ вона становить $O(n^2)$ операцій, тобто визначається кількістю непересічних блоків; можливість розпаралелювання тощо. Очевидно, що блоки в межах одного ЦЗ розрізняються по своїх властивостях і вибір серед них для вбудови ДІ принципово може забезпечити певні вимоги до СП навіть при випадковому контейнері, тобто є тим джерелом, при використанні якого існує принципова можливість забезпечити надійність сприйняття СП для довільного стеганометоду, навіть стійкого до атак проти вбудованого повідомлення, для випадкового ЦЗ-контейнера.

Метою роботи є підвищення ефективності стеганографічної системи при використанні довільного стеганометоду і випадкового контейнеру шляхом розробки методу вибору блоків ЦЗ-контейнера для СПр.

Під ефективністю стеганосистеми в роботі розуміється ступінь забезпечення надійності сприйняття СП, яка кількісно оцінюються

стандартним для стеганографії чином – різницевим показником *PSNR* - піковим відношенням «сигнал-шум» [1].

Звичайно, будь-який вибір блоків контейнера потенційно зменшує можливу пропускну спроможність формованого прихованого каналу зв'язку, але, по-перше, такий вимушений захід дасть можливість для використання існуючих ефективних методів без будь-яких їх модифікацій в умовах випадкового контейнера, по-друге, з урахуванням бурхливого розвитку стеганоаналітичних методів використання стеганометодів в умовах малої пропускної спроможності стеганоканалу є поширеною сучасною тенденцією [18].

Для досягнення поставленої мети в роботі розв'язуються наступні *задачі*:

1. Визначення параметра блоку матриці ЦЗ, що дає кількісну характеристику розподілу його частотних складових незалежно від формату збереження (з/без втрат) зображення;
2. Розробка методу вибору блоків ЦЗ-контейнера для СПр, заснованого на аналізі нормованої відокремленості максимального СНЧ блоку;
3. Розробка методу вибору ЦЗ-контейнера для підвищення ефективності довільного стеганометоду без його модифікації.

Інтегральний показник розподілу частотних складових блоку цифрового зображення

При вбудові ДІ для забезпечення різноманітних властивостей отримуваного СП важливим є те, які саме збурення отримують в результаті СПр формальні параметри контейнера, зокрема частотні коефіцієнти, параметри сингулярного розкладання відповідної матриці тощо, та локалізація цих збурень.

Нехай F – $n\times n$-матрица ЦЗ-контейнера, що піддається розбивці на непересічні $l\times l$-блоки $B_{ij}, i, j = \overline{1,\lceil n/l \rceil}$, стандартним чином [7], де $\lceil\cdot\rceil$ - ціла частина аргументу. Блоки мають різні властивості в межах одного ЦЗ, зокрема мають різний вміст частотних складових. Як вже зазначалося вище, при окремому розгляді одних з основних вимог до СП, а саме забезпечення надійності сприйняття та нечутливості до ЗД, для забезпечення першої бажаним є задіювання в процесі вбудови ДІ високочастотної складової, в той час, як для забезпечення стійкості пріоритет має низькочастотна складова. Одночасне максимальне задоволення цих вимог є неможливим, але для досягнення певного компромісу бажаним є визначення (при можливості) такого параметру, який би давав інтегральну характеристику ступеня вмісту різних частот у конкретний блок. Саме такий параметр при його використанні дасть змогу для вибору шуканих блоків для СПр з потрібним співвідношенням частотних складових.

В якості такого інтегрального параметру з урахуванням результатів досліджень, проведених в [19,20], пропонується використовувати нормовану відокремленість максимального СНЧ блоку (НВМСЧ) ЦЗ. Покажемо, що її значення є загальним показником вмісту частотних складових в блок.

Нехай

$$B = U\Sigma V^T \quad (1)$$

- нормальне сингулярне розкладання блоку *B* деякого ЦЗ [10], де *U, V* – ортогональні матриці, стовпці яких u_i, v_i , $i = \overline{1,l}$, є лівими і правими СНВ відповідно, при цьому ліві СНВ є лексикографічно додатними, $\Sigma = diag(\sigma_1(B),..., \sigma_l(B))$,

$$\sigma_1(B) \geq ... \geq \sigma_l(B) \geq 0 \quad (2)$$

– СНЧ *B*, $(\sigma_i(B), u_i, v_i)$, $i = \overline{1,l}$, - сингулярні трійки. Для блоків оригінального ЦЗ, незалежно від формату збереження (з/без втрат) співвідношення (2) можна уточнити:

$$\sigma_1(B) >> \sigma_2(B) \geq ... \geq \sigma_l(B) \geq 0 . \quad (3)$$

Нормальнее сингулярне розкладання в умовах відсутності кратних СНЧ визначається однозначно [10] на відміну від звичайного [21].

Між сингулярними трійками і частотними складовими (блоків) ЦЗ існує певна відповідність [9]: сингулярні трійки, що містять максимальні/ мінімальні/ середні за значенням СНЧ, несуть в собі, головним чином, інформацію про низькочастотну/високочастотну/середньочастотну складові *B*, але і всі інші частотні складові в різних співвідношеннях будуть присутніми в кожній матриці $\sigma_i(B)u_i v_i^T$, що є доданком в сингулярному розкладанні в формі зовнішніх добутків [21]: $B = U\Sigma V^T = \sum_{i=1}^{l} \sigma_i(B) u_i v_i^T$. При цьому саме СНЧ в цих трійках є відповідальними за основну частотну складову блока. Це випливає з наступного [9]. Енергія E (*B*) *l*×*l*-блоку *B* ЦЗ визначається у відповідності з формулами: $E(B) = \sum_{u=0}^{l-1} \sum_{v=0}^{l-1} P(u,v) = \sum_{i=1}^{l} \sigma_i^2(B)$, де $P(u,v), u = \overline{0,l-1}, v = \overline{0,l-1}$, - енергетичний спектр *B*. Для блоків ЦЗ чим менше високочастотна/низькочастотна складова, тим більше/менше перше СНЧ відрізняється від всіх інших у відповідності з (3) [19,20]. Таким чином, ступінь цієї відмінності можна розглядати як показник вмісту певних частотних складових у блок ЦЗ. Але сингулярні спектри блоків дуже розрізняються між собою. Так, наприклад, якщо розглянути два 8×8-блоки ЦЗ (рис.1(а) – місця розташування блоків позначені червоними стрілками), то їх сингулярні спектри визначаться наступним чином:

897.7032, 15.2044, 4.5597, 2.9882, 1.7688, 1.6169, 0.5087, 0.3562;
1.0e+003 *1.3086, 0.0655, 0.0137, 0.0057, 0.0029, 0.0022, 0.0017, 0.0007. (4)

Очевидно, що співвідношення між значеннями СНЧ різних блоків можуть настільки відрізнятися між собою, що встановити по них порівняне співвідношення між вмістом високо/низько/середньочастотних складових для

різних блоків буде складно. Але картина стає більш зрозумілою після нормування сингулярного спектру. Позначимо $\sigma = (\sigma_1(B), \sigma_2(B), ..., \sigma_l(B))^T$ - вектор сингулярного спектру блоку *B*; $\overline{\sigma} = \sigma / \|\sigma\|$ - нормований вектор СНЧ, де $\|\sigma\|$ норма σ. Для $\overline{\sigma}$ відносні співвідношення між СНЧ є очевидними, оскільки всі його елементи належать одній множині – сегменту [0,1]. Показником ступеня відмінності $\sigma_1(B)$ від інших СНЧ блоку, а тому кількісною характеристикою внеску певних частот в блок, виступає нормована відокремленість $\sigma_1(B)$. Нормована відокремленість $svdgap_n(i)$ СНЧ $\sigma_i(B)$ визначається у відповідності з формулою [19,20]: $svdgap_n(i) = \min\limits_{i \neq j} \left| \overline{\sigma}_j - \overline{\sigma}_i \right|$, де $\overline{\sigma}_i$, $i = \overline{1,l}$ - компоненти вектора $\overline{\sigma}$. Тоді:

$$svdgap_n(1) = \overline{\sigma}_1 - \overline{\sigma}_2, \qquad (5)$$

При цьому $svdgap_n(1) \in]0,1]$. Значення нуля виключається з можливих для $svdgap_n(1)$ завдяки (3).

Враховуючи [20], можна зробити висновок, що чим ближче $svdgap_n(1)$ до одиниці, тим більшою є ступінь відносного перевищення $\overline{\sigma}_1$, а тому і $\sigma_1(B)$ всіх інших СНЧ блоку, тим менше їх відносний внесок в сингулярний спектр (в енергію блоку), тим більше/менше відносний вклад низькочастотної/ високочастотної складової в цей блок. Для розглянутого вище приклада блоків з сингулярними спектрами (4), відповідні вектори $\overline{\sigma}$ мають вигляд:

0.9998, 0.0169, 0.0051, 0.0033, 0.0020, 0.0018, 0.0006, 0.0004;
0.9987, 0.0500, 0.0105, 0.0043, 0.0022, 0.0017, 0.0013, 0.0006,

При цьому для першого з блоків $svdgap_n(1)$=0.9829, тоді як для другого $svdgap_n(1)$=0.9487, тобто на 4% менше, що говоре про другий блок як такий, вміст високочастотної складової в який є відносно більшим, ніж в перший, що повністю відповідає реаліям, які чітко прослідковуються по матрицях блоків:

$$\begin{pmatrix} 81 & 98 & 109 & 111 & 111 & 116 & 121 & 129 \\ 83 & 103 & 108 & 110 & 113 & 117 & 121 & 127 \\ 85 & 101 & 108 & 110 & 114 & 118 & 121 & 126 \\ 88 & 102 & 108 & 109 & 114 & 119 & 122 & 127 \\ 89 & 103 & 107 & 110 & 116 & 120 & 123 & 127 \\ 91 & 106 & 106 & 110 & 116 & 119 & 122 & 127 \\ 93 & 109 & 108 & 111 & 116 & 120 & 123 & 127 \\ 96 & 112 & 108 & 109 & 116 & 120 & 125 & 129 \end{pmatrix}, \begin{pmatrix} 103 & 148 & 167 & 176 & 178 & 179 & 176 & 166 \\ 106 & 153 & 166 & 176 & 178 & 180 & 179 & 174 \\ 110 & 155 & 169 & 175 & 178 & 182 & 179 & 176 \\ 112 & 157 & 168 & 175 & 179 & 181 & 178 & 175 \\ 115 & 160 & 171 & 175 & 179 & 178 & 174 & 166 \\ 120 & 162 & 172 & 175 & 178 & 177 & 153 & 131 \\ 124 & 163 & 171 & 177 & 177 & 176 & 151 & 127 \\ 129 & 165 & 169 & 178 & 179 & 174 & 151 & 141 \end{pmatrix},$$

Тут очевидним є більш значний перепад значень яскравості для другого блоку, який говоре про наявність контурів об'єктів, що локалізуються в області першого і другого стовпців, а також 6-го, 7-го і 8-го стовпців в області рядків з тими ж номерами, в межах цього блоку.

Розглянемо ЦЗ у вигляді сукупності непересічних блоків, отриманих стандартним чином, що очевидно відрізняються ступенем внеску високо/низькочастотної складової. Для наочності приклади таких зображень представлені на рис.1(а) - ЦЗ зі значними перепадами значень яскравості – велика кількість об'єктів різного розміру, порівняно значний вміст високочастотної складової, на рис.2(а) – ЦЗ, що має велику частину з незначними перепадами значень яскравості, незначну кількість об'єктів, ліній контурів на зображенні, порівняно значний вміст низькочастотної складової. З отриманого вище випливає, що для другого (далі – «фонового») зображення кількість блоків, де $svdgap_n(1) \approx 1$, буде значно більше, ніж для першого (далі – «контурного»), крім цього, для першого може бути значна кількість блоків, які містять малі деталі, контури, мають значні перепади значень яскравості пікселів, тобто де $svdgap_n(1) << 1$. Всі ці передбачення в сукупності підтверджуються конкретними властивостями гістограм $svdgap_n(1)$ для цих ЦЗ, які очікувано відрізняються. Позначимо Γ_F і Γ_K - гістограми $svdgap_n(1)$ для фонового і контурного ЦЗ відповідно.

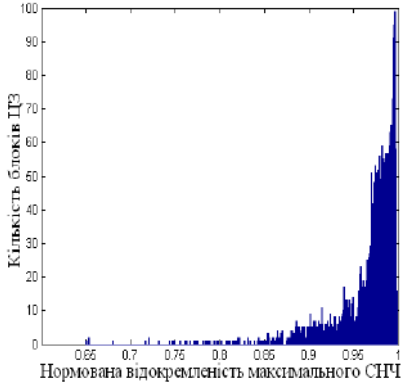
Відмінності у властивостях Γ_F і Γ_K обумовлені ступенем відносного внеску високо/низькочастотної складової в блок ЦЗ і, враховуючи, що переважна кількість блоків фонового зображення має СНЧ $\sigma_2(B),..., \sigma_l(B)$, значення яких порівняні з нулем, а $svdgap_n(1)$ порівняну з одиницею, є наступними:

- Моди $m(\Gamma_F)$ і $m(\Gamma_K)$ гістограм Γ_F і Γ_K відповідають співвідношенню: $m(\Gamma_F) \geq m(\Gamma_K)$;
- значення в модах $V(m(\Gamma_F))$ і $V(m(\Gamma_K))$ відповідають співвідношенню: $V(m(\Gamma_F)) >> V(m(\Gamma_K))$.

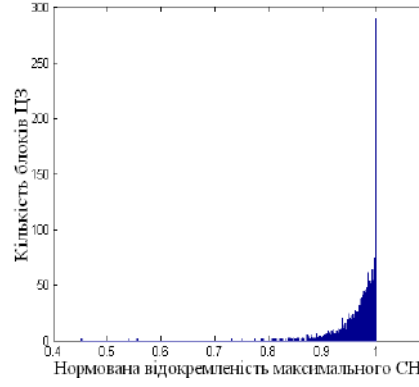
При цьому незначному околу $m(\Gamma_F)$ відповідає, як правило, вклад переважної більшості блоків ЦЗ (аж до 95%), тоді як для контурного зображення поза таким околом, як правило, буде знаходитися вклад значної кількості блоків (понад 50%);

- В Γ_F розкид можливих значень $svdgap_n(1)$ не перевищує розкид в Γ_K, при цьому значення гістограми Γ_F в аргументах, що не належать малому околу моди, будуть порівняні з нулем.

а



б



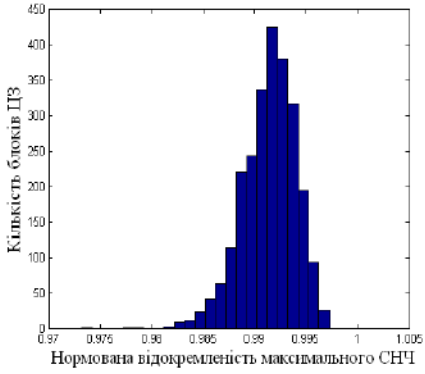
в

Рисунок 1. Гістограми значень НВМСЧ для конкретного «контурного» ЦЗ: а – оригінальне ЦЗ в форматі без втрат, б – Γ_K для оригінального ЦЗ; в - гістограма НВМСЧ для ЦЗ, перезбереженого у формат Jpeg (*QF*=75)

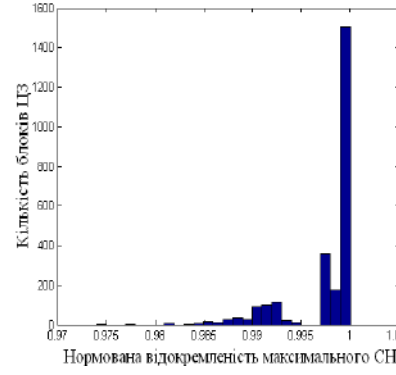
Відіб'ється на $svdgap_n(1)$ і процес зменшення високочастотної складової у блоках ЦЗ (незалежно від його первісного формату (з/без втрат)) в результаті відповідної обробки ЦЗ, такої як розмиття, збереження в форматі з втратами тощо: відбудеться «зсув» як Γ_F, так і Γ_K вправо, що є результатом незменшення $svdgap_n(1)$, який призведе до:

- незменшення значень $m(\Gamma_F)$ і $m(\Gamma_K)$; збільшення значень $V(m(\Gamma_F))$ і $V(m(\Gamma_K))$.

а



б



в

Рисунок 2. Гістограми значень НВМСЧ для конкретного «фонового» ЦЗ:
а – оригінальне ЦЗ в форматі без втрат, б – Γ_F для оригінального ЦЗ;
в - гістограма НВМСЧ для ЦЗ, перезбереженого у формат Jpeg (*QF*=75)

Отримані теоретичні висновки знайшли практичне підтвердження в ході обчислювального експерименту, типові результати якого продемонстровані на рис.1,2 у вигляді гістограм значень $svdgap_n(1)$ для блоків ЦЗ, що очевидно відрізняються відносним вмістом високочастотної/низькочастотної складової.

Все вищенаведене говорить на користь використання (5) як інтегрального параметра для оцінки вкладу високочастотної/низькочастотнох складової в блок ЦЗ і практично підтверджує перспективність підходу до вибору блоків для СПр, заснованого на аналізі $svdgap_n(1)$.

Метод вибору блоків зображення-контейнера для вбудови додаткової інформації

Як зазначено вище, для забезпечення потенційної можливості декодуваня ДІ після атаки проти вбудованого повідомлення величина збурення контейнера в результаті СПр повинна бути не менше, ніж величина збурення СП в результаті атаки, наслідком чого є більш значні збурення контейнера під час СПр за допомогою методу, стійкого до значних ЗД, а тому і забезпечення тут надійності сприйняття є більш складною задачею в порівнянні з аналогічною вимогою для метода, що не забезпечує стійкість. Яскравим прикладом тут може слугувати метод Коха і Жао, ступінь стійкості якого до атак протии вбудованого повідомлення залежить від вибори конкретних коефіцієнтів дискретного косинусного перетворення (ДКП) блоку, які задіюються в процесі вбудови ДІ [1] – стійкість буде зростати зі зменшенням частоти, коефіцієнти якої змінюються певним чином при СПр. При цьому при зменшенні частоти використовуваних коефіцієнтів ДКП при СПр буде збільшуватися ступінь спотворення контейнера і зменшуватися ймовірність збереження надійності сприйняття СП (рис.3).

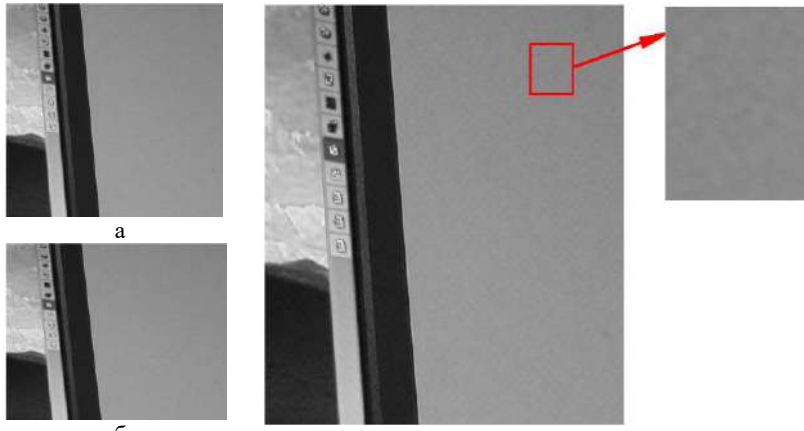


Рисунок 3. Ілюстрація візуальної якості СП, отриманого методом Коха і Жао при розбивці матриці контейнера на 8×8-блоки і задіюванні всіх блоків для СПр: а – оригінальне ЦЗ-контейнер; б – СП, отримане з використанням (4,5) и (5,4) коефіцієнтів ДКП; в - СП, отримане з використанням (2,3) и (3,2) коефіцієнтів ДКП

Ймовірність збереження надійності сприйняття (блоку) спотвореного ЦЗ, враховуючи особливості людської зорової системи [7] зростає, якщо це зображення (блок) містить маленькі деталі, значну кількість контурів, тобто має значну високочастотну складову. Таким чином, для забезпечення можливості застосування довільного, зокрема стійкого до атак проти вбудованого повідомлення, стеганоалгоритма до випадкового контейнера має

сенс задіювати в процесі СПр блоки контейнера зі значною високочастотною складовою, тобто такі, які мають відносно велике значення $svdgap_n(1)$.

Таким чином, основні кроки методу вибору блоків ЦЗ-контейнера для підвищення ймовірності забезпечення надійності сприйняття СП, в тому числі отриманого за допомогою стійкого до атак проти вбудованого повідомлення стеганографічного методу, який далі називається *SelectBlock*, наступні.

Крок 1. Матрицю F ЦЗ-контейнера розміром $n \times n$ розбити стандартним чином на непересічні $l \times l$-блоки $B_{ij}, i, j = \overline{1, \lceil n/l \rceil}$. Розрахувати t – кількість блоків контейнера, необхідну для вбудови ДІ, що є результатом попереднього кодування і представляє цифрову, як правило, бінарну, послідовність: $p_1, p_2, \ldots, p_m$.

Крок 2. Для кожного блоку $B_{ij}, i, j = \overline{1, \lceil n/l \rceil}$:

2.1. Обчислити СНЧ (за допомогою сингулярного розкладання (1)), визначити вектор сингулярного спектру: $\sigma = (\sigma_1(B_{ij}), \sigma_2(B_{ij}), \ldots, \sigma_l(B_{ij}))^T$;

2.2. Визначити нормований вектор сингулярного спектру блоку: $\overline{\sigma} = \sigma / \|\sigma\|$;

2.3. Знайти НВМСЧ блоку $svdgap_n(1)$ (5).

Крок 3. Впорядкувати блоки $B_{ij}, i, j = \overline{1, \lceil n/l \rceil}$ за неспаданням $svdgap_n(1)$. Перші t з них – блоки, що рекомендовані для вбудови ДІ.

Очевидно, що застосування запропонованого методу має сенс проводити лише тоді, коли $t < \lceil n/l \rceil$, тобто, коли є можливість використовувати не всі блоки контейнеру для пересилання наявного секретного повідомлення. Більше того, очікувати значний ефект можна тоді, коли $t << \lceil n/l \rceil$, і тут: чим більше t відрізняється від $\lceil n/l \rceil$, тим більше «поле вибору», тим кращий результат в сенсі забезпечення надійності сприйняття СП можна очікувати. Як показує практика, метод має сенс застосовувати, коли

$$t < \lceil n/l \rceil / 2. \quad (6)$$

Якщо (6) для ЦЗ не виконується, для пересилання наявного секретного повідомлення можна застосовувати у якості контейнера не одне, а декілька ЦЗ (чи цифрове відео, кадри якого розглядаються як ЦЗ), для кожного з яких умова (6) буде мати місце.

Запропонований метод *SelectBlock* є блоковим, що визначає його обчислювальну складність для $n \times n$-ЦЗ як $\underline{O}(n^2)$, але це не обмежує область його застосування тільки блоковими стеганографічними методами: він може бути використаний і для стеганометодів, що не є блоковими, наприклад, для (різноманітних модифікацій) методу модифікації найменшого значущого біта [22], виділяючи в ЦЗ-контейнері його області у вигляді об'єднання блоків, найбільш сприятливі для вбудови ДІ.

Тестування запропонованого методу *SelectBlock* проводилося з використанням множини з 800 ЦЗ розміром 800×800 пікселів з баз [23,24]. В кожне з цих ЦЗ, які розбивалися на непересічні 8×8-блоки (загальна кількість блоків – 10000) вбудовувалася ДІ – сформована випадковим чином бінарна послідовність довжини $t = 3500$, яка протягом всього експерименту залишалася незмінною. Вбудова ДІ проводилася двома стеганографічними методами: Коха і Жао [1] і методом, запропонованим в [4], що на сьогодні залишається одним з найстійкіших до атаки стиском. Для кожного ЦЗ-контейнера СПр кожним з методів проводилося двічі: блоки для вбудови ДІ обиралися довільним (випадковим) чином; блоки обиралися у відповідності з рекомендаціями, що надає *SelectBlock.* В кожному разі обчислювалася кількісна оцінка спотворення ЦЗ-контейнера – *PSNR*. Результати, які надають безумовну перевагу над довільним вибором блоку методу *SelectBlock,* наведені в табл..1. Зафіксоване підвищення значення *PSNR* на 8.39% для методу Коха і Жао, і на 7.13% для методу [4].

Таблиця 1

Середнє значення *PSNR* в результаті вбудови ДІ

Стеганометод	Середнє значення *PSNR* (dB)	
	Вибір блоків випадковим чином	Вибір блоків з використанням рекомендацій *SelectBlock*
Коха і Жао	42.54	46.11
Метод [4]	41.23	44.17

Метод визначення цифрових зображень, небажаних для використання в якості стеганографічних контейнерів

Запропонований підхід, заснований на використанні НВМСЧ, може бути застосований для відкидання ЦЗ, які не бажано використовувати в якості контейнера – зображень, що мають значну частину з невеликими перепадами значень яскравості пікселів. І хоча це не забезпечує ефективну роботу стеганометоду з випадковим контейнером, а передбачає його вибір, але дає можливість підвищити ефективність наявних методів без їх модифікації. Виходячи з вищенаведеного, потрібні висновки можна зробити, аналізуючи в сукупності властивості блоків ЦЗ, тобто аналізуючи гістограму $svdgap_n(1)$ блоків.

Основні кроки методу, що використовується для «відкидання» небажаних для ролі контейнера ЦЗ і далі називається *SelectImage*, наступні.

Крок 1. Матрицю F ЦЗ розміром $n \times n$ розбити стандартним чином на непересічні $l \times l$-блоки $B_{ij}, i, j = \overline{1, \lfloor n/l \rfloor}$.

Крок 2. Співпадає з кроком 2 методу *SelectBlock.*

Крок 3. Побудувати гістограму Γ значень $svdgap_n(1)$ для блоків досліджуваного ЦЗ.

Крок 4. Визначити моду гістограми $m(\Gamma)$.

Крок 5. Визначити окіл $m(\Gamma)$ незначного радіусу δ: $[m(\Gamma)-\delta, m(\Gamma)+\delta]$, де δ - параметр, що визначається експериментально.

Крок 6. Розрахувати відносну кількість блоків ЦЗ (%) *W*, для яких $svdgap_n(1) \in [m(\Gamma)-\delta, m(\Gamma)+\delta]$.

Крок 7. *Якщо* $W > P$,

то досліджуване зображення не бажано використовувати як контейнер,

де *P* – порогове значення, встановлюване експериментально

Для алгоритмічної реалізації використовуються наступні значення параметрів: *l*=8, $\delta = 0.006$, *P*=86%. При проведенні обчислювального експерименту для оцінки ефективності *SelectBlock* використовувалися 1000 ЦЗ розміром 400×400 пікселів з баз зображень [23, 24, 25] – множина *M*. При застосуванні *SelectImage* з множити *M* було видалено 183 ЦЗ, приклади яких представлені на рис.4, які повністю відповідають суб'єктивній оцінці їх «невідповідності» ролі контейнера.

Рисунок 4. ЦЗ з бази [23], що були обрані як небажані для застосування в ролі контейнера методом *SelectImage*

Зображення, що залишилися, були об'єднані в множину $\overline{M}$. ЦЗ з множини *M* піддавалися СПр (вбудовувалася одна й та сама ДІ) методом Коха і Жао з використанням коефіцієнтів ДКП (4,5) і (5,4). Блоки та їх послідовність обиралися випадковим чином. Для кожного СП обраховувалося значення *PSNR*. При цьому середнє значення $PSNR_M$ для ЦЗ з множини *M* склало $PSNR_M = 36.41\,dB$, а по множині $\overline{M}$ - $PSNR_{\overline{M}} = 38.10\,dB$, що говорить про покращення (підвищення) на 4.64% кількісного показника спотворення ЦЗ при СПр в результаті застосування *SelectImage*. Звичайно, цей відсоток буде залежати від вмісту множини *M*: чим менше там первісно буде фонових ЦЗ, тим меншим буде відсоток збільшення $PSNR_M$, але при застосуванні *SelectImage* зменшення $PSNR_M$ не спостерігатиметься.

Обчислювальна складність *SelectImage* для ЦЗ з $n \times n$-матрицею, враховуючу його блокову організацію, визначається аналогічно обчислювальній складності *SelectBlock* і становить $\underline{O}\left(n^2\right)$ операцій.

Висновки

В роботі вирішена важлива науково-практична задача підвищення ефективності стеганосистеми при використанні довільного стеганометоду, в тому числі, стійкого до атак проти вбудованого повідомлення, який може призвести до значних спотворень контейнера, і випадкового контейнеру шляхом розробки методу вибору блоків ЦЗ-контейнера для вбудови в них ДІ, де ефективність стеганосистеми оцінювалась ступенем спотворення контейнера в результаті вбудови ДІ, який кількісно розраховувався за допомогою показника *PSNR* - пікового відношення «сигнал-шум».

В ході роботи отримані наступні результати:

- обґрунтовано вибір інтегрального показника розподілу частотних складових блоку ЦЗ (незалежно від формату збереження: з/без втрат) – нормованої відокремленості максимального СНЧ та досліджені його властивості, що дало можливість для побудови методу вибору блоків на основі кількісного аналізу $svdgap_n(1)$, який дозволив при своєму застосуванні підвищити ефективність стеганосистеми більше, ніж на 7% при використанні стеганометодів, що не гарантують системне забезпечення надійності сприйняття формованного СП;
- встановлені відмінності між гістограмами значень $svdgap_n(1)$ для ЦЗ, що відрізняються відносним вмістом низькочастотної/високочастотної складової, що дало можливість для побудови методу визначення небажаних для ролі контейнера ЦЗ – зображень зі значною областю з малими перепадами значень яскравості, що дало можливість підвищити ефективність стеганосистеми в сенсі забезпечення надійності сприйняття СП більше, ніж на 4%;
- обидва запропоновані методи мають незначну обчислювальну складність, яка для ЦЗ з $n \times n$-матрицею становить $\underline{O}\left(n^2\right)$ операцій. Така обчислювальна складність для методу вибору блоків в ЦЗ-контейнері забезпечує перспективу його використання для потокового контейнеру.

Література

1. Konakhovich, G. F., Progonov, D. O., & Puzirenko, O. Y. (2018). *Computer steganographic processing and analysis of multimedia data*. Kyiv, UA: Center for Educational Literature.

2. Karampidis, K., Kavallieratou, E., & Papadourakis, G. (2018). A review of image steganalysis techniques for digital forensics. *Journal of Information Security and Applications, 40,* 217–235. https://doi.org/10.1016/j.jisa.2018.04.005

3. Bohang, L., et al. (2025). Image steganalysis using active learning and hyperparameter optimization. *Scientific Reports, 15,* Article 7340. https://doi.org/10.1038/s41598-025-92082-w

4. Melnik, M. A. (2012). Compression-resistant steganography algorithm. *Information Security, 2,* 99–106.

5. Kozina, M. O. (2015). Method of verification of authenticity of information that is passed by steganographic communication channel. *Visnyk of Vinnytsia Polytechnical Institute, 1,* 117–121.

6. Michaylov, K. D., & Sarmah, D. K. (2024). Steganography and steganalysis for digital image enhanced forensic analysis and recommendations. *Journal of Cyber Security Technology, 9*(1), 1–27. https://doi.org/10.1080/23742917.2024.2304441

7. Gonzalez, R. C., & Woods, R. E. (2007). *Digital image processing* (3rd ed.). London, UK: Pearson.

8. Srinivasan, D., Manojkumar, K., Syed, A., & Nutakki, H. (2025). A comprehensive review on advancements and applications of steganography. *ResearchGate.* Retrieved June 1, 2025, from https://doi.org/10.13140/RG.2.2.13568.44807

9. Kobozeva, A. A., Machalin, I. O., & Khoroshko, V. O. (2010). *Security analysis of information systems*. Kyiv, UA: DUIKT.

10. Bergman, C., & Davidson, J. (2025). Unitary embedding for data hiding with the SVD. *Iowa State University of Science and Technology.* Retrieved June 1, 2025, from https://dr.lib.iastate.edu/handle/20.500.12876/54635

11. Kobozeva, A. A., & Sokolov, A. V. (2022). The sufficient condition for ensuring the reliability of perception of the steganographic message in the Walsh-Hadamard transform domain. *Problemele Energeticii Regionale, 2,* 84–100. https://doi.org/10.52254/1857-0070.2022.2-54.08

12. Pramanik, S., Ghonge, M. M., Ravi, R. V., & Cengiz, K. (Eds.). (2021). *Multidisciplinary approach to modern digital steganography*. Hershey, USA: IGI Global.

13. Abdulla, A. A. (2024). Digital image steganography: Challenges, investigation, and recommendation for the future direction. *Soft Computing, 28,* 8963–8976. https://doi.org/10.1007/s00500-023-09130-8

14. Rudnitsky, V. M., & Kostyrka, O. V. (2013). Robust steganotransformation in spatial domain of cover image. *Informatics and Mathematical Methods in Simulation, 3*(4), 353–360.

15. Kostyrka, O. V. (2016). Modification of resistance to disturbance quilted transformation of spatial image container. *Informatics and Mathematical Methods in Simulation, 6*(1), 85–93.

16. Qiao, T., et al. (2023). Robust steganography in practical communication: A comparative study. *EURASIP Journal on Image and Video Processing, 2023,* Article 15. https://doi.org/10.1186/s13640-023-00615-y

17. Liu, Q., et al. (2022). A robust image steganographic scheme against general scaling attacks. *arXiv.* arXiv:2212.02822

18. Bobok, I. I. (2018). Steganalysis method for detection of the hidden communication channel with low capacity. *Telecommunications and Radio Engineering*, *77*(18), 1597–1604. https://doi.org/10.1615/TelecomRadEng.v77.i18.20

19. Bobok, I., Kobozieva, A., & Kushnirenko, N. (2021). Use of the normalized gap of maximum singular value of the image block to evaluate the capacity of the steganographic channel. In *Proceedings of the 3rd International Conference on Information Security and Information Technologies (ISecIT 2021)* (pp. 183–188).

20. Kobozeva, A. A., Bobok, I. I., & Kushnirenko, N. I. (2022). Method for distinguishing the digital images in different formats. *Problemele Energeticii Regionale, 1,* 109–124. https://doi.org/10.52254/1857-0070.2022.1-53.09

21. Demmel, J. W. (1996). *Applied numerical linear algebra*. Philadelphia, USA: SIAM.

22. Aslam, M. A., et al. (2022). Image steganography using least significant bit (LSB) – A systematic literature review. In *Proceedings of the 2022 2nd International Conference on Computing and Information Technology (ICCIT)* (pp. 32–38). https://doi.org/10.1109/ICCIT52419.2022.9711628

23. Gloe, T., & Böhme, R. (2010). The Dresden Image Database for benchmarking digital image forensics. In *Proceedings of the 2010 ACM Symposium on Applied Computing (SAC '10)* (pp. 1584–1590). https://doi.org/10.1145/1774088.1774427

24. NRCS Photo Gallery. (2025). *USDA.* Retrieved June 1, 2025, from https://www.nrcs.usda.gov

25. Hsu, Y.-F., & Chang, S.-F. (2006). Detecting image splicing using geometry invariants and camera characteristics consistency. In *Proceedings of the 2006 IEEE International Conference on Multimedia and Expo* (pp. 549–552). https://doi.org/10.1109/ICME.2006.262447

NORMALIZED SEPARABILITY OF THE MAXIMUM SINGULAR NUMBER OF A DIGITAL IMAGE BLOCK AS AN INDICATOR OF THE FEASIBILITY OF ITS STEGANOTRANSFORMATION

Dr.Sci. I. Bobok[1][0000-0003-4548-0709], **Dr.Sci. A. Kobozova**[2][https://orcid.org/0000-0001-7888-0499]

[1]Odesa Polytechnic National University, Ukraine,
[2]Odesa National Maritime University, Ukraine
EMAIL: [1]onu_metal@ukr.net, [2]alla_kobozeva@ukr.net

Abstract. *One of the most promising and effective directions in information security today is steganography. Among a number of requirements for a steganographic system, one of the key ones is ensuring the reliability of perception of the steganogram. However, many existing steganographic methods, particularly those robust against attacks on the embedded message, are unable to systematically*

ensure this requirement and are not designed to work with a random container, which is most frequently used in practice. The aim of this work is to increase the efficiency of a steganographic system when using an arbitrary steganographic method and a random container. This is achieved through the development of a method for selecting container blocks from a digital image to embed additional information within them. The criterion for efficiency is the reliability of perception of the generated steganogram, quantitatively evaluated using PSNR. The goal was achieved by investigating the properties of formal parameters of digital image blocks – singular values, considering the correspondence between singular triplets and the frequency components of the block. The most important result of this work is the theoretically substantiated definition of a block parameter that provides an integral quantitative characteristic of the relative distribution of its frequency components – the normalized separability of the maximum singular value of the block. The practical significance of the obtained results lies in the development of a method for selecting image blocks and a method for identifying images undesirable for use as containers. The algorithmic implementations of these methods have significantly improved the efficiency of the steganographic system when applied.

***Keywords**: steganography, digital image, reliability of perception of the steganogram, container block selection, container selection, normalized separability of the singular value*

UDC 004.623
DOI https://doi.org/10.36059/978-966-397-538-2-5

ІНТЕЛЕКТУАЛЬНИЙ ГІБРИДНИЙ ПРОЄКТ СИСТЕМИ ДЛЯ АНАЛІЗУ РИЗИКІВ ШКОДИ ЗДОРОВ'Ю ПРИ ВЕЛИКИХ ОБСЯГАХ ДАНИХ

Ph.D. М. Рудніченко[1][0000-0002-7343-8076], **Ph.D. Н. Шибаєва**[1][0000-0002-7869-9953], **Ph.D. Т. Отрадська**[2][0000-0002-5808-5647], **Д. Шведов**[1][0009-0002-4823-8782], **Ph.D. І. Шпінарева**[1][0000-0001-9208-4923], **Dr.Sci. І.Петров**[1][0000-0002-8740-6198]

[1]Національний університет «Одеська Політехніка», Україна,
[2]Одеський коледж комп'ютерних технологій «СЕРВЕР», Україна
EMAIL: nickolay.rud@gmail.com, nati.sh@gmail.com, tv_61@ukr.net, frumle@ukr.net, iryna.shpinareva@onu.edu.ua, firmn@gmail.com

***Анотація.** У статті представлено всебічне дослідження щодо розробки проекту гібридної інтелектуальної системи для аналізу ризиків шкоди здоров'ю населення на великих обсягах екологічних даних, спричиненої забрудненням повітря, із використанням гібридної архітектури на основі моделей машинного та глибинного навчання. Актуальність дослідження обґрунтована зростаючими загрозами для здоров'я, пов'язаними з атмосферними забруднювачами, такими як PM2.5, PM10, NO_2, SO_2, CO та*

O_3, особливо у густонаселених міських середовищах. Стаття підкреслює нагальну потребу використання інтелектуальних систем на основі даних для надання точних, масштабованих та інтерпретованих оцінок ризиків для здоров'я населення на рівні популяції. Проведено детальний аналіз існуючих наукових підходів, виділено обмеження класичних статистичних моделей та підкреслено переваги використання нейронних мереж і методів ансамблевого навчання для моделювання складних нелінійних залежностей та просторово-часової динаміки у гетерогенних екологічних даних. Запропонована система реалізує шестиступеневу модельну структуру, що включає дерева рішень, метод опорних векторів, випадкові ліси, XGBoost, згорткові нейронні мережі та моделі довготривалої короткочасної пам'яті (LSTM). Кожен компонент сприяє формуванню структурованого конвеєра для попередньої обробки даних, класифікації, прогнозування та стратифікації ризиків. У дослідженні описано використані набори даних для навчання та валідації, переважно Air Pollution Image Dataset з Індії та Непалу, доповнені структурованими наборами даних якості повітря. Архітектура системи реалізована за моделлю клієнт-сервер, де бекенд розроблено на Python та Flask, а фронтенд — на JavaScript. Компоненти програмного забезпечення та взаємодії класів представлені через UML-діаграми, а аналітична панель системи обладнана модулями для візуалізації даних, прогнозування моделей та порівняння їхньої ефективності. Експериментальні результати підтверджують ефективність запропонованого підходу з використанням ансамблевих та глибинних моделей. Проведено порівняльну оцінку всіх моделей за показниками точності (accuracy), точності прогнозу (precision), повноти (recall), F1-міри та середнього часу навчання.

Ключові слова: *інтелектуальний аналіз даних, аналіз здоров'я, великі дані, гібридний інтелектуальний аналіз, аналіз даних, нейронні мережі, машинне навчання*

Вступ

У останні десятиліття швидка індустріалізація, зростання міст і посилення антропогенної діяльності призвели до значного погіршення якості атмосферного повітря, особливо в густонаселених районах. Наслідки для здоров'я від тривалого впливу забруднювачів повітря, таких як дрібні частинки ($PM_{2.5}$ та PM_{10}), двоокис азоту (NO_2), двоокис сірки (SO_2), озон (O_3) та чадний газ (CO), дедалі більше визнаються світовою науковою спільнотою. Ці забруднювачі пов'язані з низкою гострих і хронічних захворювань, включаючи респіраторні та серцево-судинні хвороби, негативні наслідки для новонароджених, порушення нейропсихічного розвитку та підвищену смертність.

У результаті органи охорони громадського здоров'я, екологічні агентства та міські планувальники стикаються з зростаючим тиском щодо впровадження науково обґрунтованих і технологічно ефективних стратегій зменшення ризиків для здоров'я населення, пов'язаних із забрудненням повітря [1].

Незважаючи на зростаючий обсяг епідеміологічних досліджень, що висвітлюють причинно-наслідкові зв'язки між забруднювачами повітря та станом здоров'я, впровадження інтелектуальних систем, здатних оцінювати ризики для здоров'я на рівні популяції, залишається недостатньо розвиненим. Існує критичний розрив між теоретичними моделями взаємодії здоров'я та навколишнього середовища та практичними застосуваннями, здатними обробляти гетерогенні дані в режимі реального часу, враховувати просторові та часові варіації й надавати персоналізовані або спільнотні оцінки ризику. Відсутність інтегрованих платформ, здатних агрегувати, аналізувати та інтерпретувати великі обсяги екологічних та медичних даних, ускладнює проактивне планування громадського здоров'я, комунікацію ризиків та цільові втручання [2].

У науковій літературі все частіше підкреслюється, що традиційні методи оцінки впливу забруднювачів на здоров'я населення, зокрема класичні статистичні моделі, обмежені у можливості враховувати складні нелінійні взаємозв'язки, просторово-часову динаміку забруднення та різнорідність даних про стан здоров'я. Класичні регресійні моделі часто не здатні коректно моделювати взаємодію між багатьма змінними, що обмежує точність прогнозів та адекватність рекомендацій для політики громадського здоров'я.

Однією з фундаментальних проблем у цій сфері є відсутність стандартизованих, високоякісних та відкритих наборів даних, які одночасно містять детальну інформацію про умови навколишнього середовища та параметри здоров'я на індивідуальному або популяційному рівні. Хоча існують окремі системи моніторингу навколишнього середовища та реєстри здоров'я, їх інтеграція часто ускладнюється різницею у структурі даних, роздільній здатності, доступності та обмеженнях конфіденційності. Крім того, багато наборів даних мають проблеми, такі як відсутні значення, непослідовні вимірювання, тимчасові невідповідності та обмежене географічне охоплення. Ці обмеження знижують здатність створювати надійні моделі, які могли б узагальнюватися на різні регіони та популяції, що в кінцевому результаті зменшує ефективність оцінки ризиків для здоров'я та прогнозів [3].

Ще однією великою перешкодою є складність збору, попередньої обробки та агрегування даних з кількох джерел. Дані про якість повітря можуть отримуватися з супутникових зображень, наземних моніторингових станцій або мобільних сенсорів, кожне з яких має різну просторову та часову роздільну здатність. Дані про здоров'я можуть надходити з лікарень, клінік, громадських опитувань або носимих пристроїв, часто у несумісних форматах і з різним ступенем надійності. Процес уніфікації цих потоків даних у узгоджені, аналізовані формати потребує значних обчислювальних та методологічних зусиль, включаючи очищення даних, створення ознак, нормалізацію та імпутацію. Крім того, етичні та правові аспекти, пов'язані з конфіденційністю даних та анонімізацією, ще більше ускладнюють отримання та використання даних про здоров'я на індивідуальному рівні у таких дослідженнях [4].

Сучасні дослідження демонструють значний потенціал застосування методів машинного та глибинного навчання для прогнозування ризиків, пов'язаних із забрудненням повітря. Нейронні мережі, ансамблеві моделі (Random Forest, XGBoost) та моделі послідовностей (LSTM) дозволяють ефективно інтегрувати великі масиви гетерогенних даних, моделювати складні залежності та враховувати як просторові, так і тимчасові варіації забруднення. Наприклад, у дослідженнях останніх років показано, що ансамблеві методи значно підвищують точність прогнозів смертності та захворюваності в порівнянні зі статистичними моделями, а згорткові нейронні мережі ефективні для аналізу зображень супутникового моніторингу та дистанційного оцінювання якості повітря.

Слід зазначити, що існує кілька ключових проблем, які залишаються невирішеними та ускладнюють впровадження інтелектуальних систем для оцінки ризиків для здоров'я:

– відсутність стандартизованих та відкритих наборів даних, що одночасно містять детальну інформацію про екологічні умови та параметри здоров'я населення. Незважаючи на наявність численних систем моніторингу повітря та медичних реєстрів, їх інтеграція часто утруднена через різницю у форматах, роздільній здатності, часових масштабах та обмеження конфіденційності. Це значно ускладнює створення узагальнюваних моделей, здатних працювати на різних регіональних рівнях;

– низька якість та неповнота даних. Багато наборів даних містять пропущені значення, непослідовні або несумісні вимірювання, тимчасові зсуви та обмежене географічне охоплення. Це призводить до необхідності складної попередньої обробки даних, включно з очищенням, нормалізацією, імпутацією та інженерією ознак;

– складність інтеграції даних із різних джерел. Дані про якість повітря можуть надходити з супутників, наземних станцій або мобільних сенсорів, а дані про здоров'я – із лікарень, клінік, опитувань або носимих пристроїв. Об'єднання цих потоків у єдиний формат для аналізу потребує значних обчислювальних ресурсів та методологічної компетенції;

– етичні та правові обмеження. Використання індивідуальних даних про стан здоров'я потребує дотримання правил конфіденційності та анонімізації, що додатково ускладнює проведення досліджень на рівні населення.

Незважаючи на ці проблеми, останні дослідження [3, 4, 5] демонструють, що інтелектуальні системи мають значний потенціал для підвищення ефективності оцінки ризиків, пов'язаних із забрудненням повітря. Вони можуть стати основою для:

– раннього попередження населення про небезпечні рівні забруднення;

– персоналізованого моніторингу впливу забруднення на здоров'я конкретних груп населення;

– прийняття рішень у сфері міського планування та екологічного регулювання;

– підвищення ефективності цільових медичних та соціальних втручань.

Інтелектуальні системи також можуть інтегруватися в концепцію «розумного міста», поєднуючи технологічні інновації з цілями забезпечення рівного доступу до здоров'я та сталого розвитку.

Таким чином, перспективи використання інтелектуальних систем для революціонізації оцінки ризиків для громадського здоров'я, пов'язаних із забрудненням повітря, є значними. Зі збільшенням доступності даних та розвитком методів моделювання ці системи можуть забезпечувати механізми раннього попередження, полегшувати персоналізоване відстеження експозиції та сприяти адаптивним стратегіям міського планування, надання медичної допомоги та екологічного регулювання.

Актуальність даного дослідження обумовлена поєднанням кількох факторів: зростанням глобальної урбанізації та антропогенного навантаження, високим рівнем забруднення повітря в міських агломераціях, недостатнім розвитком інтегрованих платформ для оцінки ризиків для здоров'я, а також потребою у впровадженні сучасних методів машинного та глибинного навчання для обробки гетерогенних даних. Проведений огляд наукових досліджень підтверджує, що створення інтелектуальної системи для оцінки ризиків здоров'я на основі даних про забруднення повітря є важливим і актуальним напрямом сучасної науки та практики громадського здоров'я.

У підсумку такі інтелектуальні платформи можуть стати ключовими компонентами інфраструктури «розумних міст», поєднуючи технологічні інновації з цілями забезпечення здоров'я та сталого розвитку. У цьому контексті метою даного дослідження є внесок у цю сферу шляхом розробки інтелектуальної системи, призначеної для оцінки рівнів ризику для здоров'я населення на основі впливу забруднення повітря [5].

1. Постановка проблеми та аналіз існуючих підходів та публікацій

З огляду на ці виклики, розробка та впровадження інтелектуальних систем для оцінки ризиків для здоров'я населення є не лише необхідністю, а й складним технічним завданням. Останні досягнення у сфері машинного навчання (ML) та глибинного навчання (DL) [6] відкривають перспективні можливості для вирішення складнощів, що виникають при моделюванні впливу забруднення повітря на здоров'я. Ці підходи здатні навчатися складним нелінійним взаємозв'язкам між вхідними змінними та результатами для здоров'я, інтегрувати різноманітні типи даних і адаптуватися до динамічних умов [7]. На відміну від традиційних статистичних методів, які часто потребують сильних апріорних припущень та обмежених взаємодій змінних, моделі машинного навчання можуть виявляти приховані закономірності у даних, автоматично визначати впливові ознаки та ефективно масштабуватися для роботи з високовимірними наборами даних.

Зокрема, глибинні нейронні мережі, архітектури згорткових та рекурентних мереж, механізми уваги та ансамблеві методи навчання [8] продемонстрували високі результати у задачах моделювання впливу

забруднення на здоров'я. Це включає прогнозування рівнів концентрацій забруднювачів, оцінку ризику експозиції, моделювання захворюваності залежно від екологічних факторів та навіть імітацію довгострокових ефектів політичних втручань. При інтеграції з геопросторовими даними, часовими рядами та соціодемографічними показниками такі моделі можуть надавати високоточні та контекстно-залежні оцінки вразливості населення та розподілу ризиків. Крім того, методи пояснюваного штучного інтелекту (XAI) все частіше застосовуються у цих системах для забезпечення прозорості та підтримки науково обґрунтованого прийняття рішень органами охорони громадського здоров'я.

Незважаючи на технологічні досягнення, залишаються значні перешкоди. Багато застосувань ML [9] у цій сфері досі перебувають на експериментальному рівні та часто базуються на обмежених або синтетичних наборах даних. Для реального впровадження потрібна надійна валідація, адаптація моделей до конкретних умов та узгодження з місцевими політичними та медичними системами.

Також необхідна міждисциплінарна співпраця між дата-сайентистами, епідеміологами, інженерами-екологами та фахівцями з громадського здоров'я, щоб моделі були не лише технічно досконалими, а й контекстно релевантними та етично обґрунтованими. Крім того, необхідно ретельно оцінювати інтерпретованість та справедливість AI-оцінок здоров'я, щоб уникнути потенційних упереджень та небажаних наслідків у розподілі ресурсів або комунікації ризику.

Наприклад, деякі дослідження [10] демонструють високу точність класифікації рівнів забруднення повітря за допомогою ансамблевих ML-моделей та нейронних мереж. Особливо відзначається застосування XGBoost та LSTM, які досягли високої точності прогнозування рівнів AQI (понад 90%). Сильними сторонами цих робіт є акцент на інтерпретованості моделей та порівняльна оцінка алгоритмів. Проте, незважаючи на технічну досконалість, такі дослідження обмежуються прогнозуванням якості повітря та не включають прямої оцінки наслідків для здоров'я. Насправді ж дослідження можуть зосереджуватися не лише на прогнозуванні рівнів забруднення, а й на розробці моделей, що безпосередньо оцінюють ризики для здоров'я населення.

Робота [11] пропонує більш прикладний підхід, пов'язуючи концентрації $PM_{2.5}$ та PM_{10} з кількістю амбулаторних звернень через респіраторні захворювання. Використовуючи багатошаровий перцептрон, автори демонструють можливість побудови надійних прогнозних моделей для конкретних вікових груп, що є важливим кроком у напрямку персоналізованої оцінки ризику. Сильними сторонами цього дослідження є використання реальних медичних даних із сезонною та демографічною стратифікацією. Водночас модель обмежена своєю «мілкою» архітектурою та обмеженим географічним охопленням. Інтелектуальна система може подолати ці обмеження шляхом впровадження більш просунутих архітектур, таких як

LSTM та механізми уваги, а також розширення географічного покриття та інтеграції додаткових прогнозних показників.

За результатами досліджень [12] можна відзначити гібридний підхід глибинного навчання, який поєднує просторові згорткові нейронні мережі, часові рекурентні модулі та механізми уваги для прогнозування концентрацій забруднювачів. Ключовим внеском цього дослідження є здатність моделі працювати з неповними даними та враховувати просторово-часові залежності. Водночас її застосування обмежується прогнозуванням забруднення без прямого зв'язку з наслідками для здоров'я.

У наступній роботі [13] запропоновано інноваційне застосування нейронних мереж U-Net як сурогатної моделі для швидкої оцінки впливу короткострокової експозиції $PM_{2.5}$ на здоров'я. Модель замінює ресурсоємні атмосферні симуляції умовною моделлю глибинного навчання, що забезпечує високошвидкісні обчислення та масштабованість. Основною перевагою цього підходу є ефективність та потенціал для використання в реальному часі. Проте дослідження фокусується на імітації фізичної моделі, а не на створенні комплексного конвеєра оцінки ризику для здоров'я. Тому, хоча метод технічно обґрунтований, він не інтегрує епідеміологічні або клінічні дані.

Інший підхід [14] використовує класичні GIS-методи для картографування забруднення повітря та оцінки ризику для здоров'я, застосовуючи супутникові дані $PM_{2.5}$. Завдяки просторовій інтерполяції та використанню функцій «експозиція–відповідь» автори створюють детальну карту експозиції для міських районів із рідкісними наземними сенсорами. Хоча перевагою цього методу є висока просторова роздільна здатність, він не враховує часові динаміки та не використовує методи ML для прогнозного моделювання. У сучасних інтелектуальних системах можна інтегрувати як просторові, так і часові виміри експозиції, а також застосовувати мультимодальні дані для моделювання ризиків для здоров'я населення за допомогою передових методів глибинного навчання [15].

Таким чином, можна підтвердити, що сучасні аналітичні системи широко використовують методи ML для попередньої обробки даних, виділення ознак та прогнозного моделювання. Моделі навчаються та валідуються на інтегрованих наборах даних, що поєднують дані екологічних сенсорів та індикатори здоров'я населення. Наша мета – побудувати масштабовану та інтерпретовану платформу, яка не лише прогнозує ризики для здоров'я з високою точністю, але й допомагає у формуванні цільових втручань та прийнятті політичних рішень [16].

Наукова новизна цієї роботи полягає у розробці end-to-end конвеєра для оцінки ризиків для здоров'я на основі даних, який вирішує проблеми гетерогенності, обмеженості та складності даних. Практичне значення визначається потенційною інтеграцією системи у реальне моніторинг громадського здоров'я та управління міським середовищем. Завдяки систематичній оцінці продуктивності та інтерпретованості моделей запропонований підхід прагне створити основу для ширшого застосування

інтелектуальних систем у сфері оцінки впливу навколишнього середовища на здоров'я та підтримки прийняття рішень у політиці.

Узагальнюючи аналіз наведених джерел слід зазначити, що останні дослідження у сфері інтелектуальних систем для оцінки впливу забруднення повітря на здоров'я демонструють значне різноманіття методологічних підходів та технічних рішень, що відображає розвиток машинного та глибинного навчання у цій галузі. Одним із перспективних напрямів є гібридні моделі глибинного навчання, які поєднують просторові згорткові нейронні мережі, часові рекурентні модулі та механізми уваги для прогнозування концентрацій забруднювачів. Ці моделі вирізняються здатністю працювати з неповними або частково відсутніми даними та одночасно враховувати просторово-часові залежності, що підвищує точність прогнозів і дозволяє моделювати динамічні процеси забруднення у складних урбанізованих середовищах. Водночас такі підходи залишаються обмеженими у зв'язку з відсутністю прямого зв'язку між прогнозованими рівнями забруднення та конкретними наслідками для здоров'я населення, що обмежує їхню практичну цінність у контексті громадського здоров'я.

Іншим напрямом є застосування нейронних мереж U-Net у якості сурогатних моделей для швидкої оцінки впливу короткострокової експозиції $PM_{2.5}$. Цей підхід дозволяє замінити ресурсоємні атмосферні симуляції умовною моделлю глибинного навчання, що забезпечує значне скорочення часу обчислень і масштабованість системи. Основним досягненням таких моделей є ефективність у реальному часі, що відкриває перспективи для оперативного моніторингу ризиків та швидкого реагування на зміну рівнів забруднення. Проте даний метод фокусується на імітації фізичної моделі забруднення, не інтегруючи безпосередньо епідеміологічні чи клінічні дані, що обмежує можливості комплексної оцінки ризиків для здоров'я на рівні популяції.

У більш класичних підходах, заснованих на геоінформаційних системах, для картографування забруднення та оцінки ризиків використовуються супутникові дані $PM_{2.5}$ у поєднанні з просторовою інтерполяцією та функціями «експозиція–відповідь». Ці методи дозволяють отримувати деталізовані карти експозиції для міських районів із обмеженою кількістю наземних сенсорів. Перевагою такого підходу є висока просторова роздільна здатність, що забезпечує локалізоване відображення рівнів забруднення та потенційного ризику. Разом з тим, метод не враховує часові динаміки та не застосовує алгоритми машинного навчання для прогнозного моделювання, що обмежує його здатність до передбачення майбутніх змін у забрудненні та відповідних наслідків для здоров'я.

Сучасні інтелектуальні системи намагаються поєднувати переваги цих різних підходів, інтегруючи просторові та часові виміри експозиції, а також мультимодальні дані про стан навколишнього середовища та індикатори здоров'я населення. Це дозволяє створювати прогностичні моделі високої точності, які здатні оцінювати ризики для здоров'я з урахуванням

комплексних взаємозв'язків між екологічними та соціальними факторами. Крім того, системи останнього покоління приділяють значну увагу інтерпретованості моделей, використовуючи методи пояснюваного штучного інтелекту, що забезпечує прозорість прийняття рішень та підтримує науково обґрунтоване формулювання політики громадського здоров'я.

Порівняльний аналіз робіт останніх років дозволяє визначити кілька ключових тенденцій та обмежень. Гібридні DL-підходи демонструють високі можливості для прогнозування концентрацій забруднювачів із врахуванням просторово-часових залежностей, але потребують інтеграції з даними про здоров'я для практичної оцінки ризику.

Сурогатні моделі U-Net забезпечують швидкі та масштабовані розрахунки, проте обмежені у зв'язку з відсутністю комплексного конвеєра оцінки здоров'я. Класичні GIS-підходи відзначаються високою просторовою роздільною здатністю та здатністю працювати з обмеженими даними сенсорів, але не враховують часову динаміку та не використовують ML для прогнозування. Інтеграція цих аспектів у сучасних інтелектуальних системах дозволяє формувати моделі, що одночасно забезпечують високоточне прогнозування забруднення, оцінку ризику для здоров'я та підтримку цільових заходів та політичних рішень.

Високий потенціал сучасних систем полягає у створенні масштабованих та інтерпретованих платформ, здатних працювати з інтегрованими наборами даних, що об'єднують інформацію з сенсорів навколишнього середовища та показники здоров'я населення. Наукова новизна таких підходів полягає у розробці end-to-end конвеєрів для оцінки ризиків здоров'я на основі даних, які враховують проблеми гетерогенності, обмеженості та складності інформації. Практичне значення визначається потенційною інтеграцією цих систем у реальні процеси моніторингу громадського здоров'я та управління міським середовищем, а також їхньою здатністю підтримувати обґрунтоване прийняття рішень на рівні політики.

В результаті аналізу джерел створена порівняльна таблиця основних підходів із декількома критеріями оцінки відображає переваги та обмеження різних методологій у контексті прогнозування забруднення та оцінки ризику для здоров'я, яка наведена нижче (таблиця 1).

Таким чином, сучасні наукові дослідження демонструють тенденцію до поєднання переваг різних методологій, прагнучи до створення систем, які забезпечують високоточну оцінку ризиків для здоров'я населення на основі комплексних даних про забруднення та соціально-демографічні фактори. Ключовим завданням залишається інтеграція прогнозування екологічних показників з безпосередньою оцінкою впливу на здоров'я та розробка інтерпретованих і масштабованих платформ, здатних підтримувати прийняття рішень у сфері громадського здоров'я та управління міським середовищем.

Таблиця 1

Порівняння основних підходів із декількома критеріями оцінки

Підхід	**Тип моделі**	**Облік просторово-часових залежностей**	**Прогнозування в реальному часі**	**Основні переваги**	**Основні обмеження**
Гібридні DL-моделі	CNN + RNN + Attention	Так	Обмежено	Робота з неповними даними, точне прогнозування концентрацій	Відсутність прямого зв'язку з наслідками для здоров'я
U-Net сурогат-ні моделі	Conditional DL	Обмежено	Так	Висока швидкість, масштабованість	Фокус на фізичній моделі, відсутність інтеграції медичних даних
GIS-підхід	Просторові інтерполяції + функції «експозиція–відповідь»	Частково	Ні	Висока просторова роздільна здатність, робота з обмеженими сенсорними даними	Не враховує часові зміни, відсутність ML-прогнозування
Сучасні інтегро-вані системи	ML + DL, мультимодальні дані	Так	Так	Висока точність прогнозів, інтеграція даних, інтерпретованість	Вимагає складної попередньої обробки та міждисциплінарної співпраці

2. Розробка концепції проекту

Запропонована архітектура інтегрує декілька моделей машинного (ML) та глибинного навчання (DL), кожна з яких виконує спеціалізовану функцію в межах ієрархічного конвеєра оцінки ризиків [17]. Такий підхід забезпечує не лише високу точність прогнозування, але й підвищену інтерпретованість результатів і адаптивність до різних умов даних та екологічних контекстів.

Початковий етап передбачає попередню фільтрацію та категоризацію даних за допомогою моделі дерева рішень (DT). Завдяки своїй інтерпретованості та ієрархічній логіці ця модель дозволяє ідентифікувати ключові закономірності та порогові значення показників забруднювачів, які

справляють першочерговий вплив на стан здоров'я. Цей етап також сприяє класифікації регіонів чи часових сегментів за рівнем забруднення, формуючи основу для диференційованої обробки на наступних етапах.

Другий етап зосереджений на виявленні складних нелінійних взаємозв'язків між показниками якості повітря та медико-демографічними характеристиками населення. Це завдання реалізується через машину опорних векторів (SVM), яка демонструє високу ефективність при роботі з гетерогенними ознаками та здатна будувати точні роздільні поверхні у багатовимірному просторі ознак. Вихідним результатом цього етапу є окреслення потенційних зон ризику та формалізація критичних умов, за яких вплив забруднювачів стає суттєво шкідливим для здоров'я.

На третьому етапі застосовується ансамблеве моделювання з використанням алгоритму випадкового лісу (RF), метою якого є агрегування попередніх результатів та уточнення структури факторів ризику шляхом повторної вибірки та побудови множини дерев рішень. Це підвищує стійкість системи до викидів та шумів, а також покращує надійність і відтворюваність оцінок ризику.

Четвертий етап передбачає використання моделі градієнтного бустингу XGBoost для посилення прогностичної спроможності системи. Інтегруючи додаткові часові, просторові та соціально-економічні змінні, раніше не враховані, ця модель поступово підвищує точність оцінки ризику завдяки ітеративній корекції помилок і ефективному опрацюванню складних міжфакторних залежностей.

На п'ятому етапі інтегрується згорткова нейронна мережа (CNN), яка аналізує просторово-часові закономірності, особливо у випадках, коли доступні геопросторові шари чи структури сенсорних мереж. Ця модель дозволяє системі враховувати географічну кореляцію забруднювачів та ідентифікувати просторові кластери ризиків для здоров'я, що є надзвичайно важливим для цільових втручань у сфері громадського здоров'я та стратегічного планування політики.

Фінальний етап реалізується через модель довгої короткострокової пам'яті (LSTM), яка відображає часову динаміку впливу забруднення на здоров'я. Завдяки здатності навчатися довгостроковим залежностям модель LSTM дозволяє системі прогнозувати відстрочені ефекти для здоров'я та моделювати сценарії ризику за різних стратегій екологічної політики.

Інтеграція цих моделей у багатоступеневу архітектуру формує комплексну, багатовимірну та стійку до помилок систему, здатну не лише точно оцінювати поточні ризики, але й прогнозувати їхню еволюцію за різних сценаріїв. Запропонована гібридна концепція вносить наукову новизну завдяки структурованому поєднанню інтерпретованих та глибинних моделей у єдину аналітичну систему, тим самим розширюючи можливості моніторингу, управління та зменшення ризиків для здоров'я населення, пов'язаних із забрудненням повітря.

Основним набором даних, обраним для нашого дослідження, є Air Pollution Image Dataset з Індії та Непалу [18]. Цей набір містить колекцію анотованих зображень, що відображають різні рівні забруднення повітря у міських і сільських середовищах. Його релевантність до поставленого завдання визначається багатим візуальним представленням умов забруднення, яке дозволяє навчати DL-моделі, зокрема CNN, для виявлення та класифікації рівня тяжкості забруднення. Це добре узгоджується з метою інтеграції просторово-контекстних характеристик у конвеєр оцінки ризику.

Для підвищення надійності та узагальнюваності вихідних результатів модель була додатково розширена та валідована за допомогою структурованих екологічних вимірювань із двох авторитетних джерел [19, 20]. Ці додаткові джерела дали змогу здійснити крос-валідацію рівнів забруднення, покращити калібрування порогів прогнозування та забезпечити узгодженість між ознаками, отриманими зі зображень, та числовими показниками забруднення. Це зміцнило цілісність та практичну застосовність запропонованої мультимодельної системи для оцінки ризиків здоров'я.

Проблема нашого дослідження може бути сформульована як завдання багатокласової класифікації. У ML під багатокласовою класифікацією розуміють задачу прогнозного моделювання, у якій алгоритм повинен віднести спостережуваний об'єкт до одного з кількох попередньо визначених класів. У контексті оцінки ризику для здоров'я, зумовленого забрудненням повітря, це завдання може бути математично формалізоване відповідним чином.

Вхідний набір даних у нашому випадку можна представити наступним чином: $X = \{x_1, x_2, ..., x_n\}$, де кожен об'єкт x_i є вектором ознак з простору $\Re^d$. Для кожного об'єкта X_i мітка класу збігається $y_i \in \{1,2,...,K\}$, де K — кількість класів (6 класів у рамках нашого завдання). Потрібно побудувати функцію $f : \Re^d \to \{1,2,...,K\}$, який для будь-якого вхідного об'єкта x передбачить мітку класу y (ризик для громадського здоров'я). Модель машинного навчання побудована з використанням навчальної вибірки $\{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$, яка може бути сформована з інформативних вхідних ознак, головне завдання полягає в знаходженні наближення функції f на основі цих даних. Якщо $P(y = k \mid x)$ - ймовірність належності об'єкта x класу k, тоді функція $f(x)$ прогнозує клас з апостеріорною ймовірністю

$$f(x) = \arg \max_{k \in \{1,2,...,K\}} P(y = k \mid x)$$

Для навчання моделі використовується функція втрат, яка кількісно визначає розбіжність між передбачуваними та справжніми мітками класів.

Однією з найчастіше використовуваних функцій втрат для цієї мети є втрата перехресної ентропії:

$$L(y, y') = -\sum_{k=1}^{K} y_k \log(y'_k)$$

де y_k — двійковий індикатор (0 або 1), що вказує, чи належить об'єкт до класу k, $y' = P(y = k \mid x)$ - ймовірність того, що об'єкт належить до класу k, передбачено моделлю. Модель оптимізується шляхом мінімізації функції втрат L використання методів оптимізації, таких як градієнтний спуск.

Фінальне прогнозування здійснюється шляхом вибору класу з найвищою ймовірністю на основі навченої моделі. Загальний конвеєр системи, використаний у дослідженні, представлений на рис. 1. Імпортовані набори даних завантажуються за допомогою функцій бібліотеки Pandas та зберігаються як об’єкти DataFrame. Далі виконується етап попередньої обробки, який включає виявлення аномалій і викидів, а також усунення дисбалансу класів у вихідній змінній.

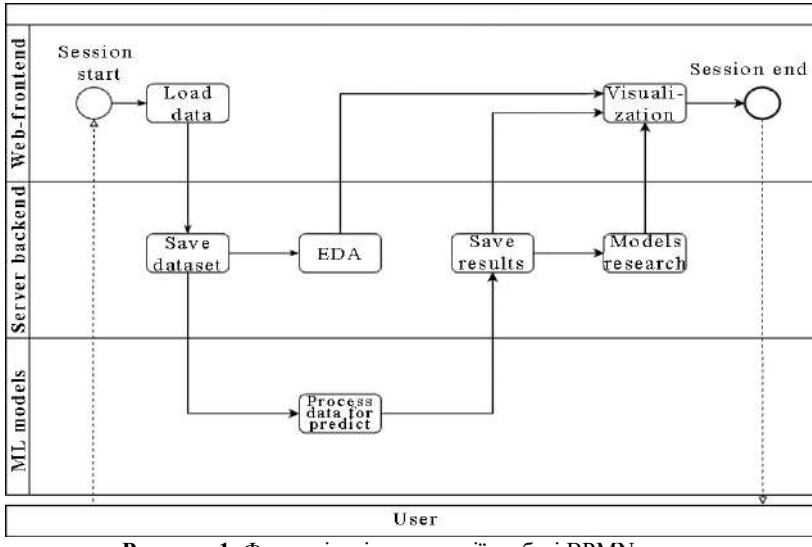


Рисунок 1. Формалізація концепції на базі BPMN схеми

Після попередньої обробки здійснюється комплекс процедур розвідувального аналізу даних (EDA). До них належать статистичний аналіз,

аналіз кореляцій, факторний аналіз, а також за необхідності - зменшення розмірності у випадках, коли кількість вхідних ознак є занадто великою.

На етапі моделювання будуються окремі ML-моделі, формується ансамбль моделей та розробляються глибокі повнозв'язні штучні нейронні мережі, навчені на тренувальних і тестових підмножинах даних. Ці підмножини формуються шляхом розподілу даних і підлягають процедурам крос-валідації для забезпечення надійності моделей.

Дисбаланс класів розв'язується за допомогою підходу зважування класів, коли ваги обчислюються як обернена частота появи класу у наборі даних. Це ефективно підвищує «штраф» за неправильну класифікацію недостатньо представлених класів, покращуючи чутливість моделі до міноритарних категорій. При цьому генерація синтетичних даних не застосовується, що позитивно впливає на надійність і валідність результатів класифікації.

Навчені моделі оцінюються за допомогою вибраних метрик продуктивності (наприклад, accuracy), тестуються на невідомих даних і серіалізуються у окремі файли для подальшого завантаження та застосування до нових наборів даних з метою оцінки рівня ризику для громадського здоров'я. Перед обробкою даних імпортуються необхідні бібліотеки. Серед них NumPy та Pandas для роботи зі структурами даних і попередньої обробки, Matplotlib та Seaborn для візуалізації, а також різні модулі бібліотеки scikit-learn (sklearn). Останні використовуються для таких завдань, як кодування категоріальних (рядкових чи текстових) ознак у числові формати, нормалізація даних, оцінка метрик продуктивності та ініціалізація моделей (наприклад, DecisionTreeClassifier).

У діаграмі класів (рис. 2) клас ExploratoryDataAnalysis відповідає за виконання розвідувального аналізу даних. Він містить атрибути для зберігання шляхів до директорій для збереження графічних результатів, а також набір методів для створення візуалізацій.

До таких методів належать generateDataDistributionPlot, generateEmissionIndexPlot, generateCorrelationMatrixPlot, generateZScorePlot, generatePairplotPlot та generateClassDistributionPlot.

Центральний файл server.py містить основну операційну логіку системи. У ньому визначені головні API-ендпоінти, включно з get_session (ініціалізація сесії), upload_file (завантаження даних), start_eda (запуск розвідувального аналізу), predict (інференція моделі) та compare (порівняння продуктивності моделей). Клас Regression відповідає за виконання регресійних моделей. Він містить атрибути для зберігання результатів прогнозування, а також серіалізованих моделей, серед яких decisionTreeModel.pkl, randomForestModel.pkl, XGBoostModel.pkl, mlpModel.keras, lstmModel.keras та cnnModel.keras. Клас забезпечує методи для масштабування даних (scaleData), генерації прогнозів (predict, predictProba) та створення різноманітних візуалізацій, зокрема generateAQIByLocationPlot, generateCorrelationMatrixPlot та generateAQIByTimePlot.

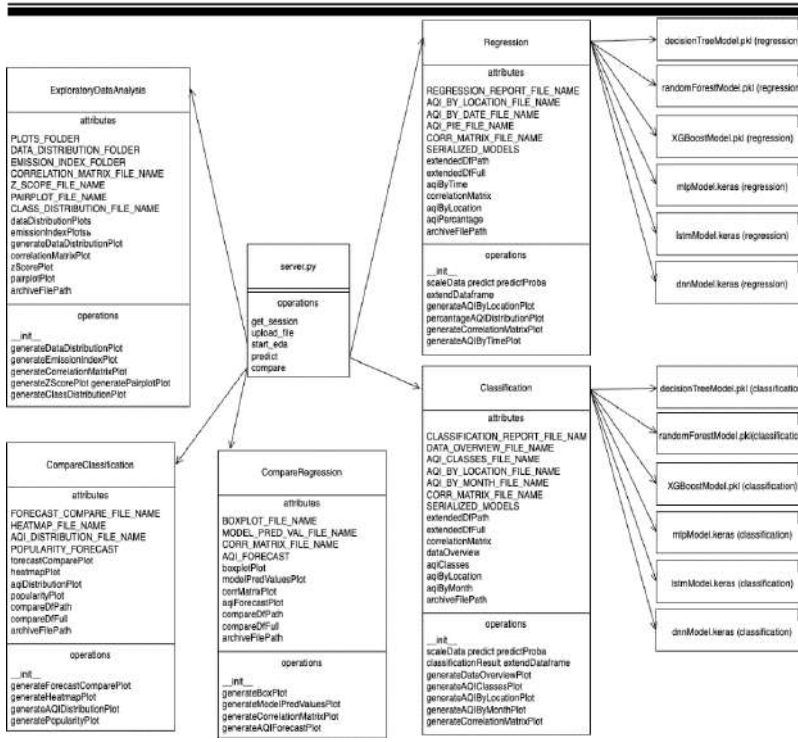


Рисунок 2. Діаграма головних класів проекту

Клас Classification, призначений для класифікаційних завдань, має подібну структуру до Regression, але зосереджується на віднесенні екологічних даних до наперед визначених класів. Він керує результатами класифікації та дозволяє розширювати вихідний датафрейм (classificationResult, extendDataframe). Для класифікації використовується той самий набір архітектур моделей: decisionTreeModel.pkl, randomForestModel.pkl, XGBoostModel.pkl, mlpModel.keras, lstmModel.keras та cnnModel.keras.

Класи CompareClassification та CompareRegression відповідають за порівняльний аналіз результатів класифікаційних та регресійних моделей відповідно. Вони надають широкий набір методів візуалізації, серед яких forecastComparePlot, heatmapPlot, compareDFPlot, boxplot, corrMatrixPlot та aqiForecastPlot, що забезпечують інтерпретованість результатів роботи моделей і дають змогу проводити міжмодельне бенчмаркінг-порівняння.

Інтелектуальна система побудована на клієнт-серверній архітектурі, де бекенд реалізовано на Python із використанням мікрофреймворку Flask, а фронтенд розроблено мовою JavaScript із застосуванням бібліотеки React.

Основний функціонал системи охоплює розвідувальний аналіз даних, прогнозне моделювання із використанням методів класифікації та регресії, а також порівняльну оцінку продуктивності моделей.

Бекенд базується на Flask, який забезпечує легковаговий і гнучкий каркас для побудови RESTful API, що дозволяє реалізувати безшовну комунікацію між фронтендом і серверними компонентами. Flask обробляє HTTP-маршрутизацію, яка відповідає за отримання, обробку та збереження екологічних даних. Структурна організація бекенд-проєкту представлена в наступному розділі (рис. 3).

Фронтенд системи реалізовано за допомогою фреймворку React, що забезпечує динамічний та зручний для користувача інтерфейс. Основні компоненти інтерфейсу включають сторінку завантаження файлів, модулі для відображення результатів розвідувального аналізу даних, сторінку прогнозування для запуску моделей та спеціальний розділ для порівняння моделей. Взаємодія з бекендом здійснюється через бібліотеку Axios, яка дозволяє виконувати HTTP-запити та забезпечує ефективну комунікацію з серверною частиною застосунку.

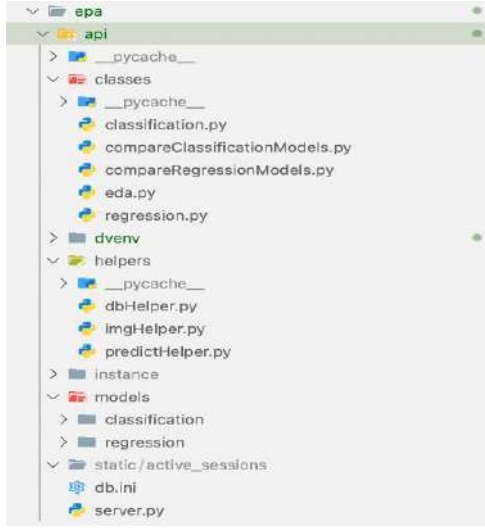


Рисунок 3. Структура проекту у середовищі розробки

3. Дослідження роботи та обчислювальні експерименти

На початковому етапі дані були імпортовані та збережені у вигляді структурованих об’єктів DataFrame за допомогою бібліотеки Pandas. Для оптимізації простору ознак було видалено непотрібні або неінформативні

атрибути, зокрема назви файлів, ідентифікатори року та місяця. До категоріальних змінних, таких як локація та година, було застосовано кодування міток (Label Encoding) з використанням LabelEncoder для перетворення текстових даних у числовий формат, сумісний із алгоритмами машинного навчання.

Далі датасет було піддано низці діагностичних процедур на наявність пропусків і аномалій. Записи з високою часткою відсутніх значень було вилучено, тоді як часткові пропуски заповнювалися методами середньої підстановки на основі атрибутної агрегації. Розподіли ключових забруднювачів ($PM_{2.5}$, PM_{10}, NO_2, SO_2, CO та O_3) були візуалізовані за допомогою гістограм та оцінок ядерної щільності для аналізу асиметрії, дисперсії та модальності. Також були використані boxplot-графіки для ідентифікації викидів у розрізі класів індексу якості повітря (AQI).

Для подолання проблеми дисбалансу класів у цільовій змінній (AQI_Class) було застосовано методи оверсемплінгу. Використовувалися алгоритми SMOTE та ADASYN, які синтетично збільшували кількість прикладів міноритарних класів, утворюючи збалансовані вибірки для навчання класифікаційних моделей. Отримані розширені датасети були експортовані для подальшого використання.

Було проведено кореляційний аналіз із використанням метрик Пірсона та Спірмена для оцінки лінійних і монотонних зв'язків між ознаками. Кореляційні матриці візуалізувалися у вигляді теплових карт, а значущість асоціацій додатково перевірялася статистичними тестами. Серед них t-тест Велча для порівняння значень AQI між парами класів, однофакторний дисперсійний аналіз (ANOVA) для багатокласових порівнянь і тест χ^2 для оцінки залежності між категоріями AQI та бінаризованими рівнями забруднення (наприклад, високий проти низького $PM_{2.5}$).

Таким чином, було виконано кореляційний аналіз ознак, а результат його візуалізації у вигляді теплової карти представлено на рис. 4. Для доповнення цих аналізів проведено виявлення викидів із використанням двох незалежних підходів = методу Z-оцінки та методу інтерквартильного розмаху (IQR). Теплові карти на основі Z-оцінки та boxplot-графіки на основі IQR надали додаткову інформацію про аномальні розподіли числових характеристик. Додатково було досліджено вимірну структуру датасету за допомогою факторного аналізу та методів зниження розмірності.

Факторний аналіз із використанням заданої трьохкомпонентної моделі виявив латентні конструкти, що лежать в основі даних про забруднювачі та значення ІСЯ (AQI). Завантаження (loadings) були візуалізовані у вигляді теплової карти для інтерпретації внеску змінних у різні фактори. Для зменшення простору ознак до двох вимірів було застосовано метод головних компонент (PCA) та t-розподілене стохастичне сусіднє вбудовування (t-SNE). Трансформовані уявлення ознак були візуалізовані для оцінки відокремлюваності класів та внутрішньої структури даних.

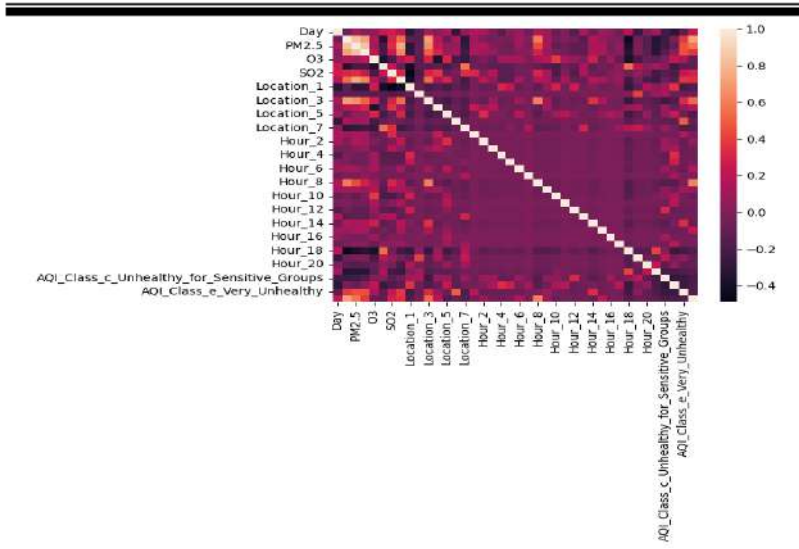


Рисунок 4. Теплова карта кореляційних оцінок між вхідними ознаками набору даних

Як було виявлено під час кореляційного аналізу, окрім неінформативної ознаки, що представляє ім'я файлу зображення, високі значення кореляції особливо спостерігаються серед атрибутів, що характеризують забруднення повітря шкідливими речовинами, зокрема $PM_{2.5}$ та PM_{10}. Це пояснюється природою їх вимірювання та схожістю інструментів і методологій, використаних для їхнього виявлення. У рамках попереднього аналізу даних, їх очищення та попередньої обробки було прийнято рішення видалити атрибут Filename із DataFrame, оскільки він не несе аналітичної цінності в контексті досліджуваного завдання.

Під час подальшого дослідження було встановлено, що ознаки Month і Year демонструють сильну міжкореляцію та роблять непослідовний внесок у структуру даних. У зв'язку з цим їх було виключено з фінального набору ознак, щоб уникнути потенційних проблем мультиколінеарності та зменшити надлишковість. Під час реалізації аналізу пропущених значень за допомогою методу isnull() було виявлено, що ознаки O_3, CO, SO_2 та NO_2 містять понад 2000 пропущених записів. Для розв'язання цієї проблеми було застосовано метод імпутації шляхом обчислення середнього значення відповідних змінних і заміни пропусків за допомогою функції mean(), що дозволило відновити цілісність набору даних і мінімізувати втрати інформації.

Розглянемо процес EDA та аналізу даних у межах локально розгорнутої системи з точки зору кінцевого користувача (рис. 5).

Розділ завантаження даних надає короткий опис базових вимог до вхідного файлу, який користувач може завантажити, а також містить

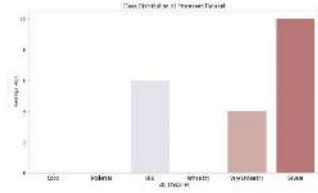
інтерактивну кнопку «Upload file». Після завантаження користувачеві відображається назва та характеристики обраного файлу для перевірки його коректності. Після підтвердження шляхом натискання кнопки «Confirm data» інтерфейс переходить до секції аналізу даних.

Location	Year	Month	Day	Hour	PM2.5	PM10	O3	CO	SO2	NO2	AQI	AQI_Categories
0	2023	2	23	6	43.0	78.0	26.0	258.0	10.0	17.0	5	Severe
2	2023	2	22	3	500.0	450.0	91.0	78.0	17.0	47.0	5	Severe
5	2022	10	2	0	185.0	199.0	10.0	52.0	12.0	26.0	4	Very Unhealthy
2	2023	2	16	3	401.0	325.0	73.0	89.0	16.0	57.666767894736842	5	Severe
4	2023	3	23	5	14.0	41.0	35.0	6.0	5.0	7.0	2	USG

Download CSV report

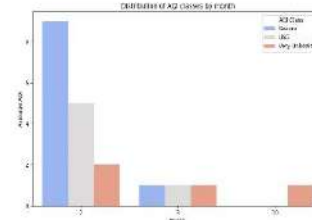
AQI Class Distribution

An AQI Class Distribution plot visualizes the frequency or proportion of different Air Quality Index (AQI) categories (e.g., Good, Moderate, Unhealthy) in a dataset, helping to understand air quality patterns.



Distribution of AQI classes by time

The Distribution of AQI Classes by Month plot shows how air quality index (AQI) categories (e.g., Good, Moderate, Unhealthy) vary across different months, helping to identify seasonal trends and pollution patterns.

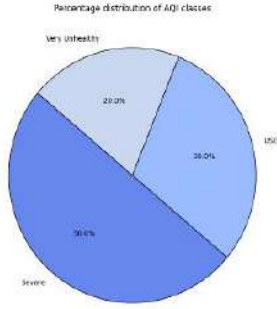


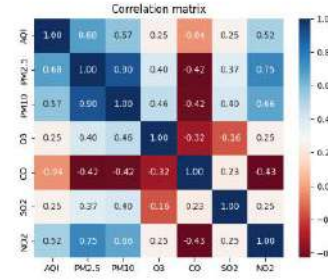
Grouping data by AQI class

Grouping data by AQI class means categorizing air quality data into predefined AQI ranges (e.g., Good, Moderate, Unhealthy) to analyze trends, compare distributions, and extract meaningful insights.

Correlation between variables

A Correlation Between Variables plot visually represents the strength and direction of relationships between multiple variables, often using a heatmap or scatter plots to identify patterns and dependencies in the data.





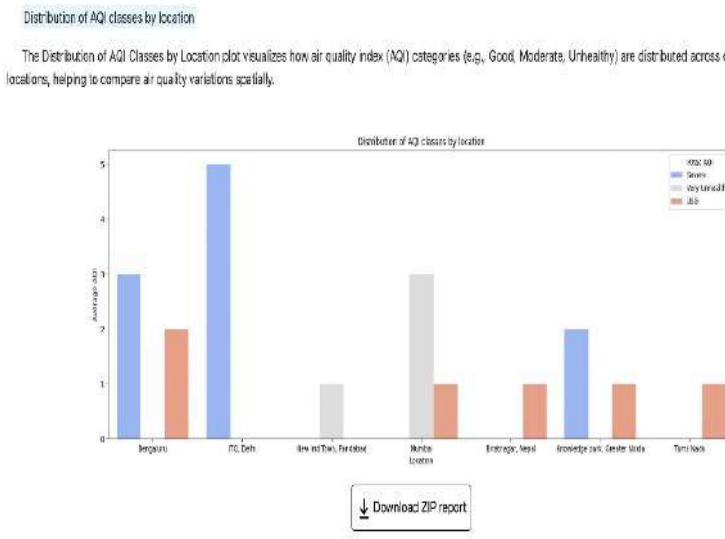


Рисунок 5. Інтерфейс користувача аналітичної панелі EDA системи

На стороні бекенду інтегровано спеціалізований модуль розвідувального аналізу даних (EDA) для обробки завантажених вхідних даних. Цей модуль забезпечує візуалізацію статистичних характеристик набору даних, побудову кореляційних матриць, графіків розподілу та реалізацію інших методів EDA. Для його імплементації використано відомі бібліотеки, зокрема Pandas, Matplotlib, Keras та Seaborn. REST API ендпоінти системи розроблені таким чином, щоб повертати аналітичні результати у вигляді зображень, які потім відтворюються у фронтенд-інтерфейсі для інтерпретації користувачем.

Для прогнозного моделювання система включає як класифікаційні, так і регресійні алгоритми, зокрема DT, SVM, RF, XGBoost, LSTM, CNN. Ці моделі попередньо навчені та збережені у серіалізованому форматі (.pkl або .keras) у файловій системі сервера. Під час ініціювання запиту бекенд-застосунок на базі Flask десеріалізує відповідну модель, обробляє підготовлені вхідні дані та повертає результат прогнозування у вигляді структурованої JSON-відповіді.

Для запуску прогнозування користувач спочатку обирає бажану модель та її тип - класифікація чи регресія. Після вибору стає доступною кнопка «Start Prediction». При її активації формується запит до бекенду для запуску обраного прогнозного конвеєра. Після генерації результатів система надсилає відповідь до фронтенду, що містить як візуальні результати (наприклад, графіки), так і детальний CSV-звіт. Зміст результатів залежить від обраного типу моделі, що відображає відмінності у логіці алгоритмів та структурі вихідних даних.

Під час оцінювання моделей машинного навчання набір даних було розділено на навчальну та тестову вибірки у співвідношенні 75:25 для підтримки процесу розробки моделей та оцінки їх продуктивності. Для структурованого логування ефективності моделей обчислені значення метрик зберігалися у спеціалізованих змінних.

Для забезпечення комплексного порівняльного аналізу ефективності класифікацій за точністю застосовувалися візуалізації матриць плутанини, згенеровані за допомогою бібліотеки Seaborn. Отримані матриці плутанини для всіх реалізованих моделей ML наведено на рис. 6. Для зручності інтерпретації вихідні класи ризику були чисельно закодовані у послідовному порядку від 0 до 5.

Узагальнені результати продемонстрували високу точність класифікації для всіх моделей, причому найвищу ефективність за показником точності класифікації продемонструвала модель XGBoost. Для подальшого порівняльного аналізу була створена візуалізація, яка ілюструє точність класифікації всіх моделей у багатокласовому середовищі. Усереднені профілі ROC-кривих для моделей наведено окремо на рис. 7.

Представлена ROC-крива демонструє продуктивність класифікації різних моделей у багатокласовому середовищі з використанням підходу one-vs-rest. Графік відображає співвідношення між показником True Positive Rate (TPR) та False Positive Rate (FPR) для кожного класифікатора. Чим ближче крива розташована до верхнього лівого кута, тим ефективніше модель розрізняє цільові класи. Серед усіх оцінених моделей найбільш сприятливу поведінку демонструє класифікатор XGBoost, ROC-крива якого максимально наближена до оптимальної межі. Це свідчить про стабільно високий рівень чутливості та специфічності в усьому діапазоні порогів прийняття рішень.

Модель LSTM також показала майже ідеальні результати, демонструючи плавну криву, що залишається близькою до верхнього лівого кута графіка, що свідчить про її сильні можливості до узагальнення. Модель CNN працює порівняно добре, хоча її крива демонструє незначні відхилення від оптимальної форми на початковому діапазоні FPR. Проте її траєкторія загалом підтверджує надійну якість класифікації. Модель RF показала помірно якісний ROC-профіль, однак її крива має більш східчастий характер, що можна пояснити її чутливістю до взаємодії ознак та розподілу даних.

Як видно, форма ROC-кривих варіюється в різних ділянках розподілу балів, проте ансамблеві моделі - насамперед Random Forest та XGBoost - виявляють найбільш наближені до оптимальних характеристики з плавними кривими, що тяжіють до значень, близьких до 1.

У порівнянні з цим, модель DT демонструє помітно нижчу ефективність. Її крива характеризується різкими змінами і розташована значно нижче кривих більш просунутих моделей, особливо у ділянках, де False Positive Rate (FPR) перевищує 0,2. Такий результат відображає обмеження базових структур дерев рішень у розпізнаванні складних меж класів у завданнях багатокласової класифікації.

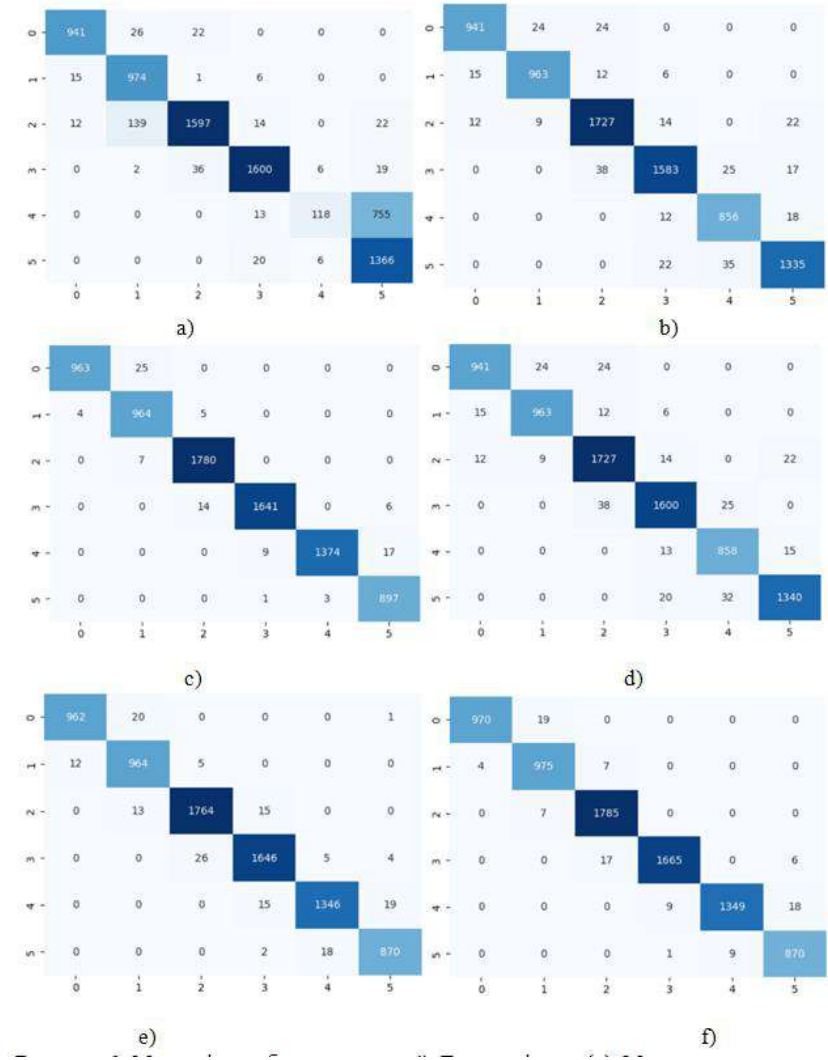


Рисунок 6. Матриці похибок для моделей: Дерево рішень (a), Метод опорних векторів (b), Випадковий ліс (c), XGBoost (d), Рекурентна нейронна мережа (e), Згорткова нейронна мережа (f)

Модель SVM показує нестабільний ROC-профіль із нерегулярним зростанням TPR, що свідчить про непослідовну продуктивність класифікації та знижену дискримінаційну здатність порівняно з ансамблевими або глибокими нейронними методами. Загалом результати підтверджують, що ансамблеві моделі, зокрема XGBoost, та глибокі нейронні архітектури, такі як LSTM та CNN, перевершують класичні методи за оцінкою на основі ROC-кривих. Ці моделі демонструють вищу здатність до узагальнення та підтримують збалансовану продуктивність між класами. Результати порівняльного аналізу метрик створених моделей в інтелектуальній системі представлені у таблиці 1.

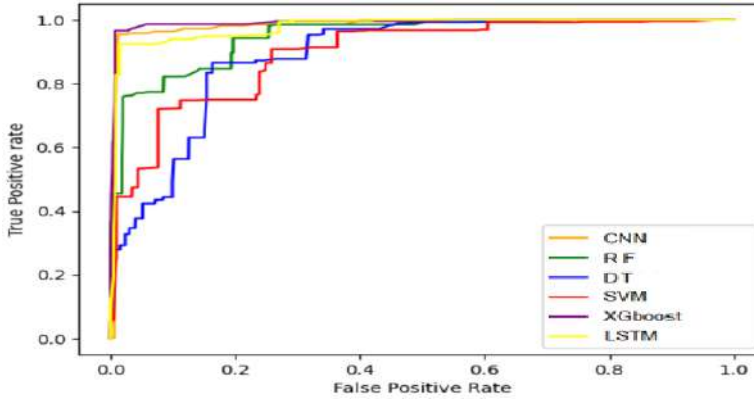


Рисунок 7. Узагальнена ROC-крива для створених моделей

Таблиця 2

Результати порівняльного аналізу метрик створених моделей

Metric	DT	SVM	RF	XGB	CNN	LSTM
Accuracy	0,879	0,961	0,973	0,993	0,987	0,981
Precision	0,861	0,957	0,967	0,991	0,981	0,973
Recall	0,856	0,958	0,969	0,989	0,977	0,969
F1-score	0,864	0,952	0,967	0,988	0,979	0,977
t_{avg}	3	18	7	8	45	67

Загалом, нейронні мережі демонструють високу точність класифікації. Модель LSTM досягає рівня точності, близького до 0,98, приблизно до 15-ї

епохи, тоді як CNN досягає аналогічного рівня лише після 25-ї епохи. Після цього подальше покращення точності значно сповільнюється, а іноді спостерігаються незначні коливання. Такі динаміки вказують на потенційний ризик перенавчання; однак обрані параметри регуляризації ефективно зменшують цю проблему. На початковому етапі навчання CNN демонструє більші значення помилки порівняно з LSTM. Незважаючи на це, її швидкість збіжності значно вища, що свідчить про більш ефективну траєкторію оптимізації, незважаючи на початкову варіативність.

Був проведений порівняльний аналіз моделей не лише за стандартними метриками оцінки, а й з урахуванням обчислювальної продуктивності. Крім точності класифікації, розглядався аспект обчислювальної складності шляхом вимірювання часу навчання моделей. Для цього було введено додатковий показник (tavg), що відображає середній час навчання у секундах.

4. Висновки

У результаті дослідження можна зазначити, що порівняльна оцінка реалізованих моделей продемонструвала явну перевагу ансамблевих та глибоких методів навчання. Класифікатор XGBoost досяг найвищої загальної точності — 0,993, з відповідними значеннями precision 0,991, recall 0,989 та F1-score 0,988. Серед моделей глибокого навчання CNN і LSTM також показали високий рівень продуктивності: CNN досягла точності 0,987, а LSTM - 0,981. Ці результати свідчать про здатність системи правильно класифікувати рівень ризику для здоров'я у понад 98% випадків при використанні найефективніших архітектур.

Окрім точності класифікації, була оцінена обчислювальна ефективність через середній час навчання моделей. Простіші моделі, такі як DT та RF, продемонстрували нижчий час навчання (відповідно 3 та 7 секунд), тоді як моделі глибокого навчання потребували більше ресурсів: CNN у середньому 45 секунд, а LSTM — 67 секунд. Ця різниця підкреслює типовий компроміс між складністю моделі та швидкістю виконання. Проте вища прогностична продуктивність моделей глибокого навчання виправдовує їх застосування, особливо у сценаріях, де критично важливі точність та надійність.

Аналіз надійності за допомогою матриць плутанини та ROC-кривих підтвердив стабільність прогнозів для всіх шести класів ризику. ROC-крива XGBoost послідовно наближалася до оптимальної форми, що відображає високу чутливість і специфічність. Подібно до цього, моделі LSTM та CNN демонстрували майже ідеальну поведінку з мінімальними відхиленнями. У той же час традиційні моделі, такі як DT та SVM, виявили обмеження продуктивності, особливо при обробці складних меж класів та підтриманні послідовності у різних ділянках розподілу балів.

З точки зору системної архітектури, поетапна структура значною мірою сприяла підвищенню надійності та інтерпретованості всього конвеєра. Використання інтерпретованих моделей на ранніх етапах дозволяло ефективно сегментувати дані та оцінювати порогові значення, тоді як глибокі

архітектури на пізніших етапах забезпечували захоплення просторово-часових залежностей і віддалених ефектів забруднювачів на здоров'я. Залучення зовнішніх наборів даних гарантувало вищу узагальнюваність і послідовність прогнозів моделей.

Перспективи подальшого розвитку включають інтеграцію механізмів уваги та трансформерів для покращення моделювання послідовностей, включення потоків даних із сенсорів у режимі реального часу та застосування федеративного навчання для оцінки ризику здоров'я з урахуванням приватності. Крім того, розширення функціоналу системи для персоналізованої оцінки ризику на рівні окремої особи та моделювання довгострокових епідеміологічних тенденцій ще більше підвищить її практичну цінність для планування громадського здоров'я та підтримки політики.

5. Література

1. Zhang, X., Liu, Y., Yang, H., & Sun, W. (2020). DeepAir: A hybrid deep learning framework for air quality forecasting. *Environmental Modelling & Software, 134***,** 104844. https://doi.org/10.1016/j.envsoft.2020.104844

2. Yao, S., Shen, M., Chen, Q., & Wang, Y. (2022). Deep learning for the prediction of population exposure to PM2.5 using remotely sensed data. *Environment International, 158***,** 106899. https://doi.org/10.1016/j.envint.2021.106899

3. Li, J., Yuan, J., & Wang, Z. (2020). Urban air pollution mapping using a deep learning approach and street-level images**.** *Environmental Pollution, 266***,** 115251. https://doi.org/10.1016/j.envpol.2020.115251

4. Fang, L., Yang, L., & Wang, C. (2019). Air quality index forecasting using a deep learning model based on 1D convolutional neural network and bidirectional long short-term memory. *Science of the Total Environment, 646,* 400–409. https://doi.org/10.1016/j.scitotenv.2018.07.400

5. Chen, F., Cao, S., Wang, Y., & Anwar, M. (2021). An interpretable machine learning framework for urban air quality analysis: A case study of PM2.5 in China. *Ecological Indicators, 124***,** 107447. https://doi.org/10.1016/j.ecolind.2021.107447

6. Rudnichenko, N., Vychuzhanin, V., Shibaeva, N., Petrov, I., & Otradskya, T. (2023). Intelligent data clustering system for searching hidden regularities in financial transactions. In *Proceedings of the 11th International Conference "Information Control Systems & Technologies" (ICST-2023)* (Vol. 3513, pp. 163–176). CEUR-WS.

7. Vychuzhanin, V., Rudnichenko, N., Sagova, Z., Smieszek, M., Cherniavskyi, V. V., Golovan, A. I., & Volodarets, M. V. (2019). Analysis and structuring diagnostic large volume data of technical condition of complex equipment in transport. *IOP Conference Series: Materials Science and Engineering, 776,* 012049. https://doi.org/10.1088/1757-899X/776/1/012049

8. Rudnichenko, N., Vychuzhanin, V., Petrov, I., & Shibaev, D. (2020). Decision support system for the machine learning methods selection in big data mining. In *Proceedings of the Third International Workshop on Computer Modeling and Intelligent Systems (CMIS-2020)* (Vol. 2608, pp. 872–885). CEUR-WS.

9. Rudnichenko, N., Vychuzhanin, V., Otradskya, T., Petrov, I., & Shpinareva, I. (2023). Hybrid intelligent system for recognizing biometric personal data. In *Proceedings of the 3rd International Workshop on Computational & Information Technologies for Risk-Informed Systems (CITRisk 2022), co-located with the XXII International Scientific and Technical Conference on Information Technologies in Education and Management (ITEM 2022)* (Vol. 3422, pp. 74–85). CEUR-WS.

10. Lipatova, E., & Potapchenko, I. (2025). Machine learning models for air pollution health risk assessment: A comparative study. *Procedia Computer Science, 230*, 1–9. https://doi.org/10.1016/j.procs.2025.01.005

11. Chen, C.-H., Wang, C.-Y., & Wu, M.-Y. (2018). Machine learning techniques for short-term prediction of outpatient visits for respiratory diseases based on air pollution data. *Atmospheric Environment, 182*, 200–208. https://doi.org/10.1016/j.atmosenv.2018.04.051

12. Wang, Y., Lu, L., & Zhang, Q. (2018). Deep inferential spatial–temporal network for air pollution forecasting. *Neurocomputing, 322*, 204–213. https://doi.org/10.1016/j.neucom.2018.09.016

13. Lee, H., Kwon, S., & Kim, S. (2024). Rapid PM2.5-induced health impact assessment using a conditional U-Net CMAQ surrogate model. *Environmental Modelling & Software, 170*, 105643. https://doi.org/10.1016/j.envsoft.2023.105643

14. Zhang, J., Xu, Y., & Chen, L. (2022). PM2.5 exposure and health risk assessment using remote sensing and GIS: A case study in Shanghai. *Ecological Indicators, 140*, 109064. https://doi.org/10.1016/j.ecolind.2022.109064

15. Yao, S., Shen, M., Chen, Q., & Wang, Y. (2022). Deep learning for the prediction of population exposure to PM2.5 using remotely sensed data. *Environment International, 158*, 106899. https://doi.org/10.1016/j.envint.2021.106899

16. Bai, X., Zhang, N., Cao, X., & Chen, W. (2024). Prediction of PM2.5 concentration based on a CNN-LSTM neural network algorithm. *PeerJ, 12*, e17811. https://doi.org/10.7717/peerj.17811

17. Koçak, E. (2025). Comprehensive evaluation of machine learning models for real-world air quality prediction and health risk assessment by AirQ+. *Earth Science Informatics, 18*, 447–463. https://doi.org/10.1007/s12145-025-01941-7

18. Kaggle. (n.d.). *Air pollution image dataset from India and Nepal***.** Retrieved from https://www.kaggle.com/datasets/adarshrouniyar/air-pollution-image-dataset-from-india-and-nepal

19. U.S. Environmental Protection Agency. (n.d.). *Air Quality System (AQS)***.** ***Retrieved from https://www.epa.gov/aqs***

20. Figshare. (n.d.). *Air pollution data for Seoul***.** Retrieved from https://figshare.com/articles/dataset/CO_csv/12805643

INTELLIGENT HYBRID SYSTEM PROJECT FOR BIG DATA VOLUMES HEALTH HARM RISKS ANALYSIS

Ph.D. M. Rudnichenko[1[0000-0002-7343-8076]], **Ph.D. N. Shibaeva**[1[0000-0002-7869-9953]],
Ph.D. T. Otradskya[2[0000-0002-5808-5647]], **D. Shvedov**[1[0009-0002-4823-8782]],
Ph.D. I. Shpinareva[1[0000-0001-9208-4923]], **Dr.Sci. I.Petrov**[3[0000-0002-8740-6198]]
[1]Odessa Polytechnic National University, Ukraine,
[2]Odesa College of Computer Technologies "SERVER", Ukraine,
[3]National University "Odessa Maritime Academy", Ukraine
EMAIL: nickolay.rud@gmail.com, nati.sh@gmail.com, tv_61@ukr.net, firmn@gmail.com, frumle@ukr.net, iryna.shpinareva@onu.edu.ua

This paper presents a comprehensive study on the design and implementation of an intelligent system for analyzing and assessing population health risks associated with large-scale environmental data, with a focus on air pollution. The system leverages a hybrid architecture integrating machine learning and deep learning models to process heterogeneous datasets, including high-volume sensor measurements and satellite-derived indicators. The relevance of this research is underscored by the growing health threats posed by atmospheric pollutants such as PM2.5, PM10, NO_2, SO_2, CO, and O_3, particularly in densely populated urban areas. The study emphasizes the necessity of employing intelligent data-driven systems to deliver accurate, scalable, and interpretable assessments of health risks at the population level. A critical review of existing methodologies highlights the limitations of traditional statistical approaches while demonstrating the advantages of neural networks and ensemble learning techniques in modeling complex nonlinear relationships and spatiotemporal dependencies within large, heterogeneous datasets. The proposed system incorporates a six-stage modeling framework, comprising decision trees, support vector machines, random forests, XGBoost, convolutional neural networks, and long short-term memory networks. Each stage contributes to a structured pipeline for data preprocessing, classification, prediction, and risk stratification. Training and validation were performed using the Air Pollution Image Dataset from India and Nepal, complemented by structured air quality datasets to enhance model generalizability. The system architecture follows a client-server design, with a Python and Flask backend and a JavaScript-based frontend, featuring an interactive analytical dashboard for data visualization, model inference, and comparative evaluation. Experimental results demonstrate the efficacy of ensemble and deep learning models, with a comparative analysis across accuracy, precision, recall, F1-score, and average training time confirming their superior performance in large-scale health risk assessment.

Keywords: *Intelligent data analysis, health analysis, big data, hybrid data mining, data analysis, neural networks, machine learning*

УДК 004.056.55:004.738.5:004.272
DOI https://doi.org/10.36059/978-966-397-538-2-6

РЕСУРСООЩАДНИЙ ПІДХІД ДО ПОБУДОВИ SEQURE BOOT У ВБУДОВАНИХ СИСТЕМАХ ІНТЕРНЕТУ РЕЧЕЙ

Ph.D. І. Розломій[1][0000-0001-5065-9004], **Ph.D. А. Ярмілко** [2][0000-0003-2062-2694],
С. Науменко[2] [0000-0002-6337-1605]

[1]*Черкаський державний технологічний університет, Україна,*
[2]*Черкаський національний університет імені Богдана Хмельницького, Україна*
EMAIL: inna-roz@ukr.net, a-ja@ukr.net, naumenko.serhii1122@vu.cdu.edu.ua

Анотація. *У роботі представлено ресурсоощадний підхід до побудови механізму Secure Boot, спеціально адаптованого для вбудованих систем Інтернету речей (IoT), що функціонують в умовах обмежених апаратних ресурсів. Запропонована архітектура враховує критичні обмеження щодо обсягу пам'яті, відсутності апаратної підтримки криптографії та необхідності мінімізації енергоспоживання. Класичні моделі безпечного завантаження малопридатні до таких умов, тому розроблено полегшений варіант Secure Boot з використанням полегшених хеш-функцій, зокрема SPONGENT-128/128/8, що забезпечує базовий рівень автентичності та цілісності програмного забезпечення при мінімальних витратах обчислювальних ресурсів. Реалізація на мікроконтролерах STM32 та ESP8266 продемонструвала високу точність виявлення модифікацій прошивки, енергоефективність (менше 11 мкДж на перевірку) і затримку запуску не більше 30 мс. Розглянуто питання безпечного зберігання еталонного хешу, використання цифрового підпису на основі ECDSA та варіанти масштабування під різні апаратні конфігурації. Запропонований підхід є перспективним для інтеграції в медичні, промислові, військові та інші прикладні IoT-рішення, де важливо забезпечити баланс між безпекою, енергоефективністю та швидкодією. У завершенні окреслено напрями подальших досліджень – впровадження обфускації, захищених OTA-оновлень та підтримки новітніх криптографічних примітивів.*

Ключові слова. *IoT; полегшений Secure Boot; вбудоване програмне забезпечення; криптографічний хеш; SPONGENT; енергоефективність*

1. Постановка проблеми

Інтернет речей (IoT) є одним з ключових напрямів цифрової трансформації у XXI столітті, який змінює уявлення про інформаційні технології та системи керування. Замість централізованих обчислювальних ресурсів у центрі уваги опиняються малі, автономні, часто розподілені пристрої, що інтегруються в реальне середовище та забезпечують моніторинг, керування та передачу даних. Такий підхід відкриває широкі перспективи для автоматизації

життєвих сфер – від «розумного дому» до медичних імплантів, військових платформ та інфраструктур критичного значення. У центрі цих змін – вбудовані мікроконтролери, що працюють у реальному часі та використовують спеціалізоване програмне забезпечення. Незважаючи на свою компактність, ці пристрої здійснюють обробку чутливих даних, керують фізичними процесами й можуть мати прямий вплив на безпеку людини або об'єктів [1].

Програмне забезпечення, зашите у вбудованих пристроях, є фактично їхнім «мозком» – воно визначає функціональність, алгоритмічну логіку, методи обміну даними, реагування на зовнішні сигнали та поведінку пристрою в нестандартних ситуаціях [2]. Втрата довіри до цього компонента призводить до ризиків, які виходять далеко за межі втрати даних. Компрометація мікропрограми може відкрити зловмиснику повний контроль над пристроєм: змінити його логіку, відключити захисні механізми, реалізувати приховані комунікаційні канали або сформувати елемент розподіленої бот-мережі. Це робить захист коду завантаження – особливо на початкових етапах запуску – критично важливою проблемою в контексті IoT-безпеки [3].

Початкова фаза завантаження є найбільш вразливою. Саме в цей момент пристрій ще не активував стандартні механізми захисту: немає шифрування трафіку, відсутні апаратні бар'єри доступу, не працює захист пам'яті. Будь-яке втручання в процес початкового завантаження може бути невидимим для користувача та системних моніторів, що робить подальшу компрометацію глибокою і довготривалою. У традиційних обчислювальних системах (ПК, сервери, мобільні пристрої) цю проблему вирішує механізм Secure Boot – концепція, відповідно до якої жоден код не може бути виконаний до його перевірки на цілісність і автентичність [4, 5, 6]. Як правило, це досягається через вбудовані засоби криптографії, цифрові підписи, зберігання еталонних хешів у захищених зонах та апаратне забезпечення (TPM, HSM або SoC з Secure Boot ROM) [7].

Проте ситуація суттєво змінюється, коли йдеться про IoT-пристрої. Більшість з них побудовано на основі недорогих мікроконтролерів (типу STM32F1, ESP8266, AVR, PIC), які не мають апаратної підтримки криптографії, обмежені у доступній пам'яті (кілька десятків або сотень КБ Flash та RAM), живляться від батарей або енергозалежних джерел і працюють в агресивних середовищах [8, 9, 10]. У таких умовах реалізувати класичну архітектуру Secure Boot практично неможливо. Обмеження за розміром коду, відсутність прискорювачів криптографічних обчислень, жорсткі вимоги до енергоспоживання та швидкодії – усе це змушує шукати альтернативні, ресурсоощадні рішення.

Водночас світова тенденція свідчить про стрімке зростання кількості IoT-пристроїв у мережах різного масштабу – від домашніх до промислових. Це створює нові ризики: атаки на прошивки стають усе більш популярними серед кіберзлочинців, адже проникнення в один слабко захищений вузол може

призвести до ланцюгової реакції в усій системі. Саме тому потреба у захищеному завантаженні в IoT є не розкішшю, а суворою необхідністю [11].

У відповідь на ці виклики необхідно розробити новий підхід до Secure Boot, який відповідатиме можливостям типових мікроконтролерів. Такий підхід повинен базуватись на використанні легковагових криптографічних примітивів (зокрема, хеш-функцій та цифрових підписів із низькими обчислювальними витратами), оптимізації структури завантаження, спрощеному зберіганні еталонних значень, і при цьому забезпечувати достатній рівень криптографічної стійкості. Крім того, важливо враховувати сценарії масштабування системи, підтримку віддалених оновлень (OTA), можливість автентифікації кількох образів прошивок та мінімізацію затримок запуску.

2. Аналіз попередніх досліджень

Захист вбудованого програмного забезпечення у пристроях з обмеженими обчислювальними ресурсами є актуальним напрямом досліджень у контексті забезпечення довіри до пристрою ще на етапі його завантаження [12, 13, 14]. Питання довіри до прошивки, яка керує взаємодією з фізичним середовищем, стає критичним не лише у побутових рішеннях, а й у промислових, медичних та військових застосуваннях. Порушення цілісності початкового коду здатне спричинити як несанкціоноване зчитування або модифікацію даних, так і фізичне пошкодження об'єкта керування. У зв'язку з цим концепція Secure Boot, яка полягає в перевірці автентичності та цілісності програмного коду перед передачею управління на наступний рівень завантаження, є важливим елементом підвищення надійності обчислювальних систем [15].

У типовій архітектурі Secure Boot цей процес реалізується через криптографічну перевірку цифрового підпису або хеш-функції, створеної на основі відкритого ключа виробника або адміністратора системи [16]. Протягом останнього десятиліття активні дослідження були спрямовані на побудову надійного довіреного ланцюга завантаження (chain of trust), в якому кожен етап перевіряє наступний, з використанням алгоритмів хешування SHA-2/SHA-3, підпису RSA, ECDSA, EdDSA тощо.

У повнофункціональних комп’ютерних системах ці механізми реалізуються на рівні UEFI, Trusted Platform Module (TPM) або з використанням апаратних безпечних зон, вбудованих у сучасні мікропроцесори [17]. Така інтеграція дозволяє забезпечити високий рівень контролю та гнучкості в управлінні довірою до програмного забезпечення, забезпечуючи формування надійного кореня довіри [18]. Однак у випадку мікроконтролерів класу STM32, ESP8266, ESP32, RISC-V або AVR, класичні рішення є надмірно ресурсомісткими. Бракує як достатнього обсягу оперативної пам’яті, так і підтримки асиметричної криптографії або спеціалізованих апаратних прискорювачів. Крім того, пристрої часто функціонують в енергозалежному або автономному режимі, що виключає застосування алгоритмів із високим енергоспоживанням.

Серед актуальних підходів до адаптації механізму Secure Boot у таких умовах виокремлюють використання легковагових криптографічних примітивів. Особливої уваги набули компактні хеш-функції, зокрема BLAKE2s, SPONGENT, PHOTON, QUARK, які можуть бути реалізовані навіть на 8-бітних мікроконтролерах [19, 20, 21]. Такі алгоритми мають обмежений розмір коду, не вимагають великих обсягів пам'яті, а також демонструють прийнятний рівень криптостійкості для задач початкової перевірки цілісності.

Ще один напрям досліджень – використання полегшених цифрових підписів на основі кривих з короткими ключами. Популярності набули реалізації Ed25519 та варіанти ECDSA з ключами 160–192 біт, які забезпечують компроміс між стійкістю і продуктивністю [22]. З метою додаткового спрощення, у дослідженнях [23, 24, 25] пропонується відмова від підпису на користь одностороннього хешування, що значно знижує обчислювальне навантаження. Водночас, одностороннє хешування не дозволяє підтвердити авторство коду, а лише перевіряє його незмінність. Такий підхід доцільний для систем із фіксованими прошивками, які не передбачають віддалених оновлень.

Деякі дослідження, зокрема [26, 27], пропонують часткове завантаження (partial or staged booting), коли автентифікації підлягають лише критично важливі сегменти коду, а решта – опціонально. Це дозволяє зменшити час перевірки та обсяг обчислень на етапі запуску. Водночас виникає питання, які саме компоненти вважати критичними та як забезпечити перевірку залежностей між ними.

Паралельно досліджується застосування механізмів Secure Element або Trusted Execution Environment (TEE), що дозволяють винести операції перевірки у спеціалізовану захищену область [28]. Проте такі рішення потребують спеціалізованих мікросхем, збільшують вартість пристрою та його енергоспоживання. Тому вони не є придатними для широкого використання у пристроях з наднизькою собівартістю або в сенсорних мережах, де головним є баланс між безпекою, ціною та тривалістю автономної роботи.

Незмінно актуальною залишається проблема вибору такого криптографічного ядра, яке з одного боку забезпечить необхідний рівень захищеності, а з іншого – не перевищить критичних порогів енергоспоживання та затримки запуску [29]. Це особливо важливо для медичних пристроїв (де затримка запуску може впливати на життєві показники) та військових сенсорних платформ (де енергонезалежність є критичною вимогою).

Попри активні дослідження, нині бракує уніфікованих, гнучких архітектур Secure Boot, які б дозволили адаптивно інтегрувати полегшені криптографічні алгоритми у типові IoT-платформи без жертвування продуктивністю або масштабованістю. Потреба у легких, оптимізованих, але безпечних механізмах безпечного старту залишається відкритим викликом у сфері захисту вбудованого ПЗ у мікроконтролерних системах. Саме тому розробка

ресурсоощадного, гнучкого, масштабованого Secure Boot з використанням легковагових примітивів та модульної верифікації є актуальним напрямом наукових досліджень і практичної реалізації в умовах сучасного IoT.

3. Невирішені питання у сфері Secure Boot

Аналіз засвідчує той факт, що, незважаючи на значний науковий прогрес у сфері забезпечення безпечного завантаження вбудованого програмного забезпечення для IoT-пристроїв, низка важливих аспектів залишається недостатньо дослідженою або має фрагментарні рішення, не придатні до практичного впровадження у реальних системах із жорсткими ресурсними обмеженнями.

Передусім, більшість наявних підходів зосереджені на забезпеченні базової цілісності прошивки, і лише одиничні рішення враховують обмеження на енергоспоживання, обсяг коду та час затримки запуску. Застосування навіть легковагових криптографічних примітивів потребує ретельного балансування між рівнем захисту та реальними можливостями мікроконтролерів. У практиці ж налагодження систем захищеного старту часто стикається з труднощами інтеграції: криптографічні бібліотеки вимагають додаткових ресурсів, а налаштування довірених ключів у пристроях без EEPROM або батареєзалежної пам'яті є технічно складним завданням [30, 31].

По-друге, існує проблема відсутності уніфікованого підходу до організації зберігання еталонних хешів або відкритих ключів у нестійкому середовищі. У багатьох випадках реалізації Secure Boot потребують запису «trusted hash» у флеш-пам'ять, однак такі рішення не завжди захищені від фізичного доступу, перепрошивання або зчитування через JTAG/UART-інтерфейси [32]. Пропозиції щодо використання прихованих сегментів пам'яті, захисту відчитування або шифрування області з еталонними значеннями лише частково вирішують проблему і не охоплюють повного життєвого циклу пристрою – від початкового прошивання до оновлення або ротації ключів [33].

Ще одним відкритим питанням є підтримка сценаріїв оновлення прошивки (OTA) із збереженням довіри. У більшості реалізацій Secure Boot система перевіряє лише першу версію прошивки, без врахування подальших оновлень. Це створює вразливість, коли автентифікований початковий код згодом завантажує модифіковану версію без додаткової перевірки. У той час як промислові рішення передбачають ланцюгову перевірку, у контексті малопотужних IoT-платформ така перевірка поки що не має ефективної реалізації з мінімальним накладом.

Крім того, бракує методик багаторівневої верифікації коду, коли компоненти прошивки (bootloader, ядро, бібліотеки, драйвери) перевіряються незалежно або вибірково залежно від критичності. Це могло б забезпечити більшу гнучкість, зменшити тривалість запуску й оптимізувати ресурсоспоживання. Проте розподіл довіри всередині вбудованої прошивки потребує її спеціальної структури, засобів маркування сегментів і методів

незалежної верифікації. Наразі ці питання не мають уніфікованих рішень для платформ з обмеженнями.

Не менш важливою є проблема адаптації запропонованих схем до широкого спектру апаратних конфігурацій – від 8-бітних мікроконтролерів до 32-бітних платформ з FreeRTOS [34]. Уніфіковані легковагові реалізації часто втрачають ефективність при переході на іншу платформу через відмінності у доступних криптографічних бібліотеках, способах доступу до пам'яті та відсутності спільного стандарту для формування прошивки.

Відсутність публічних моделей оцінки ефективності механізмів Secure Boot в умовах обмежених ресурсів (з урахуванням часу, енергоспоживання, обсягу пам'яті та рівня стійкості до атак) також ускладнює порівняння рішень і їх впровадження в критичні галузі. Оскільки не сформовано типові профілі загроз для вбудованих платформ без апаратної підтримки безпеки, то відсутні й критерії для обґрунтованого вибору мінімальних, але достатніх контрзаходів.

Таким чином, актуальним є формування нового, ресурсоощадного підходу до організації механізму Secure Boot, який базується на таких принципах:

- використання полегшених, але криптографічно стійких примітивів;
- мінімізація затримок та енергоспоживання;
- адаптивна архітектура з фазовою або модульною перевіркою коду;
- підтримка оновлень та ротації ключів;
- універсальність реалізації на платформах без апаратної криптографії.

4. Мета статті

Вирішення проблеми безпечного завантаження у вбудованих IoT-пристроях потребує ресурсоощадного підходу, який базується на сучасних, адаптованих до обмежених обчислювальних середовищ рішеннях.

Метою цього дослідження є розробка та експериментальне обґрунтування ефективного полегшеного механізму Secure Boot для захисту вбудованого програмного забезпечення IoT-пристроїв від несанкціонованої модифікації та запуску стороннього коду.

5. Технічні вимоги до полегшеного Secure Boot

Функціонування механізму Secure Boot у пристроях із низьким рівнем апаратних ресурсів потребує переосмислення стандартних вимог до безпечного завантаження та адаптації їх до специфіки вбудованих систем. На відміну від повнофункціональних комп'ютерних платформ, IoT-пристрої часто не мають розвиненої інфраструктури захисту, зокрема енергонезалежної пам'яті великого обсягу, апаратного криптографічного модуля чи захищеного сховища ключів. У таких умовах розробка полегшеного Secure Boot потребує чіткого визначення технічних вимог, які зберігають базові властивості безпечного старту, але залишаються реалізовними в обмеженому середовищі.

При побудові таких систем важливо не лише зменшити обсяг коду перевірки, а й мінімізувати кількість операцій читання і запису у пам'ять,

адаптувати механізм до енергонезалежного циклу живлення, а також гарантувати відсутність критичних затримок під час запуску. Тому механізм полегшеного Secure Boot повинен поєднувати в собі простоту реалізації, криптографічну стійкість і технологічну універсальність, що дозволить інтегрувати його як у нові пристрої, так і у вже наявні IoT-рішення з відкритим завантажувачем.

Для кращого зрозуміння технічних обмежень, за яких має працювати легкий Secure Boot, важливим є аналізи обчислювальних характеристик поширених платформ мікроконтролерів. На рис. 1 представлено порівняльний огляд ROM, RAM, and CPU для трьох репрезентативних архітектур Інтернету речей.

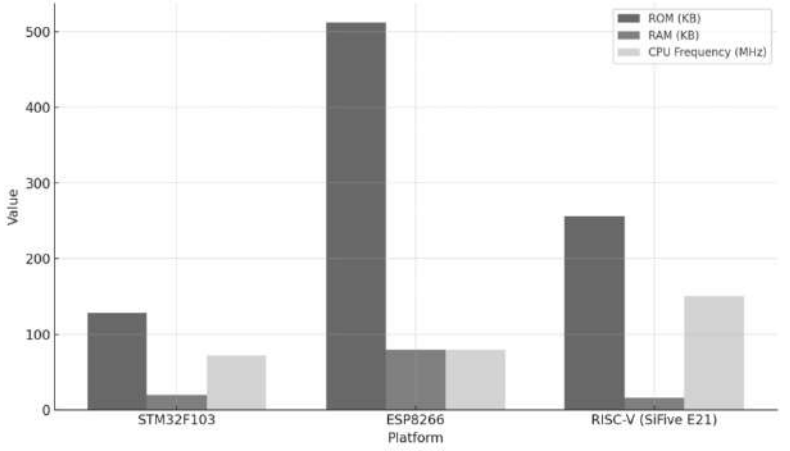


Рисунок 1. Ресурсні обмеження типових IoT-платформ [35]

Як видно з рис. 1, навіть найпотужніші мікроконтролери свого класу, такі як ESP8266, працюють з надто обмеженою оперативною пам'яттю та обчислювальною потужністю, і жодна з перелічених платформ не забезпечує вбудованих апаратних криптографічних механізмів. Ці обмеження вимагають використання легких криптографічних примітивів та мінімальної логіки перевірки в процесі безпечного завантаження.

Передусім, критичною є вимога до забезпечення перевірки цілісності вбудованого програмного забезпечення ще до його запуску. Вона передбачає верифікацію контрольної суми або хешу основної прошивки на основі заздалегідь визначеного еталонного значення. Для цього необхідно забезпечити використання легковагової криптографічної хеш-функції, яка демонструє достатню стійкість до колізій та підробок, але має низьке енергоспоживання та невеликий розмір реалізації. Зокрема, функції

SPONGENT, PHOTON або BLAKE2s у спрощеній формі можуть бути використані у відповідних архітектурах мікроконтролерів.

Зважаючи на специфіку середовища, до хеш-функції також висуваються вимоги щодо мінімального розміру коду (code size), низької затримки обчислення на слабких ядрах (наприклад, ARM Cortex-M3 або Tensilica L106), а також стійкості до сторонніх впливів, таких як помилки живлення або відмови флеш-пам'яті. Крім того, реалізація алгоритму має бути доступною для вбудування в проєкт без залучення складних зовнішніх бібліотек, які можуть перевищити допустимий обсяг прошивки. Тому на практиці ключовим критерієм вибору стає не лише математична надійність, а й придатність для оптимізації, наявність відкритого коду та успішне проходження тестів у типових IoT-середовищах.

Для вибору хеш-функції в умовах обмежених ресурсів необхідно враховувати не лише криптографічну стійкість, але й технічні характеристики реалізації. У табл. 1 наведено порівняння полегшених хеш-функцій, що можуть бути використані у механізмі Secure Boot на мікроконтролерах STM32 та ESP8266.

Таблиця 3

Порівняння полегшених хеш-функцій для застосування в Secure Boot

Хеш-функція	Вимоги до пам'яті (bytes/gates)	Тривалість обчислень	Стійкість до колізій	Енергоспоживання	Підтримка на STM32 / ESP8266
SPONGENT	2260 gates	низька	помірна	дуже низьке	так / частково
PHOTON	2177 gates	середня	висока	низьке	так / експериментально
BLAKE2s	≈ 1300 bytes	висока	дуже висока	середнє	так / оптимізовано

Як видно з табл. 1, SPONGENT має найменше енергоспоживання, однак поступається PHOTON і BLAKE2s у стійкості до колізій. Водночас BLAKE2s забезпечує найвищу криптографічну надійність, але вимагає більше обчислювальних ресурсів.

Другою важливою вимогою є захист від модифікації завантажувача, який сам виконує перевірку основної прошивки. Цей компонент повинен зберігатися у захищеній області пам'яті, за можливості – в енергонезалежній або тільки для читання. У разі відсутності апаратного поділу пам'яті важливо забезпечити принаймні уніфіковану перевірку ланцюга довіри, яка базується на криптографічному зв'язку між завантажувачем і основною прошивкою.

Ще одним ключовим параметром є обсяг доступної пам'яті для реалізації Secure Boot, зокрема кодової та оперативної. У багатьох поширених IoT-

платформах розмір флеш-пам'яті може бути обмежений до 128 або 256 КБ, а обсяг оперативної пам'яті – лише кілька кілобайтів. Тому будь-який алгоритм або структура перевірки має бути реалізована компактно, без значного накладного коду, він не повинен впливати на основну функціональність пристрою. Також необхідно враховувати час виконання перевірки та затримку запуску основної програми. Полегшений Secure Boot повинен забезпечувати мінімальний час початкової перевірки, оскільки у багатьох сенсорних пристроях, особливо в реальному часі, навіть незначна затримка може бути критичною. Баланс між рівнем безпеки та швидкістю запуску має бути визначений експериментально, з урахуванням допустимих часових меж для конкретного застосування. Крім цього, передбачається наявність джерела достовірного еталонного хешу або цифрового підпису, який має бути захищений від підміни. У випадку відсутності апаратного сховища ключів, як у Trusted Platform Module або Secure Element, доцільним є збереження еталонного значення в зашифрованому вигляді або вбудованим безпосередньо в початковий завантажувач.

У практиці проектування захищених IoT-систем одним із підходів є формування хешу прошивки на етапі компіляції з подальшим кодуванням цього значення у внутрішній сегмент пам'яті завантажувача, недоступний для зміни. Таке рішення зменшує потребу в додатковій пам'яті для зберігання ключів і дозволяє реалізувати початкову перевірку без звернення до зовнішніх джерел. Водночас безпечне зберігання ключових елементів у Boot ROM або у hardcoded вигляді потребує додаткового захисту від зчитування, наприклад, шляхом відключення інтерфейсів відладки або використання апаратної фіксації конфігурацій.

На рис. 2 представлено розширену архітектурну модель механізму Secure Boot у мікроконтролерах із обмеженими ресурсами. Ця модель включає не лише базове порівняння хешів, але й безпечне зберігання, генерацію випадкових чисел та посилання на відкритий ключ, що відображає практичні реалізації в реальних пристроях IoT.

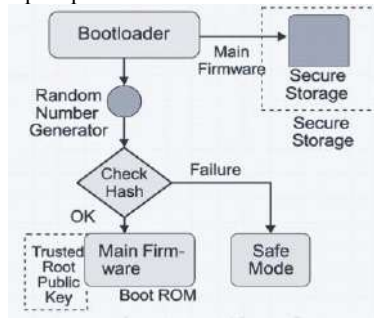


Рисунок 2. Мінімальні компоненти Secure Boot для мікроконтролера

Включення до структури моделі безпечного сховища хешів та довірених ключів дозволяє перевіряти автентичність без використання важких криптографічних модулів, тоді як резервний шлях забезпечує стійкість у разі втручання або невідповідності хешів.

Додатковою перевагою запропонованої архітектури є наявність аварійного режиму (Safe Mode), який дозволяє уникнути повної втрати працездатності пристрою у випадку виявлення некоректного коду. Це важливо для вбудованих систем, що виконують критичні функції, адже збереження мінімального рівня керованості навіть у разі порушення цілісності прошивки підвищує загальну стійкість системи. Саме таке багаторівневе проектування дозволяє досягти балансу між безпекою і витратами ресурсів, що є ключовим для low-power IoT-рішень.

Полегшений Secure Boot має задовольняти сукупність вимог, які забезпечують базову криптографічну перевірку автентичності програмного забезпечення, зберігаючи працездатність системи у межах обмежених обчислювальних, енергетичних та часових ресурсів.

6. Запропонована архітектура полегшеного механізму Secure Boot

У запропонованому підході Secure Boot реалізується як полегшена багатофазна перевірка цілісності та автентичності вбудованого програмного забезпечення, орієнтована на IoT-пристрої з обмеженими обчислювальними ресурсами. Архітектура враховує вимоги до мінімального обсягу пам’яті, обмеженої обчислювальної потужності та швидкодії системи, забезпечуючи при цьому базовий рівень безпеки без потреби в апаратних криптографічних модулях.

6.1. Загальна схема роботи

Механізм Secure Boot у запропонованій архітектурі реалізується як послідовність етапів перевірки цілісності основної прошивки перед її запуском. Його завдання – гарантувати, що на виконання буде передано лише автентичне та незмінене програмне забезпечення. Відразу після подачі живлення або перезавантаження пристрою, система ініціалізує Bootloader, який є єдиною точкою входу в систему до виконання основної програми.

Особливістю реалізації є те, що перевірка виконується в межах мінімально можливої кількості кроків, що знижує загальну затримку старту системи до прийнятного рівня навіть у пристроях з обмеженим тактовим генератором або частотою до 80 МГц. Механізм не передбачає складної структури переходів між фазами, а функції контролю винесені у початкову ділянку коду, що дозволяє жорстко контролювати потік виконання. У випадках, коли пристрій використовує FreeRTOS або подібне середовище, верифікація завершується ще до запуску планувальника. Тому забезпечується чітка логічна межа між захищеним запуском і звичайним виконанням коду.

Bootloader звертається до захищеної області пам’яті, де зберігається еталонний хеш основної прошивки. Паралельно здійснюється обчислення хешу актуального вмісту прошивки з використанням полегшеної хеш-функції.

Далі система виконує порівняння еталонного та розрахованого хешів. У разі збігу контрольна передача переходить до основного коду прошивки. У протилежному випадку пристрій переводиться у режим блокування (Safe Mode) або переходить у резервний сценарій.

На рис. 3 наведено загальну логіку Secure Boot із позначенням ключових вузлів і потоків обробки. Після завершення етапу верифікації система переходить до виконання основного функціонального коду.

Як видно з рис. 3, запропонована структура реалізує багатофазний підхід, у якому ключові перевірки виконуються ще до запуску основного коду. Завдяки використанню лише легковагових криптографічних примітивів і перевірці лише критичних ділянок пам'яті (таких як таблиця переривань, блоки ініціалізації або функції обробки даних), система мінімізує обчислювальні навантаження. Додаткові компоненти, як-от генератор випадкових чисел, відкритий ключ і захищене сховище, активуються лише за необхідності й не задіяні в основному циклі перевірки, що знижує споживання енергії. Такий підхід дозволяє гнучко масштабувати архітектуру відповідно до можливостей конкретної апаратної платформи, зберігаючи при цьому базову перевірку автентичності та цілісності. Завдяки цьому досягається ефективний баланс між швидкодією запуску, енергетичною доцільністю та відповідністю обмеженим ресурсам типових IoT-пристроїв.

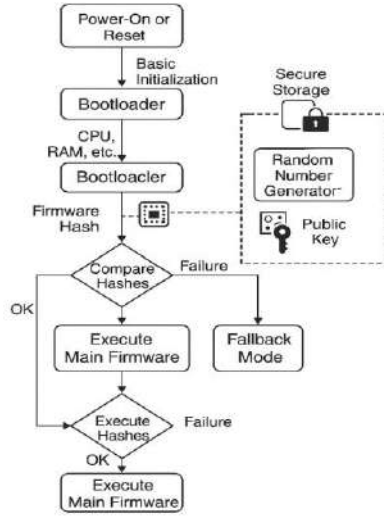


Рисунок 3. Загальний робочий процес запропонованого легкого механізму Secure Boot

6.2. Використані криптографічні елементи

З огляду на обмежену обчислювальну потужність цільових пристроїв, в якості основного криптографічного елемента обрано хеш-функцію SPONGENT-128/128/8, яка забезпечує баланс між стійкістю до атак і компактністю реалізації. Ця функція побудована на основі sponge-конструкції, що складається з раундів перестановок та побітових операцій, і дозволяє формувати хеш-підпис зі змінною довжиною на основі обмежених вхідних блоків. Характерною особливістю SPONGENT є її здатність ефективно працювати навіть на мікроконтролерах із частотою менше 100 МГц, споживаючи мінімум енергії (менше 10 μJ на операцію). Алгоритм хешування можна представлений, як (1).

$$Z = Sponge[f, pad, r](M), \quad (1)$$

де f – функція внутрішнього перестановки (у SPONGENT – побудована на основі π-раундів), pad – алгоритм доповнення до потрібної довжини, $r=8$ – кількість бітів, які обробляються за раунд, M – вхідне повідомлення (вміст прошивки або її критичні ділянки), Z – вихідний хеш.

Послідовність обробки виглядає так:

1. Ініціалізація внутрішнього стану: $S_0 = IV$.
2. Раунди поглинання: $S_{i+1} = f(S_i \oplus m_i)$ для кожного блоку m_i.
3. Формування результату: перші n бітів з фінального стану $Z = first_n(S_k)$.

Цей підхід дозволяє отримувати хеш зі змінною довжиною без потреби у використанні складних або енергозатратних операцій. У критичних пристроях реального часу SPONGENT дає змогу забезпечити перевірку цілісності з мінімальною затримкою старту системи.

У тих випадках, коли архітектура пристрою допускає використання більш складних алгоритмів, може застосовуватись BLAKE2s, який має вищу криптографічну надійність та перевірену реалізацію в низці безпекових бібліотек. Також може бути використано PHOTON, який демонструє стійкість до атак диференціального та лінійного криптоаналізу й має компактну реалізацію для платформ із пам'яттю до 8 КБ.

У базовому варіанті архітектури механізм Secure Boot не використовує цифровий підпис, що дозволяє зменшити обсяг коду, кількість операцій із великими числами та час затримки запуску. Однак у випадках, де важлива автентифікація джерела прошивки, передбачено розширений варіант з використанням полегшеного цифрового підпису на основі ECDSA з 160-бітовими кривими (наприклад, secp160r1). Такий підпис потребує мінімум двох еліптичних множень під час перевірки, однак здатен гарантувати достовірність навіть у разі доступу з боку атакуючого до зовнішньої пам'яті.

Процедура перевірки підпису у цьому випадку виконується таким чином:

Обчислюється хеш повідомлення (прошивки): $e = HASH(M)$.

Визначаються допоміжні значення:

$\varpi = s^{-1} \bmod n,\ u_1 = e \cdot \varpi \bmod n, u_2 = r \cdot \varpi \bmod n.$

Обчислюється точка на кривій: $P = u_1 G + u_2 QP$, де G – базова точка, Q – відкритий ключ, r, s_r – параметри підпису.

Перевірка автентичності: $r \equiv x_p \bmod n.$

Тут x_p – x-координата точки P, що отримана в результаті обчислень. Усі еліптичні операції реалізуються за допомогою арифметики, а відкрита частина ключа може бути збережена в ROM, вбудована у завантажувач або зашифровано витягнута з зовнішнього джерела.

Автентичність хешу підтверджується відкритим ключем, який може бути збережений кількома способами:

- у енергонезалежній ROM-пам'яті (захищений варіант);
- у початковому завантажувачі (жорстко зашитий ключ);
- завантажений з зашифрованого джерела, за умови наявності модуля дешифрування.

Цей підхід дозволяє забезпечити односторонній довірчий ланцюг навіть за відсутності повноцінного TPM або Secure Element. Обчислення цифрового підпису або його перевірка активуються лише у фазі оновлення або перевірки еталонного хешу, не зачіпаючи основний цикл запуску, що дає змогу уникати енергетичних піків у критичних режимах роботи.

6.3. Збереження та перевірка коду

Еталонний хеш основної прошивки зберігається у захищеній області флеш-пам'яті, до якої Bootloader має доступ лише в режимі читання. У деяких реалізаціях, де можливе апаратне розділення пам'яті, використовується окремий ROM-сегмент, що забезпечує незмінність еталонного значення.

Особливістю реалізації є перевірка не всієї прошивки, а лише її найбільш критичних ділянок: початкового коду ініціалізації, таблиці переривань, функцій обробки мережевого трафіку. Це зменшує обсяг обчислень і дозволяє виконувати перевірку за прийнятний час. Стратегія запуску базується на логіці fail-stop: у разі виявлення невідповідності виконання основної програми не отримує дозволу, а пристрій переходить у заблокований або резервний режим.

Послідовність дій з перевірки автентичності та цілісності програмного забезпечення включає такі етапи:

1. Після подачі живлення або перезапуску пристрою активується Bootloader, який здійснює базову ініціалізацію системи та перевірку доступу до пам'яті.

2. Визначаються межі критичних ділянок прошивки, що підлягають перевірці. Це можуть бути області з кодом початкової ініціалізації, таблиця векторів переривань або функції, що обробляють мережевий трафік.

3. Bootloader ініціює хешування обраних сегментів з використанням полегшеної хеш-функції (SPONGENT, PHOTON або BLAKE2s), яка адаптована до умов низької продуктивності та обмеженого енергоспоживання.

4. Обчислений хеш порівнюється з еталонним значенням, збереженим у захищеній області пам'яті. Порівняння виконується без проміжних копій у змінній пам'яті, що унеможливлює маніпуляції під час перевірки.

5. У разі успішного збігу Bootloader передає управління основній програмі, яка починає своє виконання згідно із завантаженою логікою.

6. Якщо виявлено розбіжність між обчисленим і еталонним хешем, система негайно зупиняє завантаження та переходить у безпечний стан Safe Mode, з блокуванням інтерфейсів або ініціацією процесу оновлення з довіреного джерела.

7. Для підвищення надійності у деяких реалізаціях допускається зберігання декількох допустимих хешів, що дозволяє підтримувати кілька версій прошивки та реалізовувати механізм резервного запуску.

Запропонована послідовність перевірки дозволяє дотриматися принципів безпеки з урахуванням технічних обмежень платформи та зберегти продуктивність системи навіть в умовах критично низьких ресурсів.

Розроблений механізм забезпечує не лише перевірку цілісності ключових сегментів програмного коду, але й створює довірене середовище для подальшої роботи пристрою. Адаптація класичної моделі Secure Boot до умов обмежених ресурсів дозволяє забезпечити базові гарантії безпеки без суттєвого зростання часу завантаження або обсягу коду. Завдяки модульності та підтримці кількох хешів у пам'яті, механізм легко масштабувати та інтегрувати у гетерогенні IoT-системи, де критично важливими є автономність, енергоефективність та надійність запуску.

7. Експериментальна частина та оцінка ефективності

Для перевірки працездатності запропонованої архітектури Secure Boot було проведено її експериментальну реалізацію на двох типових платформах IoT-класу: STM32F103C8T6 (мікроконтролер на базі ARM Cortex-M3 з тактовою частотою 72 МГц) та ESP8266 NodeMCU (процесор Tensilica L106, 80 МГц). Обидва мікроконтролери широко використовуються в проєктах з обмеженим енергоспоживанням і мають низький обсяг оперативної пам'яті, що дозволяє адекватно оцінити ефективність механізму в умовах обмежених ресурсів.

Було реалізовано прототип Secure Boot, що включав Bootloader обсягом 5,1 КБ, модуль обчислення хешу SPONGENT-128/128/8, захищене збереження еталонного хешу, а також верифікацію перед виконанням основної програми. В якості тестової прошивки використовувалась програма обсягом 32 КБ, у якій навмисно модифікувались критичні ділянки, такі як таблиця переривань та функції обробки даних.

Програмування здійснювалось мовою C з використанням компілятора GCC (для STM32) та Arduino Core (для ESP8266). Параметри часу перевірки, енергоспоживання та обсяг коду оцінювались з використанням STM32CubeMonitor, INA219 (для вимірювання миттєвого струму) та осцилографа Tektronix. У результаті експериментів встановлено, що для

STM32 повний цикл перевірки займає 21,7 мс, для ESP8266 – 29,1 мс. Обчислення хешу SPONGENT тривало 18,2 мс і 25,4 мс відповідно, при цьому енергоспоживання однієї перевірки не перевищувало 10–11 мікроджоулів. Обсяг коду Secure Boot після компіляції склав 6,4 КБ для STM32 та 7,1 КБ для ESP8266.

Під час реалізації особливу увагу було приділено оптимізації обчислювального ядра хеш-функції та мінімізації звернень до пам'яті. Виявлено, що найбільший вплив на час перевірки має кількість обраних для верифікації сегментів, а також глибина циклів обробки даних у SPONGENT. У випадку STM32 було досягнуто стабільної роботи навіть при роботі з малими блоками даних по 64 байти, тоді як на ESP8266 виявлено затримки при роботі з блоками понад 128 байт. Це свідчить про критичну роль оптимального розміру оброблюваного сегмента, що має враховуватись при адаптації механізму до конкретної платформи.

Для оцінки ефективності використовувалися три основні метрики: тривалість перевірки, енергоспоживання під час старту, а також розмір коду завантажувача. Тестові сценарії передбачали як перевірку цілісної прошивки, так і симуляцію модифікованого коду. Симуляція атаки шляхом модифікації коду показала, що механізм коректно виявляє зміну навіть одного байта в перевірюваній області, що підтверджує високу чутливість і надійність алгоритму. У всіх тестах спрацювання fail-safe механізму спостерігалося негайно – з переходом до Safe Mode менш ніж за 1 мс після невдалої перевірки. Таким чином, запропонований механізм забезпечує високу швидкість реакції на загрози без суттєвого навантаження на апаратну частину пристрою. Результати експерименту наведено у табл. 2.

Таблиця 2

Експериментальна оцінка запропонованої реалізації Secure Boot

Метрика	STM32F103C8T6	ESP8266 NodeMCU
Розмір коду завантажувача	6.4 KB	7.1 KB
Тривалість хешування (SPONGENT)	18.2 ms	25.4 ms
Тривалість перевірки	21.7 ms	29.1 ms
Енергоспоживання на перевірку	9.8 µJ	11.3 µJ
Витрати пам'яті (RAM)	< 512 bytes	< 768 bytes
Виявлення втручання	100%	100%
Реакція захисту основної прошивки	Safe Mode або відкат	Safe Mode або відкат

Як показано в таблиці, обидві платформи демонструють високу точність у виявленні зміненої прошивки без помітної затримки запуску системи. Найнижчі значення енергоспоживання та часу перевірки досягнуто завдяки

використанню легковагових криптографічних примітивів та обмеженню перевірки лише до критичних сегментів пам'яті. Це дозволяє ефективно інтегрувати захист в системи з чутливістю до затримки або автономним живленням. Отримані результати підтверджують перспективність запропонованої архітектури як гнучкого рішення для безпечного старту у пристроях з обмеженими ресурсами.

У проведеній серії експериментів система успішно ідентифікувала підміну прошивки: у випадках зміни контрольних байтів пристрій переходив у безпечний режим без запуску основної програми. Це свідчить про здатність механізму протидіяти несанкціонованим модифікаціям навіть у разі фізичного доступу до пам'яті пристрою. Порівняно з традиційними підходами, які потребують цифрового підпису та еліптичної криптографії, запропонований варіант з полегшеним хешуванням показав у 3–4 рази нижчу тривалість запуску та на порядок менше енергоспоживання.

Однак у базовій реалізації не враховано захист еталонного хешу від прямого зчитування, що може бути критичним у разі фізичного злому. Подальші дослідження планується спрямувати на інтеграцію методів обфускації, використання Secure Element, а також впровадження довіреного віддаленого оновлення (OTA) з криптографічною перевіркою на сервері. Отримані результати підтверджують доцільність впровадження механізму в практичні IoT-рішення, де критичним є поєднання безпеки та енергоефективності.

8. Обговорення

Результати експериментів підтвердили, що запропонований механізм полегшеного Secure Boot може бути ефективно реалізований на мікроконтролерах із обмеженими ресурсами, зберігаючи баланс між продуктивністю та базовим рівнем безпеки. Використання полегшеної хеш-функції та обмеження перевірки до критичних сегментів прошивки дозволило досягти прийнятної тривалості запуску та мінімального енергоспоживання, що є надзвичайно важливим для автономних IoT-пристроїв. Успішне виявлення модифікованих прошивок у всіх тестових сценаріях засвідчує здатність запропонованої архітектури протидіяти базовим загрозам без потреби в складних апаратних модулях.

Разом із тим, певні обмеження все ще залишаються актуальними. Зокрема, зберігання еталонного хешу у відкритому вигляді навіть у захищеній флеш-пам'яті залишає простір для потенційних атак у разі фізичного доступу до пристрою. Також у поточній реалізації відсутній захищений механізм оновлення прошивки, що ускладнює використання рішення в динамічному середовищі з частими оновленнями. Не враховано також можливість інтеграції із зовнішніми джерелами довіри або віддаленими серверами перевірки, що є необхідним для побудови повноцінного ланцюга довіри.

Перспективними напрямами подальших досліджень є інтеграція механізмів шифрування або обфускації еталонного хешу, впровадження розширеного варіанту із цифровим підписом, а також використання зовнішніх апаратних компонентів, таких як Secure Element або TPM. Окрема увага має бути приділена оптимізації енергоспоживання в процесі оновлення та розробці архітектури, яка підтримує безпечне віддалене оновлення з верифікацією на сервері. Доцільним є також розширення підтримки новітніх криптографічних алгоритмів, орієнтованих на енергоефективність, та включення інструментів моніторингу поведінки пристрою під час завантаження для побудови адаптивної, самовідновлюваної системи захисту.

9. Висновки та перспективи

У статті запропоновано архітектуру полегшеного механізму Secure Boot, адаптовану до умов обмежених ресурсів IoT-пристроїв. Розроблений підхід ґрунтується на використанні легковагових криптографічних примітивів, зокрема хеш-функції SPONGENT-128/128/8, та передбачає перевірку лише критичних ділянок прошивки з метою мінімізації енергоспоживання та часу запуску. Запропоноване рішення не вимагає наявності апаратних криптомодулів і може бути реалізоване на популярних мікроконтролерах, таких як STM32 та ESP8266. Архітектура зберігає логіку fail-stop і дозволяє уникати запуску зміненого коду без необхідності у складному апаратному захисті. Водночас, її реалізація потребує мінімального обсягу пам’яті, що дозволяє використовувати її у вкрай ресурсно обмежених пристроях.

Експериментальна реалізація підтвердила працездатність і ефективність розробленого механізму: затримка перевірки становила менше 30 мс, а енергоспоживання залишалося в межах 10–11 мікроджоулів. Усі модифікації прошивки були успішно виявлені, що свідчить про доцільність застосування запропонованого підходу в системах, де критичними є одночасно безпека та енергоефективність. Прототип функціонував стабільно при багаторазових перезапусках, демонструючи низький ризик збоїв навіть за умов коливань напруги. Це особливо важливо для IoT-рішень, які працюють у нестабільному середовищі або на живленні від батарей.

Результати дослідження відкривають перспективу подальшого вдосконалення архітектури, зокрема впровадження підтримки цифрового підпису в розширеній конфігурації, обфускації еталонних хешів та інтеграції із захищеним модулем оновлення програмного забезпечення. Запропоноване рішення є гнучким і придатним для масштабування на інші класи пристроїв, що робить його потенційною основою для створення довірених IoT-середовищ.

Література

1. Rozlomii, I., Yarmilko, A., & Naumenko, S. (2025). Innovative resource-saving security strategies for IoT devices. *Journal of Edge Computing*, *4*(1), 35–56.

2. Yarmilko, A., Rozlomii, I., & Naumenko, S. (2024, May). Dependability of Embedded Systems in the Industrial Internet of Things: Information Security and Reliability of the Communication Cluster. In *International Scientific-Practical Conference" Information Technology for Education, Science and Technics"* (pp. 235-249). Cham: Springer Nature Switzerland.

3. Qasem, A., Shirani, P., Debbabi, M., Wang, L., Lebel, B., & Agba, B. L. (2021). Automatic vulnerability detection in embedded devices and firmware: Survey and layered taxonomies. *ACM Computing Surveys (CSUR)*, *54*(2), 1-42.

4. Wang, R., & Yan, Y. (2022, February). A survey of secure boot schemes for embedded devices. In *2022 24th International Conference on Advanced Communication Technology (ICACT)* (pp. 224-227). IEEE.

5. Cano-Quiveu, G., Ruiz-de-Clavijo-Vazquez, P., Bellido, M. J., Juan-Chico, J., & Viejo-Cortes, J. (2023). IRIS: An embedded secure boot for IoT devices. *Internet of Things*, *23*, 100874.

6. de Clavijo Vázquez, P. R., Bellido, M. J., Chico, J. J., Cortes, J. V., & Vallecillo, J. B. (2024). Hardware Secure Boot. A Review Of" IRIS: An Embedded Secure Boot for IoT devices". *IX Jornadas Nacionales de Investigación En Ciberseguridad*, 512-513.

7. Faure, E., Rozlomii, I., & Naumenko, S. (2025). Cryptographic load sharing method in critical infrastructure sensor networks. In *Proceedings of the Fourth International Conference on Cyber Hygiene & Conflict Management in Global Information Networks (CH&CMiGIN'25)* (pp. 5–15).

8. Dhrubo, A. F. H., Hasan, S., & Qayum, M. A. (2025). STM32-Based IoT Framework for Real-Time Environmental Monitoring and Wireless Node Synchronization. *arXiv preprint arXiv:2506.17295*.

9. Ahemd, A., & Badawy, W. (2023). Design of IoT Wireless Programmer for Microchip AVR Microcontrollers' OTA Firmware Updates. DOI: 10.21203/rs.3.rs-3504697/v1.

10. Vdovichenko, O., & Perepelitsyn, A. (2024). Analysis of product range of microcontrollers for creation of elements of home automation. *Aerospace Technic and Technology*, (5), 118-126.

11. Rozlomii, I., Yarmilko, A., Naumenko, S., & Mykhailovskyi, P. (2024, September). Hardware encryptors and cryptographic libraries for optimizing security in IoT. In *Proceedings of the 12th International Conference Information Control Systems & Technologies (ICST 2024), Odesa, Ukraine* (pp. 99-109).

12. Kornaros, G. (2022). Hardware-assisted machine learning in resource-constrained IoT environments for security: review and future prospective. *IEEE Access*, *10*, 58603-58622.

13. De Oliveira Nunes, I., Jakkamsetti, S., Kim, Y., & Tsudik, G. (2022, October). Casu: Compromise avoidance via secure update for low-end embedded systems. In *Proceedings of the 41st IEEE/ACM International Conference on Computer-Aided Design* (pp. 1-9).

14. Grisafi, M., Ammar, M., Roveri, M., & Crispo, B. (2022). {PISTIS}: Trusted computing architecture for low-end embedded systems. In *31st USENIX Security Symposium (USENIX Security 22)* (pp. 3843-3860).

15. Arunkumar, S. (2024, May). Implementation and Verification of Secure Boot in Embedded Systems. In *2024 International Conference on Advances in Modern Age Technologies for Health and Engineering Science (AMATHE)* (pp. 1-4). IEEE.

16. Jyothi, T., & Jain, S. (2023, May). Tpm based secure boot in embedded systems. In *2023 Third International Conference on Secure Cyber Computing and Communication (ICSCCC)* (pp. 786-790). IEEE.

17. Schmidt, S., Tausig, M., Koschuch, M., Sheng, B., Hudler, M., Simhandl, G., ... & Michael, M. K. (2025). Leveraging Trusted Platform Modules (TPM) for Cryptographic Anchoring and Remote Attestation of UEFI Capsule Updates in Secure Boot Environments.

18. Ling, Z., Yan, H., Shao, X., Luo, J., Xu, Y., Pearson, B., & Fu, X. (2021). Secure boot, trusted boot and remote attestation for ARM TrustZone-based IoT Nodes. *Journal of Systems Architecture*, *119*, 102240.

19. Tandel, P., & Nasriwala, J. (2024). Efficient authentication framework with Blake2s and a hash-based signature scheme for Industry 4.0 applications. *International Journal of Intelligent Engineering Informatics*, *12*(4), 410-432.

20. Hiremath, S. B., Heblikar, S., Javali, M. S., Tejas, M., & Iyer, N. C. (2024, August). Side Channel Analysis and Hardware Implementation of AES CBC Algorithm on Artix A7 FPGA Development Boards. In *International Conference on ICT for Sustainable Development* (pp. 111-120). Singapore: Springer Nature Singapore.

21. Al-Shatari, M., Hussin, F. A., Aziz, A. A., Eisa, T. A. E., & Tran, X. T. (2023). IoT Edge Device Security: An Efficient Lightweight Authenticated Encryption Scheme Based on LED and PHOTON. *Applied Sciences*, *13*(18), 10345.

22. Brendel, J., Cremers, C., Jackson, D., & Zhao, M. (2021, May). The provable security of ed25519: theory and practice. In *2021 IEEE Symposium on Security and Privacy (SP)* (pp. 1659-1676). IEEE.

23. Román, R., Arjona, R., & Baturone, I. (2023). A lightweight remote attestation using PUFs and hash-based signatures for low-end IoT devices. *Future Generation Computer Systems*, *148*, 425-435.

24. Songshen, H. A. N., Kaiyong, X. U., Zhiqiang, Z. H. U., Songhui, G. U. O., Haidong, L. I. U., & Zuohui, L. I. (2022). Hash-based signature for flexibility authentication of IoT devices. *Wuhan University Journal of Natural Sciences*, *27*(1), 1-10.

25. Chakraborty, A., & Biswas, A. (2025, February). A Comprehensive and Systematic Analysis of One-Way Hashing and Its Advancements. In *2025 3rd International Conference on Intelligent Systems, Advanced Computing and Communication (ISACC)* (pp. 1309-1316). IEEE.

26. Li, B., & Dong, W. (2021). Edge-centric programming for iot applications with automatic code partitioning. *IEEE Transactions on Computers*, *71*(10), 2408-2422.

27. Dong, Y., Duan, S., Lyu, F., Zhao, P., Zhang, Y., Ren, J., & Zhang, Y. (2024). NC-Load: On-Demand Program Loading and Running for Computing Sharing Among IoT Devices. *IEEE Internet of Things Journal*.

28. Muñoz, A., Ríos, R., Román, R., & López, J. (2023). A survey on the (in) security of trusted execution environments. *Computers & Security*, *129*, 103180.

29. Rozlomii, I., Naumenko, S., Mykhailovskyi, P., & Monarkh, V. (2024, October). Resource-Saving Cryptography for Microcontrollers in Biomedical Devices. In *2024 IEEE 5th KhPI Week on Advanced Technology (KhPIWeek)* (pp. 1-5). IEEE.

30. Bruno, G. (2023). *Design and Implementation of Trusted Channels in the Keystone Framework* (Doctoral dissertation, Politecnico di Torino). URL: https://webthesis.biblio.polito.it/29457/.

31. Rowland, M., & Karch, B. (2022). *A Review of Technologies that can Provide a 'Root of Trust' for Operational Technologies*. URL: https://www.osti.gov/servlets/purl/1861944.

32. Kumar, O. (2024). *Embedded Firmware Debugging and Telemetry* (Master thesis, TU Delft). URL: https://repository.tudelft.nl/record/uuid:f3b05137-f9aa-49c9-bee9-6797a742b14f.

33. Duan, X., Li, J., Pei, X., Ergesh, T., Wen, Z., & Chen, M. (2022, August). Implementation of CASPAR Firmware Interface on PYNQ-Z2. In *2022 IEEE 9th International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications (MAPE)* (pp. 53-57). IEEE.

34. He, C., Zhang, J., Cai, Y., Zi, C., & Jiang, X. (2021). An implementation of module management controller for MicroTCA data processing system. *Journal of Instrumentation*, *16*(03), T03005.

35. Thakor, V. A., Razzaque, M. A., & Khandaker, M. R. (2021). Lightweight cryptography algorithms for resource-constrained IoT devices: A review, comparison and research opportunities. *IEEE Access*, *9*, 28177-28193.

RESOURCE-EFFICIENT APPROACH TO SEQURE BOOT IN EMBEDDED INTERNET OF THINGS SYSTEMS

Ph.D. I. Rozlomii[1][0000-0001-5065-9004], **Ph.D. A. Yarmilko**[2][0000-0003-2062-2694], **S. Naumenko**[2][0000-0002-6337-1605]

[1]*Cherkasy State Technological University, Ukraine*
[2]*Bohdan Khmelnytsky National University of Cherkasy, Ukraine*
EMAIL: inna-roz@ukr.net, a-ja@ukr.net, naumenko.serhii1122@vu.cdu.edu.ua

Abstract. *This work presents a resource-efficient approach to the design of a Secure Boot mechanism specifically tailored for embedded Internet of Things (IoT) systems operating under constrained hardware resources. The proposed*

architecture addresses critical limitations such as restricted memory capacity, lack of hardware cryptographic support, and the necessity to minimize energy consumption. Classical secure boot models are largely unsuitable for these conditions; therefore, a lightweight Secure Boot variant has been developed using lightweight hash functions, in particular SPONGENT-128/128/8, which ensures a basic level of software authenticity and integrity at minimal computational cost. Implementation on STM32 and ESP8266 microcontrollers demonstrated high accuracy in detecting firmware modifications, energy efficiency (less than 11 μJ per verification), and a startup delay not exceeding 30 ms. The study discusses secure storage of the reference hash, the use of ECDSA-based digital signatures, and scaling options for different hardware configurations. The proposed approach is promising for integration into medical, industrial, military, and other applied IoT solutions where balancing security, energy efficiency, and performance is crucial. The conclusion outlines directions for future research, including obfuscation techniques, secure OTA updates, and support for emerging cryptographic primitives.

Keywords. *IoT; lightweight Secure Boot; embedded software; cryptographic hash; SPONGENT; energy efficiency*

UDC 004.8
DOI https://doi.org/10.36059/978-966-397-538-2-7

METHOD FOR ANALYZING THE UKRAINIAN LANGUAGE TEXTS SENTIMENT USING NATURAL LANGUAGE PROCESSING

O. Zalutska[0000-0003-1242-3548]
Khmelnytskyi National University, Ukraine
EMAIL: zalutsk.olha@gmail.com

Abstract. The paper focuses on intelligent sentiment analysis of text related to named entities. The proposed method combines a neural network-based natural language processing model, a lexical NLP library, and a Ukrainian sentiment dictionary. It provides results in the form of sentiment scores for named entities at the sentence and text levels, as well as an overall sentiment evaluation of the analyzed content. The relevance of the research is determined by the growing need for accurate sentiment analysis in the context of large-scale digital information flows. Identifying emotional attitudes toward specific persons, organisations, or events has essential applications in monitoring public opinion, brand perception, political discourse, and financial market analysis. The scientific novelty lies in developing and implementing a method that supports Ukrainian-language texts and evaluates sentiment across negativity, neutrality, positivity, and emotionality dimensions. The practical significance is creating a software system capable of semantic sentiment analysis of textual content, achieving higher effectiveness than

translation-based approaches. The developed method can analyze public opinion, social media reactions, market trends, and individual texts.

Keywords: *named entity recognition, emotional tone, emotional tone detection, Stanza, VADER*

Problem Statement

Intelligent sentiment analysis of textual information, particularly in the context of named entities, has become increasingly relevant today, where vast amounts of data are constantly exchanged and processed. In the era of digital technologies and information overload, quickly and accurately identifying the emotional context and subjective attitudes toward specific named entities – such as individuals, organizations, or events – has gained crucial importance [1]. Applying these methods to named entities has numerous practical implications, including monitoring brand perception in reviews and mentions, identifying public sentiment toward political figures, events, or products in social media, and analyzing the impact of news on market indicators and financial risks. Moreover, sentiment analysis at the entity level not only reveals the general emotional tone of a text but also distinguishes attitudes toward specific entities, which may vary within a single document [2]. From a natural language processing perspective, sentiment analysis related to named entities requires the design of advanced algorithms and neural models capable of correctly interpreting linguistic but also semantic, contextual, and culturally specific features of language [3]. The study introduces innovations in sentiment analysis, particularly by enhancing the method of intelligent sentiment evaluation relating to named entities. The proposed approach generates comprehensive output data based on a neural network NLP model, a lexical NLP library, and a Ukrainian sentiment dictionary. This includes identified named entities, sentiment scores for each entity at the sentence level, aggregated sentiment scores for the entire text, and overall sentiment assessments at both sentence and text levels [4].

Unlike existing methods, the developed approach is adapted explicitly to Ukrainian texts and enables sentiment evaluation for named entities within individual sentences and across the whole text. Furthermore, it provides sentiment analysis along four key dimensions: negativity, neutrality, positivity, and emotionality.

Analysis of Previous Studies

Existing methods and tools for intelligent analysis of the tone of textual information in relation to named entities. The review of methods and tools for intelligent sentiment analysis concerning named entities demonstrates that this task requires a combination of advanced natural language processing techniques and machine learning models. The process is generally divided into three main stages:

– named entity recognition, where algorithms identify and classify persons, organizations, locations, and other entities in text;

– sentiment determination, which analyzes the local context of each entity to detect its emotional tone;

– contextual analysis considers the document's broader semantic and thematic background to ensure a more accurate interpretation of sentiment [5].

To implement these stages, various approaches are employed, such as Bidirectional Encoder Representations from Transformers (BERT), Long Short-Term Memory (LSTM) networks, Conditional Random Fields (CRFs), and NLP libraries like spaCy and NLTK. These methods are among the most powerful and widely used entity detection and sentiment analysis tools. In addition, lexicon-based and hybrid approaches such as VADER, TextBlob, and Google Cloud Natural Language API are actively applied for sentiment evaluation. However, most of these solutions are primarily optimized for English, which limits their effectiveness for Ukrainian texts due to linguistic and cultural differences [6].

The analysis shows that while lexicon-based tools such as VADER or TextBlob can effectively detect sentiment in short and informal texts, they often require adaptation or developing specialized resources for Ukrainian. On the other hand, deep learning models such as BERT and LSTM provide higher accuracy and adaptability by capturing semantic and contextual dependencies. However, they demand large annotated datasets and significant computational resources [7].

The study highlights traditional and state-of-the-art approaches to entity-level sentiment analysis, outlining their strengths and limitations. It also emphasizes the growing relevance of neural network-based methods, which offer more profound and precise sentiment interpretation. In the modern information environment, the ability to automatically determine attitudes toward specific named entities is becoming increasingly important for applications such as monitoring public opinion, analyzing social media reactions, studying consumer behavior, and evaluating market trends [8].

Research on the analysis of scientific publications in the intellectual study of text tone in relation to named entities. The review focused on several key aspects of sentiment analysis methods: the type of algorithm, the use of supervised or unsupervised learning, the specific models applied, characteristics of datasets including size and collection methods, vectorization techniques, and evaluation metrics. Table 1 summarizes the findings from selected studies, highlighting the predominance of supervised learning methods, while unsupervised approaches were used much less frequently.

The comparative analysis of the reviewed studies highlights the predominance of supervised learning methods in sentiment analysis, as most authors relied on these approaches [8, 9, 10, 11, 12]. In contrast, unsupervised methods were applied only occasionally in a single study [10]. A wide range of algorithms has been explored, including classical machine learning models such as logistic regression (LR), support vector machines (SVM), random forest classifiers (RFC), and naïve Bayes, as well as deep learning models like CNNs, LSTMs, and advanced transformer-based architectures such as BERT, RoBERTa, XLNet, ALBERT, BART, and DistilBERT [8, 9, 10, 11, 12]. Simpler lexicons- and rule-based tools

like TextBlob, VADER, and Stanza were also tested, demonstrating their relevance for rapid sentiment detection in short texts [9].

Table 1

Results of research scientific papers

№ in the References	Methods	Training type	Metrics	Vectorisation
8	LR, SVM, RFC, NBC	Supervised	Accuracy, Precision, Recall, F1-score	TF-IDF
9	RoBERTa	Supervised	Accuracy, Precision, Recall, F1-score	-
10	TextBlob, VADER, Stanza	Both	-	-
11	SVM, XGBoost, RNN, RNN + Attention, CNN, Dense Network, BERT, XLNet, RoBERTa, ALBERT, BART, DistilBERT	Supervised	Accuracy, Precision, Recall, F1-score, MCC	BoW, TF-IDF, Harvard IV-4, LM, Word2Vec, GloVe, 23FastText, ELMo, Doc2Vec, Skip-Thought Vectors, InferSent
12	Ensemble of LR, RC, and Weightless Neural Network (WNN)	Supervised	F1-score	TF-IDF

Evaluation across studies was mainly based on accuracy, precision, recall, and F1-score, with some works also employing the Matthews correlation coefficient (MCC) for a more robust performance measure [11]. Different vectorization techniques were adopted, ranging from traditional bag-of-words (BoW) and TF-IDF to semantic embeddings like Word2Vec, GloVe, FastText, ELMo, InferSent, and Skip-Thought Vectors [10, 11]. Notably, [10] focused on unigram and bigram TF-IDF features, while [11] emphasized emotion recognition in dialogue systems, a domain increasingly relevant for conversational agents. Dialogue-related approaches were divided into party-dependent models (e.g., DialogueRNN) and party-independent ones (e.g., AGHMN), with additional frameworks like BiERU, GNTB, and TFE proposed to capture high-quality emotional features [9]. Ensemble methods, such as combining LR, RC, and weightless neural networks, were also investigated to enhance robustness [12].

These studies reveal a trend toward leveraging deep neural networks and transformer-based architectures for large-scale sentiment analysis, highlighting the complementary role of simpler models and lexicon-based tools. Integrating diverse

vectorization techniques and advanced architectures opens new possibilities for capturing subtle emotional nuances in text [13].

The analysis revealed various algorithms, including Logistic Regression, SVM, Random Forest, CNN, LSTM, RoBERTa, TextBlob, and VADER, with particular attention given to SVM due to its frequent application. Evaluation metrics most commonly employed were accuracy, precision, recall, and F1-score, while advanced studies also incorporated MCC. Vectorization methods varied from classical BoW and TF-IDF to modern embeddings such as Word2Vec, GloVe, FastText, ELMo, and BERT-based encodings [14].

Unresolved Issues

The studies illustrate that sentiment analysis of textual data is an intensively researched domain with diverse approaches. Research efforts are primarily directed toward improving the processing of massive text datasets through advanced machine learning and deep neural networks, capable of automatically learning complex dependencies between text and its emotional expression. These innovations pave the way for more accurate sentiment analysis and a deeper understanding of emotional dynamics in textual content.

Objective of the Article

The object of research is the process of determining the tone of textual information related to named entities. The subject of research is methods, algorithms, information technologies, models, and tools for determining the tone of textual information related to named entities.

Method of intelligent analysis of the tone of textual information in relation to named entities

The methodology of intellectual analysis of texts' emotional tome, especially in determining attitudes toward named entities, plays a significant role in decision-making in various fields, including commerce, media, finance, and politics. This field of research opens up broad prospects for developing and improving text analysis algorithms that take named entities into account [15].

Figure 1 shows the stages of the method of intelligent analysis of the tone of textual information in relation to named entities, which allows the selected text under study, using a neural network model of natural language processing, a lexical library for natural language processing, and a Ukrainian-language tonal dictionary, to obtain output data in the form of a conclusion about the tonality of the text under study, which includes a set of named entities, a numerical assessment of tonality in relation to each of the named entities for individual sentences; the value of the overall tone assessment for the entire text in regard to each of the named entities, the value of the tone assessment of the text for individual sentences of the text, and the value of the overall tone assessment for the entire text.

The main goal of the proposed approach is to identify positive, negative, or neutral sentiment in a text and to establish links between these sentiments and the specific named entities mentioned. This allows for determining how textual information is perceived concerning particular entities. The input data consists of textual posts describing events, activities, or individuals, primarily sourced from

"Ukrainska Pravda" [12]. While the method can handle texts of any length, processing extensive texts may reduce efficiency and speed, requiring more computational resources. Significant texts can be divided into smaller segments or sentences before being fed into the model to optimise performance.

A central component of the method is a neural language model responsible for assessing the emotional tone of the text. For this purpose, the Stanza model was selected [16]. In the first step, named entities are extracted using the neural model. The model identifies all mentions of an entity, including variations due to inflexion or case. The second step applies lemmatisation and stemming to consolidate these mentions. Lemmatization reduces words to their base form, often using Part of Speech (POS) tags to account for context and distinguish between words with different meanings [17]. For example, accurately extracting a person's name requires differentiating it from other words in the text, which can be achieved using NLP libraries such as NLTK, spaCy, or TextBlob. During the second step, variations of the same entity (e.g., «Петренка», «Петренко», «Петренкові») are grouped under the nominative form for sentiment evaluation.

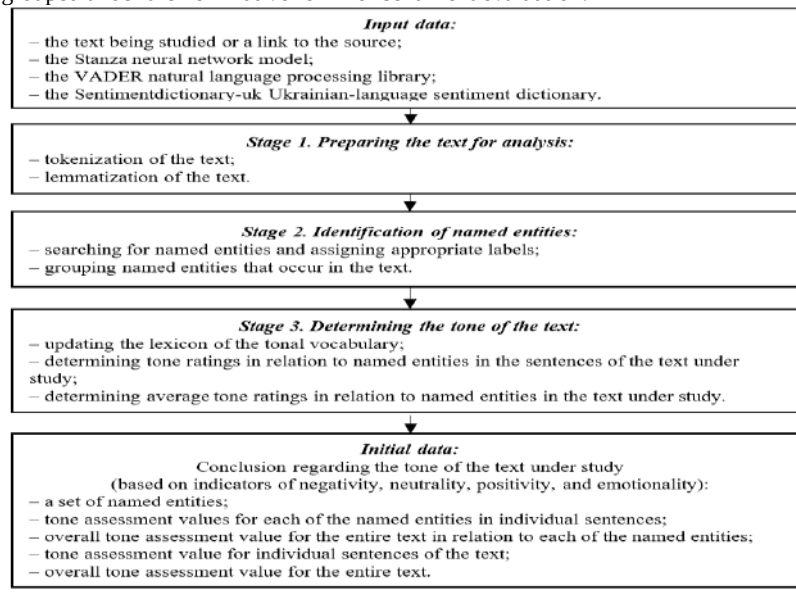


Figure 1. Stages of the method of intellectual analysis of the tone of textual information in relation to named entities

The third step involves assessing text sentiment regarding each named entity using the VADER approach [18]. VADER (Valence Aware Dictionary for sEntiment Reasoning) is an NLP algorithm that combines a lexicon-based approach with grammatical and syntactic rules to determine polarity and sentiment intensity. Its lexicon contains approximately 7,500 words, phrases, emoticons, and

abbreviations, each rated on a scale from -4 (extremely negative) to +4 (extremely positive) by 10 independent experts. Words not present in the lexicon are scored as 0 (neutral).

VADER also calculates a compound metric, a normalized indicator of overall sentiment ranging from -1 (strongly negative) to +1 (strongly positive), with values near 0 reflecting neutrality or mixed emotions. This metric provides a single numeric representation of the overall sentiment of the text, incorporating the intensity of individual words [19].

Table 2 illustrates an example of how sentiment scores are computed based on the writing style of the text, demonstrating the method's ability to capture nuanced emotional tones related to named entities.

Table 2

An example of calculating the emotional tone of a text, depending on the style of writing the message

Input text	Negative level	Neutral level	Positive level	Compound
Цей комп'ютер був гарною покупкою.	0	0.58	0.42	0.44
*Цей комп'ютер був **дуже** гарною покупкою.*	0	0.61	0.39	0.49
*Цей комп'ютер був дуже гарною покупкою**!!***	0	0.57	0.43	0.58
*Цей комп'ютер був дуже гарною покупкою!! **:)***	0	0.44	0.56	0.74
*Цей комп'ютер був **ДУЖЕ** гарною покупкою**!! :)***	0	0.393	0.61	0.82

To calculate the compound, VADER scans the text for known sentiment features, modifies the intensity and polarity according to the rules, sums up the ratings of the features found in the text, and normalizes the final rating to (-1, 1) using the following function (1):

$$compound = \frac{x}{\sqrt{x^2 + \alpha}}, \quad (1)$$

where x is the sum of the scores for all words in the text that have an emotional connotation (each word has a predefined score according to its positive or negative connotation); α is a parameter used to normalize the sum of the scores of word x (in VADER, it is set to 15, which is considered to approximate the maximum expected value of x). The function normalizes the sum of scores to not exceed the scale from -1 to 1. This makes sentiment assessment more comparable between texts of different lengths and emotional intensity [20].

Step 4 involves determining the tone based on the values obtained in the previous step using the corresponding scale, where compound >= 0.5 is negative, compound > -0.5 or compound < 0.5 is neutral, and compound <=-0.5 is positive.

The last step of the method is to form a conclusion about the tonality of the text post in relation to named entities. The process can return the emotional tonality value for each sentence and an overall assessment of the named entity. For example,

a person or organization may appear several times in the text and have a different tone in different sentences.

The output data of the method is a conclusion about the tonal state of the text in relation to the named entity. It should be noted that if a person, event, or place appears several times in the text, the method will return a value for the tone for each sentence separately, as well as an overall assessment in the context of the entire text.

Thus, intelligent analysis of the emotional tone of text, focusing on named entities, is an essential tool for automating text data processing, which contributes to the formation of more reasonable and informed decisions based on in-depth analysis of extensive text databases. As a result of this section, a method for the intelligent analysis of the tone of textual information in relation to named entities was developed, and the steps of its operation were described in detail. To implement the intelligent analysis of the tone of textual information in relation to named entities, it is necessary to retrain VADER on a Ukrainian-language dataset, which will allow determining sentiment in Ukrainian-language posts without resorting to machine translation tools. The corresponding dataset should be marked as follows: word; discrete sentiment (from the range: -2, -1, 0, 1, 2).

The Ukrainian tonal dictionary [21] sentimentdictionary-uk [22] was selected. The Ukrainian tonal dictionary contains an extensive database: 3442 words of the Ukrainian language with explicitly defined emotional tonality. Each word has an assigned tonality value ranging from -2 to 2. This dictionary was constructed based on two key sources:

– the tone-dict-uk-manual.tsv file was formed by averaging the ratings provided by several experts. This process ensured an objective assessment of the emotional tone of each word.

– the tone-dict-uk-auto.tsv file was created by automatically expanding the base tone-dict-uk-manual dictionary. This was done using machine learning methods that apply vector representations of words, such as word2vec and lex2vec. This process also included human post-processing to ensure data accuracy and consistency.

The data in the dictionary is organised in a format that includes two columns separated by a tab: the first column contains the word, and the second contains its discrete tone. It is important to note that most words in the dictionary are given in their basic grammatical form. In addition, where possible, adverbs have been replaced with cognate adjectives to ensure greater analysis accuracy. The Ukrainian sentiment dictionary sentimentdictionary-uk contains 14,856 words and phrases presented in the Dictionary. Creating a specific sentiment dictionary for a particular language is a key aspect in developing sentiment analysis systems. For the Ukrainian language, there is only one freely available tonal dictionary (the lang-uk project) with 6,859 words, which was last updated in September 2016. Given the dynamics of language and changes in word usage, there is a need to develop a new tonal dictionary that reflects contemporary word usage. This need created a new dictionary, “tonSUM.2.0,” available in various formats.

Explanation of the dictionary layout: column 1 contains the word or phrase (Word//word combination); columns 2 to 9 contain the sentiment scores for the word or phrase listed in column 1; the 10th column shows the average sentiment score, which is calculated automatically (using built-in functions) by dividing the sum of the specified scores by their number. Figure 2 shows the distribution of words and phrases in the Ukrainian Sentiment Dictionary dataset. Score -2: 874 reviews, score -1: 1370 reviews, score 1: 1081 reviews, score 2: 117 reviews.

Figure 2. Distribution of words and phrases in the Ukrainian tonal dictionary dataset

The sentimentdictionary-uk dictionary contains a significantly larger amount of data. Figure 3 shows the distribution of words and phrases in the sentimentdictionary-uk dataset. The dataset contains 14,855 records: 1,006 reviews with a rating of -2, 2,093 reviews with a rating of -1, 5,093 reviews with a rating of 0, 5,844 reviews with a rating of 1, and 819 reviews with a rating of 2.

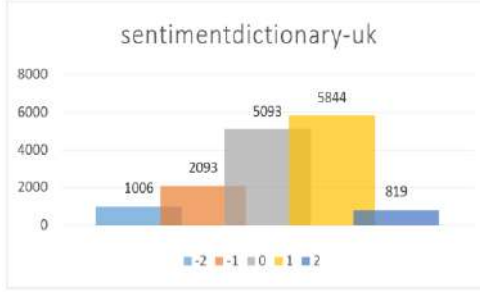


Figure 3. Distribution of words and phrases in the dataset “sentimentdictionary-uk”

Thus, combining data from datasets, a sample was obtained to supplement the VADER dictionary, containing 18,297 entries.

Approach to verifying library training for natural language processing for text sentiment analysis. A dataset containing 7,656 records was formed, of which 6,655 were used for training and 1,331 (about 20% of the training set) for

validation. This set is characterized by the presence of Russian language and slang and Russian-language elements in 37% of Ukrainian-language documents. This reflects specific linguistic dynamics in social networks after the start of the war. There is also a high frequency of spelling errors in the reviews.

The dataset obtained earlier contains binary ratings: recommend/do not recommend, and to validate VADER, it is necessary to check whether the method returns the correct "score" rating. Therefore, after downloading the validation sample, it will be checked whether the method returns a score >= 0.5 and the validation sample label "1" (recommended) – the test is considered passed. Accordingly, if the method returns a score <= -0.5 and the validation sample label "0" (do not recommend), the check is considered successful.

Experiment

To study the effectiveness of the method of intelligent analysis of the tone of textual information in relation to named entities, it is necessary to compare the results of the information system with automatic machine translation and subsequent determination of the tone of textual information in the text.

First, the software product was tested based on the intelligent determination of the tone of textual information in relation to named entities using the input parameter – an article from the resource "Ukrainska Pravda". The following sentence was entered for the study: "*Петро Василенко – безжальний вбивця! Хто б міг подумати, що виконавець пісні Мамині світлиці такий жорстокий!*". The program for determining the emotional tone of named entities, which is configured to work with English-language texts, returns the following results. Translated sentence: *"Petro Vasilenko is a ruthless killer! Who would have thought that the performer of the song of the mother's room is so cruel!"*. For comparison, the exact text was analyzed using Stanza and VADER, which had been pre-populated with tone dictionaries for the Ukrainian language. The method returned the following results. The result of running the program code to test the method of determining the emotional tone of text in relation to named entities for Ukrainian-language texts is shown in Figure 4. Using empirical approaches, a sample of data was collected to compare the performance of the methods (Table 3).

```
INFO:stanza:Using device: cpu
INFO:stanza:Loading: tokenize
INFO:stanza:Loading: mwt
INFO:stanza:Loading: lemma
INFO:stanza:Loading: ner
INFO:stanza:Done loading processors!
Entity: Петро Василенко, Type: PERS
Entity: Мамині, Type: PERS
Петро василенко – безжальний вбивець ! хто б могти подумати , що виконавець пісня мамині світлиця такий жорстокий !
Sentiment: {'neg': 0.533, 'neu': 0.349, 'pos': 0.118, 'compound': -0.8885}
```

Figure 4. Result of executing the program code for determining the emotional tone of the text in relation to named entities for Ukrainian-language texts

Table 3

An example of calculating the emotional tone of a text, depending on the style of writing the message

Input text		Negative level	Neutral level	Positive level	Compound
Нарешті жителям нашої громади покращили прибудинкові території! Тепер дітки можуть гратись на сучасних майданчиках, а мами можуть не хвилюватись за їх безпеку! Дякуємо!	Proposed method	0.00	0.744	0.256	0.6757
	Machine translation	0.00	0.886	0.114	0.4754
«Російський терорист не знає меж! Путін ніколи не зупиниться, українцям потрібно готуватись до тривалої війни», - коментар Мельника	Proposed method	0.411	0.519	0.07	-0.8808
	Machine translation	0.219	0.638	0.143	-0.3964
42-річний житель Хмельницького району, що вже відсидів свій строк за вбивство, сяде за вбивство вчергове. Підсудний вбив товариша по чарці, прийнявши його за російського солдата.	Proposed method	0.333	0.528	0.139	-0.8779
	Machine translation	0.292	0.601	0.107	-0.7068

To justify the superiority of the implemented method, experiments were conducted in which both methods (proprietary and machine translation-based) were compared on the same dataset. It is important to note that the implemented method shows a more distinct and precise distribution of sentiments (negative, neutral, positive), which indicates its greater sensitivity to the emotional nuances of the text.

The compound value indicates the overall sentiment of the text. The proprietary method gives a more accurate overall value that corresponds to the expected sentiment of the text, which indicates its advantage.

A heat map (Figure 5) and a bar chart (Figure 6) were used to visualize the results. Each row represents a sentiment analysis method (the implemented method and machine translation), and the columns reflect different aspects of sentiment: negativity, neutrality, positivity, and the overall compound indicator.

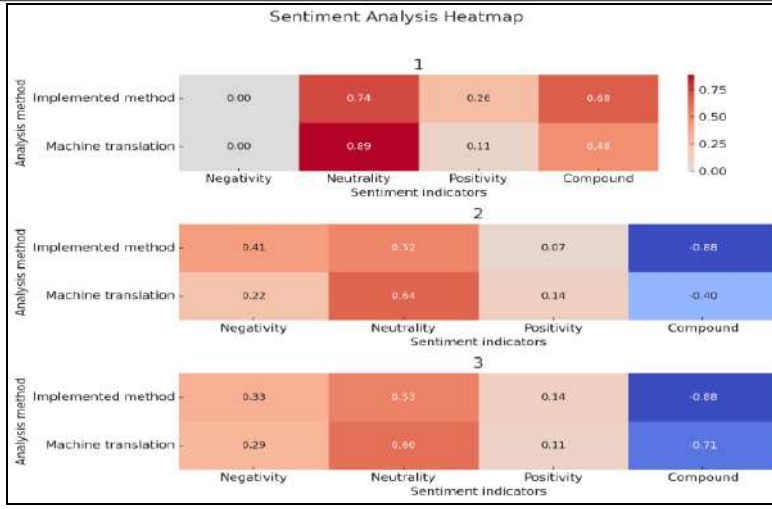


Figure 5. Heat map based on the obtained results

Based on data analysis and visualization using a heat map and bar chart, it can be concluded that the implemented method was more effective than using machine translation tools, particularly in determining the tone of textual information relating to named entities. In particular, our method demonstrated higher positive tone and compound values in the first text, indicating its ability to recognise positive nuances better. At the same time, for texts with negative sentiment, our method also showed a greater ability to identify negative emotions, as evidenced by lower (more negative) compound values compared to machine translation. This ability to more accurately determine sentiment is essential when analyzing context related to named entities, as it requires a deeper understanding of linguistic nuances and culturally specific expressions that machine translation is often unable to interpret correctly.

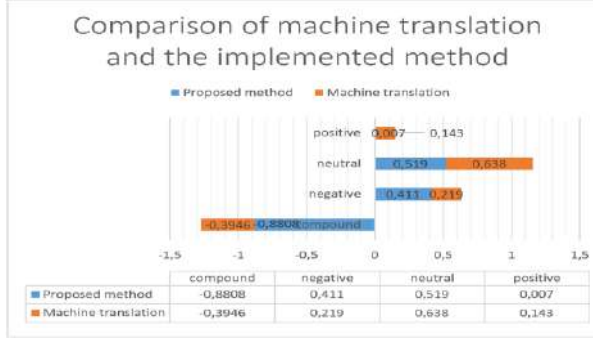


Figure 6. Distribution diagram of results

Overall, the implemented method showed better results in analyzing the tone of texts, making it a more reliable tool for sentiment analysis, especially in cases where contextual dependence and language specificity must be considered.

Conclusions and Prospects

The study addresses the problem of determining the sentiment of textual content ccerning named entities. A method for intelligent sentiment analysis of text information in relation to named entities has been developed, which enables, for a given text, the identification of key named entities and the assessment of sentiment toward each entity. The method combines a neural network-based natural language processing model, a lexical processing library, and a Ukrainian sentiment lexicon to produce outputs that include: a set of named entities, sentiment scores per entity for individual sentences, overall sentiment scores per entity for the entire text, sentiment scores for individual sentences, and overall text sentiment scores.

A corresponding software implementation was created to validate the proposed method. The study conducted the following tasks: a review of the current state of intelligent sentiment analysis regarding named entities, development of the sentiment analysis method, implementation of the process in software, and evaluation of its practical effectiveness. The results demonstrate the scientific novelty and innovation of the developed method. It allows for the analysis of Ukrainian-language texts. It provides sentiment assessment at both the sentence and text levels for each named entity, measuring negativity, neutrality, positivity, and emotionality.

Comparative evaluation indicates that the proposed method is more effective than approaches relying on translation into English followed by sentiment assessment. Overall, the developed approach can be applied to monitor public opinion and direct semantic analysis of individual texts, providing a reliable tool for sentiment assessment concerning named entities in Ukrainian-language content.

References

1. Taherdoost, H., & Madanchian, M. (2023). Artificial intelligence and sentiment analysis: A review in competitive research. *Computers, 12*(2), 37. https://doi.org/10.3390/computers12020037

2. Mazurets, O., Sobko, O., Dydo, R., Zalutska, O., & Molchanova, M. (2025). Transformer-based multilabel classification for identifying hidden psychological conditions in online posts. *CEUR Workshop Proceedings, 4004*, 347–361. https://ceur-ws.org/Vol-4004/paper26.pdf

3. Molchanova, M., Didur, V., Sobko, O., & Mazurets, O. (2025). Detection of web propaganda patterns by transformer neural networks: Improving efficiency via dataset balancing. *CEUR Workshop Proceedings, 3988*, 112–126. https://ceur-ws.org/Vol-3988/paper10.pdf

4. Murava, V., Zalutska, O., Didur, V., & Mazurets, O. (2025, June 25–27). Software architecture of information system for exchanging LLM thematic prompts. In *Proceedings of the IV International Scientific and Practical Conference Global Trends in the Development of Information Technology and Science* (pp. 121–127).

Stockholm, Sweden. https://isu-conference.com/wp-content/uploads/2025/06/Stockholm_Sweden_25.06.25.pdf

5. Andrushchenko, D., Klimenko, V., & Mazurets, O. (2025, May 28–30). Vector databases search for adaptive filtering of scientific articles. In *Proceedings of the XXII International Scientific and Practical Conference Scientific Trends in the Development of Modern Technologies* (pp. 189–195). Krakow, Poland. https://eu-conf.com/wp-content/uploads/2025/05/SCIENTIFIC-TRENDS-IN-THE-DEVELOPMENT-OF-MODERN-TECHNOLOGIES.pdf

6. Mazurets, O., & Ovcharuk, O. (2024, November 21). Efficiency research of method for detecting mental disorders by analysis of user content. In *Proceedings of the 11th International Conference on Information Technology Implementation (Satellite)* (pp. 46–47). Kyiv, Ukraine.

7. Mazurets, O., & Vit, R. (2024, November 21). Practical application of method of thematic classification of text information using LDA. In *Proceedings of the 11th International Conference on Information Technology Implementation (Satellite)* (pp. 151–152). Kyiv, Ukraine.

8. Mazurets, O., Tymofiiev, I., & Dydo, R. (2024, November 15). Approach for using neural network BERT-GPT2 dual transformer architecture for detecting persons depressive state. In *Proceedings of the VI International Scientific and Practical Conference Ricerche scientifiche e metodi della loro realizzazione* (pp. 147–151). Bologna, Italy. https://archive.logos-science.com/index.php/conference-proceedings/issue/view/29

9. Rendón-Cardona, P., Gil-Gonzalez, J., Páez-Valdez, J., & Rivera-Henao, M. (2022). Self-supervised sentiment analysis in Spanish to understand the university narrative of the Colombian conflict. *Applied Sciences, 12*(11), 5472. https://doi.org/10.3390/app12115472

10. Li, W., Shao, W., Ji, S., & Cambria, E. (2022). BiERU: Bidirectional emotional recurrent unit for conversational sentiment analysis. *Neurocomputing, 467*, 73–82. https://doi.org/10.1016/j.neucom.2021.09.057

11. Mohan, S., Solanki, A. K., Taluja, H. K., Anuradha, & Singh, A. (2022). Predicting the impact of the third wave of COVID-19 in India using hybrid statistical machine learning models: A time series forecasting and sentiment analysis approach. *Computers in Biology and Medicine, 146*, 105354. https://doi.org/10.1016/ j.compbiomed.2022.105354

12. Tymofiiev, I., Mazurets, O., Hardysh, D., & Molchanova, M. (2024, November 6–8). Neural network dual architecture for depression detection using cloud services. In *Proceedings of the XLVI International Scientific and Practical Conference Scientific Research in the Era of Digital Technologies: Challenges and Opportunities* (pp. 84–88). Barcelona, Spain. https://isu-conference.com/wp-content/uploads/2024/11/Scientific_research_in_the_era_of_digital_technologies_challenges_and_opportunities_November_6_8_2024_Barcelona_Spain.pdf

13. Dharma, E., & Muntina, E. (2022). The accuracy comparison among word2vec, GloVe, and fastText towards convolution neural network (CNN) text

classification. *Journal of Theoretical and Applied Information Technology, 100*(2), 54–64. http://www.jatit.org/volumes/Vol100No2/5Vol100No2.pdf

14. Yurchenko, D., Mazurets, O., Didur, V., & Molchanova, M. (2024, November 13–15). Approach to using cloud services for visual analytics of neural network analysis of texts emotional tonality. In *Proceedings of the XLVII International Scientific and Practical Conference The Future of Scientific Discoveries: New Trends and Technologies* (pp. 108–113). Marseille, France. https://isu-conference.com/wp-content/uploads/2024/11/The_future_of_ scientific_ discoveries_new_trends_and_technologies_November_13_15_2024_Marseille_France.pdf

15. Hladun, O., Mazurets, O., Molchanova, M., & Sobko, O. (2024, November 25–27). Real time detection the person emotion state using neural network. In *Proceedings of the II International Scientific and Practical Conference Scientific Research: Modern Innovations and Future Perspectives* (pp. 119–123). Montreal, Canada. https://www.eoss-conf.com/wp-content/uploads/2024/11/Montreal_Canada_25.11.24.pdf

16. Ukrainska Pravda. (n.d.). Retrieved from https://www.pravda.com.ua

17. Blazhuk, V., Mazurets, O., & Zalutska, O. (2024, October 23–25). An approach to using the mBERT deep learning neural network model for identifying emotional components and communication intentions. In *Proceedings of the XLIV International Scientific and Practical Conference The Impact of Scientific Research on the Development of the Modern World* (pp. 79–84). Dubrovnik, Croatia. https://isu-conference.com/wp-content/uploads/2024/10/The_impact_of_scientific_research_on_the_development_of_the_modern_world_October_23_25_2024_Dubrovnik_Croatia.pdf

18. Mazurets, O., Molchanova, M., Klimenko, V., & Prosvitliuk, M. (2024, June 12–14). Practice implementation of neural network model BART-Large-CNN for text annotation. In *Proceedings of the XXVII International Scientific and Practical Conference Prospects of Scientific Research in the Conditions of the Modern World* (pp. 97–102). Rotterdam, Netherlands. https://isu-conference.com/wp-content/uploads/2024/06/Prospects_of_scientific_research_in_the_conditions_of_the_modern_world_June_12_14_2024_Rotterdam_Netherlands.pdf

19. Sobko, O., Mazurets, O., Didur, V., & Chervonchuk, I. (2024, June 5–7). Recurrent neural network model architecture for detecting a tendency to atypical behavior of individuals by text posts. In *Proceedings of the XXVI International Scientific and Practical Conference Theoretical and Practical Aspects of Modern Research* (pp. 113–117). Ottawa, Canada. https://isu-conference.com/wp-content/uploads/2024/06/Theoretical_and_practical_aspects_of_modern_research_June_5_7_2024_Ottawa_Canada.pdf

20. Mazurets, O., Sobko, O., Molchanova, M., Zalutska, O., & Yurchak, A. (2024, May 31). Practical implementation of neural network method for stress features detection by social internet networks posts. In *Proceedings of the II International Scientific Theoretical Conference Global Science: Prospects and*

Innovations (pp. 160–167). Berlin, Germany. https://previous.scientia.report/index.php/archive/issue/view/31.05.2024/89

21. Lang-uk. (n.d.). *Ukrainian tonal dictionary*. GitHub. https://github.com/lang-uk/tone-dict-uk

22. Sentimentdictionary-uk. (n.d.). GitHub. https://github.com/Oksana504/sentimentdictionary-uk

23. Abiola, O., Abayomi-Alli, A., Tale, O. A., et al. (2023). Sentiment analysis of COVID-19 tweets from selected hashtags in Nigeria using VADER and Text Blob analyser. *Journal of Electrical Systems and Information Technology, 10*(5). https://doi.org/10.1186/s43067-023-00070-9

24. Bing, L. (2022). *Sentiment analysis and opinion mining*. Springer Nature. https://link.springer.com/book/10.1007/978-3-031-02145-9

МЕТОД АНАЛІЗУ ЕМОЦІЙНОЇ ТОНАЛЬНОСТІ УКРАЇНОМОВНИХ ТЕКСТІВ ІЗ ВИКОРИСТАННЯМ ЗАСОБІВ ОБРОБКИ ПРИРОДНОЇ МОВИ

О. Залуцька[0000-0003-1242-3548]
Хмельницький національний університетУкраїна
EMAIL: zalutsk.olha@gmail.com

Анотація. *Дослідження присвячене інтелектуальному аналізу емоційної тональності текстів, пов'язаних з іменованими сутностями. Запропонований метод поєднує модель обробки природної мови на основі нейронної мережі, лексичну бібліотеку NLP та україномовний словник термінів, що містять емоційну тональність Він надає результати у вигляді оцінок емоційного забарвлення іменованих сутностей на рівні речень і текстів, а також загальну оцінку емоційного тону проаналізованого контенту.Актуальність дослідження визначається зростаючою потребою в точному аналізі емоційної тональності в контексті великомасштабних цифрових інформаційних потоків. Визначення емоційного ставлення до конкретних осіб, організацій або подій має важливе значення для моніторингу громадської думки, сприйняття бренду, політичного дискурсу та аналізу фінансових ринків.Наукова новизна полягає у розробці та впровадженні методу, який підтримує тексти українською мовою та оцінює сентимент за такими вимірами, як негативність, нейтральність, позитивність та емоційність. Практичне значення полягає у створенні програмної системи, здатної здійснювати семантичний аналіз сентименту текстового контенту, досягаючи вищої ефективності, ніж підходи, засновані на машинному перекладі. Розроблений метод може аналізувати громадську думку, реакції в соціальних мережах, ринкові тенденції та окремі тексти.*

Ключові слова: *розпізнавання іменованих сутностей, емоційна тональність, визначення емоційної тональності, Stanza, VADER*

Section 2. Intelligent systems and data analysis

UDC 004.942
DOI https://doi.org/10.36059/978-966-397-538-2-8

MODELING OF NONLINEAR DYNAMICS USING THE SUPPORT MODEL METHOD

Dr.Sci. O. Fomin[0000-0002-8816-0652], **Dr.Sci. S. Polozhaenko**[0000-0002-4082-8270], **Ph.D. A. Savieliev**[0000-0002-6949-5959], **O. Tataryn**[0009-0005-3888-6569]
Odesa Polytechnic National University, Ukraine
EMAIL:fomin@op.edu.ua

Abstract. *This work is devoted to resolving the contradiction between the speed of constructing nonlinear dynamic models and their accuracy. The goal is to reduce the time required to build nonlinear dynamic models in the form of neural networks while ensuring the specified modeling accuracy. This goal is achieved by developing a modeling method based on the use of a set of pre-trained neural networks (support models) that reflect the main properties of the subject area. The scientific novelty of the developed method lies in the use of pre-trained neural networks with time delays as support models. Unlike existing pre-training methods, the developed method allows building simpler models with less training time while ensuring the specified accuracy. The practical benefit of the results of the work lies in the development of an algorithm for the support model method, which allows for a significant reduction in the training time of neural networks with time delays without losing modeling accuracy.*

Keywords: *identification; nonlinear dynamics; pre-training; neural networks, training speed.*

Introduction

The current stage of modelling development, which is mainly based on the use of intelligent technologies, is characterized by a number of requirements for both high accuracy of models and speed of their construction [1].

High accuracy in modelling nonlinear dynamics is currently achieved using machine learning methods, in particular neural networks (NN) [2, 3, 4]. However, the application of such methods is often associated with increased computational complexity, which leads to significant time cost on model construction [3, 4, 5].

The problem of reducing modelling time remains one of the most pressing, especially in areas related to the personalization of models that must adapt to changes in user behaviour or the environment (e.g., in authentication tasks, biomedical applications, human-machine systems), working in real time [6, 7].

Analysis of Previous Studies

One common approach to solving the problem of increasing the learning speed of neural networks is to optimize the architecture of models, which involves reducing the number of model parameters without significantly compromising their performance. Examples of such approaches include network simplification methods such as pruning, quantitation, and model compression [8, 9, 10].

Another relevant direction for increasing the training speed of NN is the use of accelerated learning algorithms, such as stochastic gradient descent with momentum (SGD with momentum) and adaptive optimization methods (e.g., Adam, RMSprop) [11, 12]. These methods make it possible to accelerate NN convergence thanks to an improved weight update strategy and a reduction in the number of epochs required to achieve the specified accuracy.

Another way to accelerate the NN training process is transfer learning [13, 14]. The main advantage of this method is the ability to use models pre-trained on a dataset from one subject area to solve target tasks from another subject area. This approach is used when the amount of specific data is limited or there are no high-quality datasets for the target task. Transfer learning is particularly effective when working with large neural networks, such as deep convolutional networks (CNNs), which require significant computational resources for training [15, 16]. One common approach to solving the problem of increasing the learning speed of neural networks is to optimise the architecture of models, which involves reducing the number of model parameters without significantly compromising their performance. Examples of such approaches include network simplification methods such as pruning, quantization, and model compression [17].

A special case of transfer learning can be considered preliminary training, in which the model is first trained on a large set of general data and then fine-tuned on more specific data for the target task [18, 19]. This approach significantly reduces the time and computational resources used during the fine-tuning of the target model, compared to training the model entirely on the target task data. As a result, pre-trained models converge faster and require fewer resources to achieve the desired model quality [18].

Transfer learning and pre-training technologies have become an integral part of many research and development projects in the field of artificial intelligence. They have proven their effectiveness in natural language processing tasks (BERT natural language processing network, GPT text generation network) [20], computer vision systems (pre-trained convolutional networks DenseNet, VGG) [17], biomedical research, and human-machine interfaces [20].

The widespread use of this approach to building models based on pre-training NN has been made possible by its practical advantages: a significant reduction in the cost of training models and an increase in their effectiveness in real-world applications, where not only accuracy but also speed is important. In addition, pre-trained models can be easily adapted to new tasks, making them indispensable tools in dynamically changing requirements and tasks.

This direction seems promising in the tasks of identifying nonlinear dynamic objects. At the same time, there is a lack of work in the field of preliminary training of neural networks that simulate the nonlinear dynamic properties of objects with continuous characteristics.

Based on the above, this paper develops an approach to building NN based on pre-training, which is capable of effectively coping with the requirements of modern modelling tasks, in particular the identification of nonlinear dynamic objects.

Unresolved Issues. The spread of the approach to building models based on prior training of neural networks has been made possible by its practical advantages: a significant reduction in the cost of training models and an increase in their effectiveness in real-world applications, where not only accuracy but also speed is important. In addition, pre-trained models can be easily adapted to new tasks, making them indispensable tools in dynamically changing requirements and tasks.

This direction also looks promising in the identification of nonlinear dynamic objects. At the same time, there is a lack of work in the field of pre-training NN that simulates the nonlinear dynamic properties of objects with continuous characteristics.

Based on the above, the paper develops an approach to building NN based on pre-training, which is capable of effectively coping with the requirements of modern modelling tasks, in particular the identification of nonlinear dynamic objects.

Problem statement

In practice, the use of prior training of neural networks involves considerable time expenditure on building a general model [14, 15]. To achieve the research goal, it is necessary to invent a method for accelerating the construction of general models of nonlinear dynamics. The formal formulation of the problem of accelerating the construction of general models of nonlinear dynamics based on prior training of neural networks is as follows. Let S be the domain for which there is enough labeling data N_S (D_S dataset):

$$D_S=\{(\mathbf{x}_i^S, y_i^S)\}, \quad (1)$$

where $\mathbf{x}_i^S$ is a vector of independent variables, y_i^S is the corresponding target variable (label), $i=1,\ldots, N_S$.

Let $f_{\theta S}$ be the general NN model with θ_S parameters, which is trained on D_S dataset.

Let T be the target task in the subject area S, for which there are marked-up data of limited N_T size (D_T dataset):

$$D_T=\{(\mathbf{x}_j^T, y_j^T)\}, \quad (2)$$

where $\mathbf{x}_j^T$ is the vector of the independent variables, y_j^T is the corresponding target variable (label), $j=1,\ldots, N_T$.

Let $f_{\theta T}$ be the target NN model with θ_T parameters, trained on D_T dataset, which ensures the accuracy $E_{\theta T}$ and the duration $t_{\theta T}$ of target model training.

It is necessary to find the parameters θ_S of the general model. Using them as starting values θ_{T0} during training, the target model $f_{\theta T}$, a specified accuracy level $E_{\theta T}$ is achieved in the minimum time $t_{\theta T}$:

$$\theta_{T0} = \theta_S : \underset{t_{\theta T}}{arg\ min} L_T(f_{\theta T}(\mathbf{x}_j^T), y_j^T) = E_{\theta T} \quad (3)$$

where L_T is the loss function adopted for the target model.

The objective of the article is to reduce the time required to build nonlinear dynamic continuous-time models in the form of neural networks while ensuring the specified modelling accuracy by developing a method based on prior training of neural network models.

To achieve this aim, the following tasks were formulated.

1. Development of a modelling method based on prior training by superimposing a set of pre-trained neural networks (support models) that reflect the main characteristics of the subject area.

2. Construction of support models in the form of neural networks that reflect the main nonlinear and dynamic characteristics of the subject area.

3. Investigation of the speed of modelling complex nonlinear dynamics using the developed support model method.

Support models method

The principle of identification based on support models. An approach to NN training based on the use of prior training in practice may result in a slight increase or, in general, a decrease in the training performance of the target model.

One of the factors reducing the productivity of the target model training process is the general nature and large volume of the training dataset D_{S0}, which must contain a description of the object's behavior in the widest possible range of external conditions and under the action of a wide range of input signals [66, 94]. As a result, in a specific case, when solving object modelling problems in a narrower range of external conditions and input influences, the coarse $f_{\theta S0}$ and accurate $f_{\theta Ti}$ models (i=1,…,g, where g is the number of target modelling problems) have excessive complexity (Fig. 1).

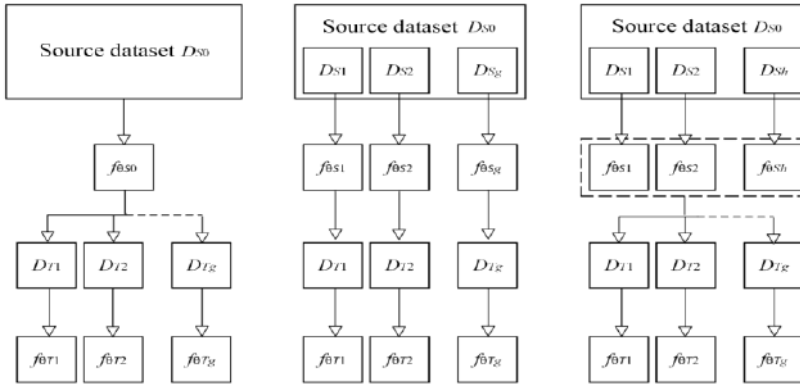
There are two ways to overcome this problem.

1. Forming a set of separate specialized training datasets D_{Si} (i=1,…,q, q – number of specialized training datasets) for building rough models $f_{\theta Si}$ with subsequent construction of accurate models $f_{\theta Ti}$ by further training the rough model on the target dataset D_T (Fig. 1b). In this case, the advantages of prior training are largely lost, since for each target task it is necessary to form a separate training dataset D_{Si} and train a rough model $f_{\theta Si}$. This leads to time losses when solving each target task.

The use of an approach based on the formation of separate specialized training datasets largely negates the advantages of prior training, since each new task requires the creation of a separate dataset and the training of a new model. This leads to significant time costs even at the task setting stage.

2. Formation of a set of separate training datasets D_{Sj} (j=1,…,h, where h is the number of basic characteristics reflecting the individual properties of the subject

area), to build support pre-trained neural networks $f_{\theta Sj}$ with subsequent construction of accurate models $f_{\theta Ti}$ by retraining a rough model in the form of a superposition of support pre-trained neural networks on the target dataset D_{Ti} (Fig. 1c).



a b c

Figure 1. Structural diagram of the preliminary training process: a – preliminary training of a universal rough model; b – preliminary training of specific rough models; c – preliminary training of support models

The result of this approach is a set of support pre-trained NN $f_{\theta Sj}$, which are determined only once. At the same time, each of the basic NN $f_{\theta Sj}$ is significantly simpler than the rough model $f_{\theta S0}$.

The coarse model of the object $f_{\theta Si}$, which contains combined characteristics in the form of a superposition of typical dynamic and nonlinear links, is constructed on the basis of a set of support pre-trained NN $f_{\theta Sj}$ that correspond to the existing characteristics of the object. At the same time, the structure of the coarse model $f_{\theta Si}$ (the dimension of the parameter vector θ_{Si}) must coincide with the structure of the support NN $f_{\theta Sj}$ (the dimension of the parameter vector θ_{Sj}):

$$f_{\theta Si} : dim(\theta_{Si}) = dim(\theta_{Sj}), \quad (4)$$

which ensures the simplicity of the accurate model $f_{\theta Ti}$.

This approach assumes that a coarse model is built not on the basis of a lengthy preliminary training process, but on the basis of the superposition of several simple models (support models). Each support model describes a separate property of the subject area, and their superposition creates a refined coarse model.

The advantage of this approach is a significant increase in the speed of coarse model synthesis by eliminating the lengthy training phase of the coarse model for each new task in the subject area under consideration. In addition, the modularity of this approach makes it possible to use support models in different tasks, which reduces the time required to build models and provides flexibility and ease of replacement or improvement of individual support models.

The approach used in this work consists of using g support datasets D_{Sv} (v=1,…,g, g – the number of basic characteristics of the subject area), each of which describes a separate basic property of the area under study. Based on these datasets, support pre-trained models $f_{\theta Sv}(D_{Sv})$ with parameters θ_{Sv} are constructed. By combining and adapting the corresponding support models, a rough model $f_{\theta Sv}(D_{Sv})$ is constructed, which has a certain set of characteristics (non-linear and dynamic) of the object under study. The target model $f_{\theta Tk}(\theta_{Sv}, D_{Tk})$ of objects with certain characteristics is built by retraining the rough model on the D_{Tk} dataset.

The method of constructing complex models based on the superposition of simpler models is actively used in various scientific fields, such as theoretical physics, engineering, and, less frequently, in the fields of mathematical modelling and machine learning. Work in these fields shows that this method makes it possible to effectively solve complex system modelling problems, improving the accuracy and flexibility of models by dividing the problem into simpler subtasks. These methods are based on the principle of dividing a complex problem into subtasks, which simplifies the modelling process and improves the accuracy of solutions. Today, most of the efforts of researchers to solve the problems of identifying complex objects and improving the accuracy and speed of modelling are focused on the use of a rough model with subsequent retraining of an accurate model. Not enough attention is paid to the use of support models, although this approach seems promising due to its flexibility and modularity, which provides the following advantages.

1. Flexibility in development: Each support model is responsible for its own aspect of the system and can be easily replaced or improved without affecting the overall structure. This makes the development process modular and more manageable.

2. Simplification of complex tasks: Dividing a task into several subtasks with simplified models makes the process of modelling complex systems more understandable and easier to calibrate.

3. High accuracy: Synthesizing a target model from several base models allows for multiple aspects of the system to be taken into account, which ultimately leads to more accurate predictions.

The method of support models

To accelerate the construction of general models of nonlinear dynamics, a new approach is proposed in this paper. It consists in using a set of separate pre-trained NN (support models), each of which reflects separate basic characteristics of the area.

To construct a set of support models, a set of support datasets D_{Rk} is used (k=1,…,g, g is the number of basic characteristics of the domain). Each of the support datasets D_{Rk} describes a separate basic characteristic of the domain. On the basis of these datasets, support models $f_{\theta Rk}$ with the parameters θ_{Rk} are built. By combining support models that reflect the characteristics of the target problem, a general model $f_{\theta S}$ is built. It has a set of specified properties (nonlinear and

dynamic). The target model $f_{\theta T}(\theta_S, D_T)$ is built by pre-training the general model $f_{\theta S}$ obtained on the basis of the set of support models $f_{\theta Rk}$ on the D_T dataset.

This approach allows to preserve the advantages of pre-training, since support models obtained once can be repeatedly used for different domains and target tasks, significantly reducing the total time and resources for training models without collecting additional data [17, 18, 19]. The structural scheme of the training process based on support models is presented in Fig. 2.

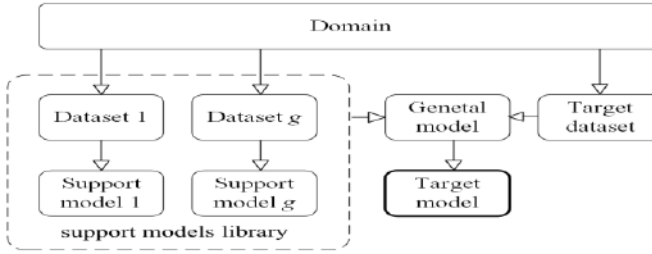


Figure 2. Structural diagram of the training process using support models

The algorithm of the suggested method consists of the following steps.

Step 1. Selection of basic domain properties and formation of a set of datasets D_{Rk} reflected the selected properties.

Step 2. Construction of support models set $f_{\theta Rk}$ in the form of separate NN corresponding to the established properties of the domain. Training of built models based on the generated datasets D_{Rk}.

Step 3. Determination of the list of p properties of the target problem from the set of g basic properties of the domain and construction of the general model $f_{\theta S}$ based on the superposition of the corresponding support models $f_{\theta Rh}$ (h=1,…,p, $p \le g$) obtained in *Step* 2.

Step 4. Training of the target model $f_{\theta T}(\theta_S, D_T)$ based on the general model $f_{\theta S}$ obtained in *Step* 3.

Step 5. Determination of the accuracy indicators $E_{\theta T}$ and training time $t_{\theta T}$ of the target model $f_{\theta T}$. In case of unsatisfactory quality indicators of the target model $f_{\theta T}$, the control transfers to *Step* 2 to correct the structure θ_{Rk} of the support models $f_{\theta Rk}$, and, if necessary, to *Step 1* to correct the set of basic properties of the domain and the set of datasets D_{Rk}, reflecting the selected properties.

Selecting the basic properties of the domain for forming datasets

The basic properties of the domain in the work are understood as the characteristics of objects that reflect the essential features of their behavior. These properties can include real or abstract parameters that are important for solving the modeling problem [20, 21].

The procedure for selecting the basic properties of the domain and generating datasets D_{Rk} reflecting the selected properties is as follows.

1. Defining the range of tasks to be solved in a given domain; analyzing properties that are important for objects in a given domain, significantly affect the results of modeling and should be taken into account when forming the dataset D_S.

2. Determination of signal types (for example, periodic, random, pulsed) that best reflect the properties of the objects under study.

3. Formation of the dataset D_S based on the list of basic properties of the domain established in paragraph 1, the set of input signals and reactions of the object formed in paragraph 2.

4. Segmentation of the dataset D_S into separate datasets D_{Rk} according to the defined list of basic properties of the domain.

In the above sequence of steps, the task of determining the type of signals that best reflect the properties of the object under study remains the least formalized. Automating the selection of support models is crucial for scaling the method to complex, multidimensional dynamical systems and bridges the gap between human intuition and automated processing, allowing the application of machine learning methods for objective identification and segmentation.

Determination of the structure of support models and their pre-training

To modelling nonlinear dynamics, the work uses time-delayed NN (TDNN) [21, 22, 23]. Due to their simplicity and versatility, TDNNs are most widely used in modeling problems of nonlinear dynamic objects. In practice, the TDNN structure is most often used, consisting of three layers: input, hidden, and output [22]. The size of the layers in this TDNN structure is determined as follows: the input layer consists of M neurons and is responsible for the memory (dynamic characteristics) of the model; the hidden layer consists of K neurons and is responsible for the nonlinear characteristics of the model; the output layer contains the number of Y neurons, which is equal to the number of outputs of the model.

For each support model, a labeled data set $D_{Si}=\{(x^{Pk}{}_{Hj}(t), f_{Vi}[x^{Pk}{}_{Hj}(t)])\}$ is formed based on the signals $x(t)$ at the input of the object and the responses $y(t)= f_{Vi}[x(t)]$ at its output. Typical signals are often used as input signals: impulse $x(t)=a\delta(t)$, step $x(t)=a\Theta(t)$, linear $x(t)=at$, and harmonic $x(t)=a\,sin(t)$ signals of various amplitudes $a \in (0, 1]$. Time delays at the input are implemented through shifts in time series and the inclusion of previous values in the input vector.

The main steps for implementing time delays and forming a training sample are as follows:

Step 1. Selecting the number of delays. Determine the number of delays M that will be used.

Step 2. Forming the input vector. For each time moment t_k, the input vector is formed as a sequence of current and previous values.

Step 3. Transferring delays to the network. Each formed input vector is transmitted to the input layer of the neural network, where it is processed as a regular input.

Construction of a general model based on the superposition of the corresponding support models

After completing the process of pre-training the set of support models $f_{\theta Rk}$, a general model $f_{\theta R}$ is built on their basis. This model is composed of a set of p support models $f_{\theta Rh}$ that correspond to the available basic properties of the object. The selection of support models that reflect the basic properties of an object is generally subjective. To reduce subjectivity and increase the reproducibility of the selection of support models, it is advisable to use methods of clustering input data or feature contribution analysis.

After completing the preliminary training process for the family of support models that correspond to the existing basic characteristics of the object, a rough model $f_{\theta S}(D_S)$ is constructed based on them. Assuming that the general $f_{\theta S}$ and support $f_{\theta Rk}$ models are constructed in the form of NN with the same structure (dimension of the parameter vector $dim(\theta_S)=dim(\theta_{Rk})$), the definition of the general model is reduced to calculating the arithmetic operations on corresponding components of the parameter vectors of the support models θ_{Rk}.

At the same time, several approaches to the superposition of support models are considered:

additive superposition is used when each support model is responsible for independent aspects of the system (e.g., dynamics and environmental impact). The outputs of the support models $f\theta Si(DSi)$ are determined based on the calculation of the arithmetic mean of the corresponding components of the parameter vectors of the support models θSv:

$$\theta_S^i = \frac{1}{h}\sum_{k=1}^{h}\theta_{Rk}^i \tag{5}$$

where i are the indices of the corresponding elements of the parameter's vectors of the general θ_S and the support θ_{Rk} models.

multiplicative superposition is used in the case of interacting processes (e.g., where some processes modify others). In this case, the outputs of the support models $f_{\theta Sv}(D_{Sv})$ are multiplied:

$$\theta_S^i = \prod_{v=1}^{b}\theta_{Sv}^i \tag{6}$$

combined superposition methods use complex combinations of support models, such as weighted sums of outputs of nonlinear functions to combine the outputs of several models, the application of nonlinear functions to the outputs of individual models, etc. Such methods include determining the parameter vector θ_S of the approximate model as the maximum value among the corresponding components of the parameter vectors of the support models θ_{Sv}:

$$\theta_S^i = max(\theta_{Sv}^i), v = \overline{1,b} \tag{7}$$

Thus, the advantage of forming an approximate model using the support models method is the absence of a training procedure, which reduces computational

complexity and significantly speeds up the process of building an approximate model. At the same time, the dimension of the approximate model $f_{\theta S}$ (the dimension of the parameter vector θ_S) remains the same as in the base models, i.e., the complexity of the approximate model does not increase. Another advantage of forming a general model using expressions (5)-(7) is the absence of a training procedure, which significantly speeds up the process of constructing a approximate model.

Experiment setup

The study of the support models method is carried out on the example of a nonlinear dynamics test object. A simulation model of a test object in the form of a sequence of a nonlinear link with saturation and a dynamic link of the first order [24, 25] is shown in Fig. 3.

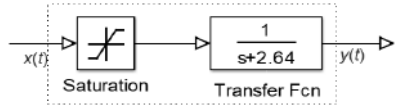


Figure 3. Test object simulation model

As typical characteristics from the set of properties of the domain of nonlinear dynamics, a nonlinear characteristic in the form of saturation and a dynamic link of the first order were chosen to describe the behavior of the test object. For the test object, a labeled D_S dataset generated on the base of signals $x(t)$ at the input of the object and the responses $y(t)$ at its output. The inputs are pulsed $x(t)=a\delta(t)$, stepped $x(t)=a\Theta(t)$, linear $x(t)=at$ and harmonic $x(t)=a\cdot sin(t)$ signals of different amplitudes $a \in (0, 1)$. Based on the dataset D_S, two support datasets are formed:

input stepped signals $x(t)=a\Theta(t)$ and responses $y(t)$ of the object with nonlinearity in the form of saturation D_{R1};

input stepped signals $x(t)=a\Theta(t)$ and responses $y(t)$ of the object in the form of a dynamic link of the first order D_{R2}.

The experiment consists in studying the training speed of the target model of the test object, built by various methods:

based on the general model $f_{\theta S}(D_S)$, pre-trained on the general D_S dataset;

based on individual support models $f_{\theta R1}(D_{R1})$, $f_{\theta R2}(D_{R2})$, pre-trained on datasets D_{R1} and D_{R2};

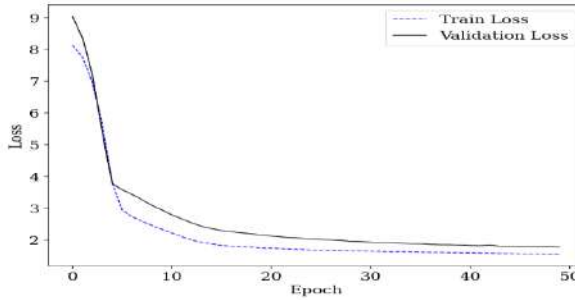
based on the general model $f_{\theta R}(f_{\theta R1}, f_{\theta R2})$ in the form of a superposition of support models $f_{\theta R1}(D_{R1})$ and $f_{\theta R2}(D_{R2})$.

Simulation and results

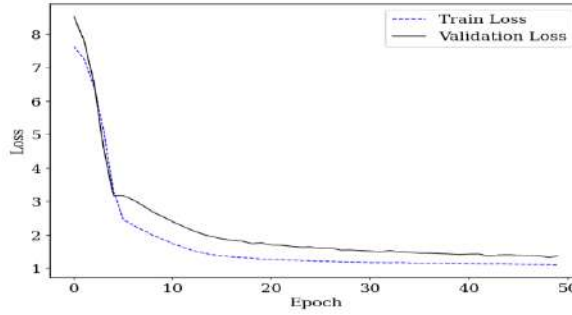
Building a rough model. To determine the structure of the model $f_{\theta S}(D_S)$, which is a three-layer TDNN, based on the results of additional research, the number of neurons M=30 in the input layer and K=30 in the hidden layer was accepted. The model is trained using the backpropagation method with network parameter

updating using the Levenberg-Marquardt method. Preliminary training is limited to 50 epochs to prevent overfitting and preserve adaptability.

Fig. 4 shows the dependence of loss functions (*mse*, *mae*) during training of the coarse model on the number of training epochs.



a



b

Figure 4. Dependence of loss functions during training of a rough model on the number of training epochs: *a – mse*, *b – mae*

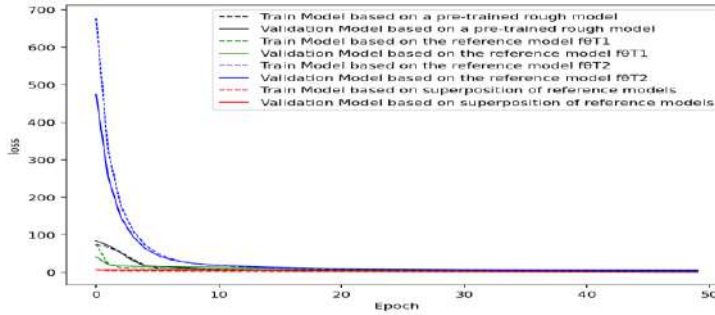
Construction of an accurate model. The structure of the accurate model $f_{\theta T}(\theta_S, D_T)$ based on the pre-trained coarse model $f_{\theta S}(D_S)$, models $f_{\theta TK}(D_{Tk})$ based on separate support models of datasets D_{T1} and D_{T2}, and model $f_{\theta T}(D_T)$ based on the superposition of support models is selected to be identical to the coarse model $f_{\theta S}(D_S)$ in the form of a three-layer TDNN. The NN is trained using the backpropagation method with network parameter updating using the Levenberg-Marquardt method. Training of the accurate model is carried out over 50 epochs.

Fig. 5 shows the dependence of loss functions (*mse*, *mae*) on the number of training epochs for accurate models based on the coarse model, based on separate support models $f_{\theta T1}(D_{T1})$ and $f_{\theta T2}(D_{T2})$, and based on the superposition of support models. The results of the experiment on studying the training speed of the accurate model of the test object, built on the basis of a rough model, on the basis of separate support models, and on the basis of a superposition of support models, are given in Table 1.

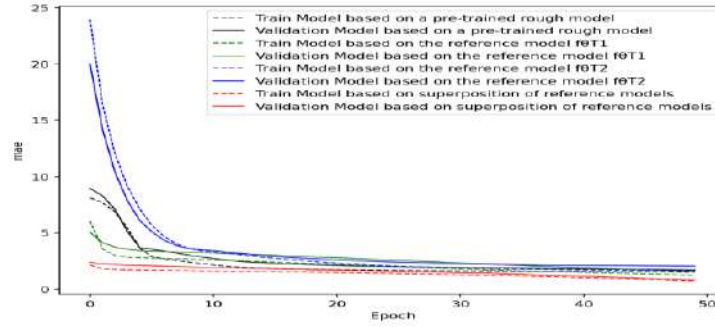
Table 4

Results of an experiment to study the learning speed of a target model of a test object

No	Model based on	Training		Validation		Number of epoch
		mse	*mae*	*mse*	*mae*	
1	general model	4,9105	1,8890	6,9738	2,3972	14
		2,2267	1,2686	2,9774	1,4809	50
2	support model (saturation)	4,9596	1,8623	6,8963	2,4457	8
		2,7553	1,4094	4,1863	1,7621	50
3	support model (first-order dynamic link)	4,8058	1,8544	6,8876	2,2571	23
		3,9412	1,6869	5,7969	2,0801	50
4	superposition of support models	4,5837	1,8347	6,6431	2,2256	3
		0,9254	0,7941	1,0704	0,8807	50



a



b

Figure 5. Dependencies of loss functions during training of accurate models based on a coarse model, separate support models, and superposition of support models on the number of training epochs: *a – mse*, *b – mae.*

Fig. 5 and Table 1 show the advantage of using pre-trained support NN during the identification of nonlinear dynamic objects, namely, a significant reduction in the training time of the TDNN model (4.6 times) compared to the traditional approach of building an accurate model based on a pre-trained coarse model with comparable accuracy of both models (established quality requirements at the validation stage: *mse*=7.0, *mae*=2.5). The use of individual support models as coarse models can also reduce the training time of the accurate model (by 1.8 times) with comparable accuracy of both models.

Conclusions and prospects

The paper successfully solves the problem of reducing the time of building nonlinear dynamics continuous-time models in the form of neural networks while ensuring the specified accuracy of modeling. To resolve the conflict between the accuracy of modeling nonlinear dynamic objects and the speed of model construction, a modeling method was developed based on pre-training through the superposition of support models that reflect the basic properties of the subject area.

The effectiveness of the developed method for modeling nonlinear dynamics was proven when solving the problem of modelling a test nonlinear dynamic object. The experiment demonstrates a 4.6-fold reduction in the time of building a target model using support models compared to the traditional modeling method based on pre-training. The advantages of the proposed approach are the ability to quickly adapt to changing operating conditions, high speed of building the target model while ensuring the specified modeling accuracy. In addition, the developed method allows improving the efficiency of model training in the lack of labeled data for the target task. The disadvantages of the proposed approach, inherited from methods based on pre-training, are the dependence of the modeling results on the quantity and quality of data of the target dataset.

The practical limitations of the application of the proposed method are the a priori need for support models built on a sufficient amount of qualitative data. Insufficient data or poor data quality can significantly reduce the accuracy of support models and, as a result, significantly reduce the training time of an accurate model.

Thus, the area of effective application of the proposed method is allocated: lack of marked data of the target task in the presence of a general dataset of sufficient size; no significant discrepancies between the characteristics of the general and target datasets.

To improve and expand the scope of application of the support models method, it is necessary to take into account the real conditions of the external environment by expanding the experimental part at different levels of noise distortion and under conditions of time drift of the observed parameters. In order to fully assess the potential of the proposed method and determine its place among advanced solutions in further research, it is planned to expand the range of test objects, including systems with different dynamics, multidimensional systems, objects with delays, etc.

References

1. Chisty, N. M. A., & Adusumalli, H. P. (2022). Applications of artificial intelligence in quality assurance and assurance of productivity. *ABC Journal of Advanced Research, 11*(1), 23–32. https://doi.org/10.18034/abcjar.v11i1.625

2. Sen, J. (2021). *Machine learning – algorithms, models and applications*. IntechOpen. https://doi.org/10.5772/intechopen.94615

3. Li, M. (2025). The past decade and future of the cross application of artificial intelligence and decision support system. In *2025 IEEE 6th International Seminar on Artificial Intelligence, Networking and Information Technology (AINIT)* (pp. 1666–1669). IEEE. https://doi.org/10.1109/AINIT65432.2025.11036058

4. Chitty-Venkata, K. T., Emani, M., Vishwanath, V., & Somani, A. K. (2023). Neural architecture search benchmarks: Insights and survey. *IEEE Access, 11*, 25217–25236. https://doi.org/10.1109/ACCESS.2023.3253818

5. Adikari, I., Lakmali, S., Dhananjaya, P., & Herath, D. (2023). Accuracy comparison between recurrent neural networks and statistical methods for temperature forecasting. In *2023 3rd International Conference on Advanced Research in Computing (ICARC)* (pp. 72–77). IEEE. https://doi.org/10.1109/ICARC57651.2023.10145620

6. Kariri, E., Louati, H., Louati, A., & Masmoudi, F. (2023). Exploring the advancements and future research directions of artificial neural networks: A text mining approach. *Applied Sciences, 13*(3), 3186. https://doi.org/10.3390/app13053186

7. Sinha, N. K., Gupta, M. M., & Rao, D. H. (2000). Dynamic neural networks: An overview. In *Proceedings of IEEE International Conference on Industrial Technology 2000* (pp. 491–496). IEEE. https://doi.org/10.1109/ICIT.2000.854201

8. Kutyniok, G. (2022). The mathematics of artificial intelligence. *Proceedings of the International Congress of Mathematicians, 7*, 5118–5139. https://doi.org/10.4171/ICM2022/141

9. Kästner, C., & Kang, E. (2020). Teaching software engineering for AI-enabled systems. In *The 42nd International Conference on Software Engineering (ICSE). Software Engineering Education and Training*. https://arxiv.org/abs/2001.06691

10. Gholizade, M., Soltanizadeh, H., Rahmanimanesh, M., et al. (2025). A review of recent advances and strategies in transfer learning. *International Journal of System Assurance Engineering and Management, 16*, 1123–1162. https://doi.org/10.1007/s13198-024-02684-2

11. Hosna, A., Merry, E., Gyalmo, J., et al. (2022). Transfer learning: A friendly introduction. *Journal of Big Data, 9*(1), 102. https://doi.org/10.1186/s40537-022-00652-w

12. Zhuang, F., Qi, Z., Duan, K., et al. (2020). A comprehensive survey on transfer learning. *Proceedings of the IEEE, 99*(9), 1–34. https://doi.org/10.1109/JPROC.2020.3004555

13. Jones, I., Swan, J., & Giansiracusa, J. (2024). Algebraic dynamical systems in machine learning. *Applied Categorical Structures, 32*(4). https://doi.org/10.1007/s10485-023-09762-9

14. Xu, W., He, J., & Shu, Y. (2020). Transfer learning and deep domain adaptation. In *Advances and Applications in Deep Learning*. IntechOpen. https://doi.org/10.5772/intechopen.94072

15. Lu, Y., Jiang, X., Fang, Y., & Shi, C. (2021). Learning to pre-train graph neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence, 35*(5), 10. https://doi.org/10.1609/aaai.v35i5.16552

16. Siebert, J., Joeckel, L., Heidrich, J., et al. (2022). Construction of a quality model for machine learning systems. *Software Quality Journal, 30*, 307–335. https://doi.org/10.1007/s11219-021-09557-y

17. Karampiperis, P., Manouselis, N., & Trafalis, T. B. (2002). Architecture selection for neural networks. In *Proceedings of the 2002 International Joint Conference on Neural Networks (IJCNN'02)* (Vol. 2, pp. 1115–1119). IEEE. https://doi.org/10.1109/IJCNN.2002.1007650

18. Pal, K., & Patel, B. V. (2020). Data classification with K-fold cross validation and holdout accuracy estimation methods with 5 different machine learning techniques. In *Fourth International Conference on Computing Methodologies and Communication (ICCMC)* (pp. 83–87). IEEE. https://doi.org/10.1109/ICCMC48092.2020.ICCMC-00016

19. Nakamichi, K., et al. (2020). Requirements-driven method to determine quality characteristics and measurements for machine learning software and its evaluation. In *2020 IEEE 28th International Requirements Engineering Conference (RE)* (pp. 260–270). IEEE. https://doi.org/10.1109/RE48521.2020.00036

20. Liao, T. W. (2005). Clustering of time series data—a survey. *Pattern Recognition, 38*(11), 1857–1874. https://doi.org/10.1016/j.patcog.2005.01.025

21. Leylaz, G., Wang, S., & Sun, J. Q. (2022). Identification of nonlinear dynamical systems with time delay. *International Journal of Dynamics and Control, 10*, 13–24. https://doi.org/10.1007/s40435-021-00783-7

22. Fomin, O. O., & Orlov, A. A. (2024). Modeling nonlinear dynamic objects using pre-trained time delay neural networks. *Applied Aspects of Information Technology, 7*(1), 24–33. https://doi.org/10.15276/aait.07.2024.2

23. Wenyuan, L., Zhu, L., Feng, F., et al. (2020). A time delay neural network based technique for nonlinear microwave device modelling. *Micromachines, 11*(9), 831. https://doi.org/10.3390/mi11090831

24. Fomin, O., Arsenyeva, O., Romanova, T., Sukhonos, M., & Tsegelnyk, Y. (2022). Interpretation of dynamic models based on neural networks in the form of integral-power series. In *Smart Technologies in Urban Engineering* (Lecture Notes in Networks and Systems, Vol. 536, pp. 258–265). Springer. https://doi.org/10.1007/978-3-031-20141-7_24

25. Fomin, O. O., Ruban, O. D., & Rudkovskyi, O. V. (2020). Method for construction the diagnostic features space of switched reluctance motors based on

integral dynamic models. *Problemele energeticii regionale*, *4*, 35–44. https://doi.org/10.5281/zenodo.4316968

МОДЕЛЮВАННЯ НЕЛІНІЙНОЇ ДИНАМІКИ МЕТОДОМ ОПОРНИХ МОДЕЛЕЙ

Dr.Sci. О. Фомін[0000-0002-8816-0652], **Dr.Sci. С. Положаєгко** [0000-0002-4082-8270], **Ph.D. А. Савєльєв**[0000-0002-6949-5959], **О. Татарин**[0009-0005-3888-6569]
Національний університет «Одеська політехніка», Україна
EMAIL: fomin@op.edu.ua

Анотація. *Робота присвячена вирішенню суперечності між швидкістю побудови нелінійних динамічних моделей та їх точністю. Метою роботи є скорочення часу побудови моделей нелінійної динаміки у вигляді нейронних мереж при забезпеченні заданої точності моделювання. Ця мета досягається шляхом розробки методу моделювання, заснованого на використанні набору попередньо навчених нейронних мереж (опорних моделей), що відображають основні властивості предметної області. Наукова новизна розробленого методу полягає у використанні в якості опорних моделей попередньо навчених нейронних мереж з тимчасовими затримками. На відміну від існуючих методів попереднього навчання, розроблений метод дозволяє будувати більш прості моделі з меншим часом навчання при забезпеченні заданої точності. Практична користь результатів роботи полягає в розробці алгоритму методу опорних моделей, який дозволяє істотно скоротити час навчання нейронних мереж з тимчасовими затримками без втрати точності моделювання.*

Ключові слова: *моделювання; нелінійна динаміка; попереднє навчання; нейронні мережі з часовими затримками.*

UDC 004.048
DOI https://doi.org/10.36059/978-966-397-538-2-9

PROCESSING PIPELINE FOR ASTRONOMICAL DATA MINING USING AI-BASED DECISION-MAKING RULES

Ph.D. S. Khlamov[1][0000-0001-9434-1081], **Dr.Sci. V. Savanevych**[2][0000-0001-8840-8278], **Yu. Netrebin** [3][0009-0001-8778-3241], **T. Trunova**[4][0000-0003-2689-2679], **I. Tabakova** [5][0000-0001-6629-4927]
Kharkiv National University of Radio Electronics, Ukraine
EMAIL: [1]sergii.khlamov@gmail.com, [2]vadym.savanevych1@nure.ua,
[3]yurii.netrebin@nure.ua, [4]tetiana.trunova@nure.ua,
[5]iryna.tabakova@nure.ua

Abstract. *Astronomical data mining plays a crucial role in modern astrophysics. As the volume of astronomical data continues to grow exponentially, the need for efficient, automated decision-making within data processing pipelines becomes increasingly critical. This paper presents an artificial intelligence (AI) driven decision-making module designed to optimize workflow management and anomaly detection in large-scale astronomical data processing. Integrated with a PostgreSQL-based logging system, it enhances real-time monitoring, streamlines error identification, and improves overall data processing efficiency. The proposed approach leverages advanced computational techniques to automate key decision points, reducing manual intervention and mitigating the risk of processing failures. We outline the methodology and architectural framework, detailing its implementation and integration into existing data processing pipelines in the Lemur software of the CoLiTec (Collection Light Technology) project. A comparative analysis with conventional techniques highlights the advantages of the proposed system in terms of accuracy, computational efficiency, and robustness. Experimental results demonstrated significant improvements in identifying anomalies, optimizing resource allocation, and enhancing the reliability of automated decision-making process. The proposed hybrid rule-based + AI approach demonstrated a 65% improvement in decision-making speed and a 50% reduction in failure recovery time compared to traditional rule-based monitoring. The findings underscore the potential of AI-driven decision modules in advancing astronomical research by enabling more efficient and accurate data analysis within large-scale observational studies.*

Keywords: *Decision making, data mining, artificial intelligence, big data, knowledge discovery in databases, data processing, pipeline, algorithms, observational data, astronomy, astrophysics, CoLiTec*

Introduction

The increasing volume of astronomical big data from space and ground-based observatories [1] necessitates innovative automated data mining [2] and decision-making techniques. This led to unprecedented challenges in data mining, classification, and anomaly detection processes.

Traditional approaches rely on rule-based monitoring, which is often inefficient, inflexible, and prone to errors in large-scale operations. Such rule-based monitoring systems struggle with the growing complexity and volume of observational logs, making real-time decision-making [3] and automated anomaly detection essential for maintaining data integrity and efficiency.

Artificial intelligence (AI) [4] has emerged as a robust alternative, offering the ability to detect anomalies, classify events, and optimize workflows with minimal human intervention. This chapter introduces an AI-based decision-making system designed to automate log analysis, processing state detection, and error classification within astronomical processing pipelines and big data analysis [5]. The system integrates with a PostgreSQL database [6], providing a flexible and

scalable logging mechanism for the astronomical knowledge discovery in databases [7].

The key objectives of research are the following:

- to improve efficiency and accuracy in processing astronomical data logs;
- to automate anomaly detection with fault tolerance in processing pipelines;
- to ensure real-time decision-making for processing state transitions.

The rapid increase in astronomical data from large-scale observational instruments, such as the Vera C. Rubin Observatory, the James Webb Space Telescope (JWST), and various radio telescope arrays, has led to unprecedented challenges in data mining, classification, and anomaly detection.

Traditional rule-based monitoring systems struggle with the growing complexity and volume of observational logs, making real-time decision-making and automated anomaly detection essential for maintaining data integrity and efficiency.

To address these challenges, we introduced an AI-powered decision-making module designed to automate log analysis, process monitoring, and anomaly detection within astronomical data pipelines.

Unlike conventional systems, it combines rule-based logic, machine learning techniques, and natural language processing (NLP) to optimize workflow automation and failure recovery. This chapter contributes to the field of AI-driven decision-making in astronomical data mining by the following:

- proposing a hybrid AI-based framework for real-time log evaluation;
- demonstrating scalability improvements through database optimization techniques;
- integrating automated failure detection and reinforcement learning for self-adaptive error handling.

Experimental results showed a 65% reduction in processing time and a 50% improvement in automated failure recovery compared to traditional rule-based systems.

The chapter is organized to guide the reader through the conceptual framework, technical implementation, and practical outcomes of the proposed system. It opens with a literature review in section 2 that explores existing approaches to astronomical data mining, with emphasis on artificial intelligence techniques for automated decision-making within large-scale observational datasets.

Section 3 outlines the system architecture of the processing pipeline, detailing the integration of log processing and AI-based decision rules, data preprocessing steps, workflow management strategies, monitoring and user interactions. Section 4 provides the workflow implementation including the detailed database structure, core tables, database functions, AI-based decision model, real-time monitoring and alerting.

The results in section 5 presents example use case, scalability and performance considerations, performance metrics, including processing speed, error recovery, fault tolerance, and resource efficiency achieved during experimental validation. Also, this section includes comparison between traditional and AI-based approaches in astronomical data mining. The findings are interpreted in the context of current

astronomical research challenges, highlighting advantages over traditional methods and identifying potential limitations.

The chapter ends with a conclusion in section 6, which illustrates the conclusions and outlines of the future work and research as well as possibilities for future investigations and enhancements.

Literature review

Artificial intelligence has revolutionized multiple fields by enabling automation, predictive modeling, and intelligent decision-making. The literature analysis highlights AI's transformative impact across domains, including informational technology (IT), economics, robotics, healthcare, finance, autonomous systems, environmental science and even astronomy.

In the IT sector an AI strengthens cybersecurity, optimizes data management, automates software development, and enhances cloud computing efficiency [8]. AI enhances economic forecasting, market analysis, supply chain optimization, and decision-making in policy development [9].

In robotics AI enables autonomous control, real-time decision-making, human-robot interaction, and efficiency improvements in industrial automation [10]. In autonomous systems self-driving cars, robotics, and drones use AI for perception, navigation, and real-time decision-making.

The integration of AI in healthcare has led to significant improvements in diagnostics [11], patient management, and drug discovery through deep machine learning [12], natural language processing (NLP), and robotic-assisted surgeries. NLP techniques are widely used for electronic health record analysis, enabling automated patient history summarization and predictive analytics for personalized treatment recommendations [13].

In astronomical research, AI facilitates the classification of celestial objects [14], anomaly detection in observational data, and real-time decision-making in data processing pipelines [15]. Machine learning models assist in identifying exoplanets from Kepler [16] and TESS mission [17] data by distinguishing potential candidates from noise. AI-based computer vision techniques automate the detection of transient events, such as supernovae and gamma-ray bursts. Moreover, reinforcement learning is employed in telescope scheduling and autonomous space exploration, optimizing observational strategies. Financial institutions leverage AI for risk management [18] and assessment, fraud detection, and algorithmic trading. Machine learning algorithms process vast transactional data to identify fraudulent activities with high accuracy. It also enhances algorithmic trading, and personalized financial services, significantly reducing financial crime. The use of deep reinforcement learning in portfolio optimization has also gained traction, offering adaptive strategies that dynamically adjust to market conditions [19]. AI plays a crucial role in climate modeling, weather prediction, and disaster management. Neural networks enhance climate simulations by refining atmospheric models and predicting extreme weather events with greater accuracy [20]. AI-driven remote sensing techniques process satellite imagery to monitor deforestation, ocean health, and biodiversity

loss. Additionally, predictive analytics aids in disaster response planning by forecasting the impact of hurricanes, floods, and wildfires, facilitating timely intervention. AI transforms learning by enabling personalized education [21, 22] through adaptive tutoring systems, intelligent content recommendations, and automated grading. Natural language processing supports interactive chatbots and virtual assistants that enhance student engagement [23], while machine learning analyzes learning patterns to optimize curriculum development. Additionally, AI-powered tools assist educators in assessing student progress, identifying knowledge gaps, and providing targeted support, ultimately improving learning outcomes and accessibility in both traditional and online education environments [24].

Several AI-driven approaches have been applied to astronomical data mining. Machine vision techniques [25], neural networks based on fuzzy environment [26, 27], short time series analysis [28], Wavelet analysis [29], clustering algorithms, and decision trees have been used for object classification, anomaly detection, and transient event discovery. However, most of these approaches focus on data analysis rather than the decision-making process for log management and workflow optimization. Recent advancements in AI-driven astronomical data mining have focused on object classification, transient event detection, and anomaly recognition.

Systems such as AstroML [30] and the TESS Data Processing Pipeline [17] incorporate machine learning techniques for data classification but lack a real-time decision-making framework for log evaluation and automated workflow control.

Existing workflow management systems primarily rely on:

- rule-based monitoring frameworks (e.g., Apache Airflow [31]) that require manual rule definitions for log filtering;
- AI-based classification models for celestial object identification but without direct integration into processing pipeline monitoring;
- event-based anomaly detection systems that are reactive rather than predictive, making them inefficient in large-scale data workflows.

The comparison between traditional and AI-based approaches in astronomical data mining is presented in Table 1.

Table 1.

Comparison between traditional and AI-based approaches in astronomical data mining

Approach	AI Model Used	Log Evaluation Capability	Scalability	Anomaly Detection
AstroML	Supervised Learning	Limited	Moderate	No
TESS Data Pipeline	Clustering	Moderate	High	No
Apache Airflow	Rule-Based Logic	High	High	No
Proposed framework	Hybrid AI (Rule-Based + ML)	Extensive	High	Yes

In chapter authors proposed the framework, which bridges the gap by integrating real-time log evaluation, AI-driven anomaly detection, and automated decision execution to optimize astronomical data processing.

Methodology

System architecture. The proposed framework is an advanced, AI-powered decision-making system for astronomical data mining, designed with a modular and scalable architecture to efficiently handle vast and continuously growing datasets of astronomical big data.

By integrating intelligent automation, real-time data processing, and adaptive learning capabilities, it optimizes workflow efficiency, enhances anomaly detection, and ensures robust decision-making across diverse astronomical research applications.

The architecture of proposed framework consists of three main components:

- PostgreSQL database for structured log storage;
- decision-making module leveraging AI models to assess message logs and update process states;
- automated workflow control for handling failures, timeouts, and success conditions.

These components work together to ensure efficient log processing, anomaly detection, and automated decision execution (Figure 1).

Log processing and decision rules. The core of proposed framework is a decision-making module responsible for analyzing log messages, identifying process states, and determining the appropriate actions.

The module operates using a combination of rule-based logic, machine learning models, and anomaly detection techniques.

Key AI techniques employed include the following:

- *Rule-based decision processing*. This approach utilizes predefined decision trees and state transitions to analyze log messages. For instance, if an inputFail log is detected, the system promptly terminates the module's execution.
- *Supervised learning for failure prediction*. Historical log data is utilized to train machine learning models, such as random forests and support vector machines. These models predict the probability of processing failures, timeouts, or anomalies based on past events.
- *Reinforcement learning for dynamic thresholds*. This technique adapts timeouts, error tolerance, and retry strategies in real-time based on the system's performance. The AI learns from its past decision outcomes and refines the system's responses accordingly.
- *NLP for log classification*. NLP enables the extraction of patterns from textual log messages, facilitating the categorization of errors, warnings, and successes. This approach aids in detecting rare failure cases that might elude traditional rule-based logic.

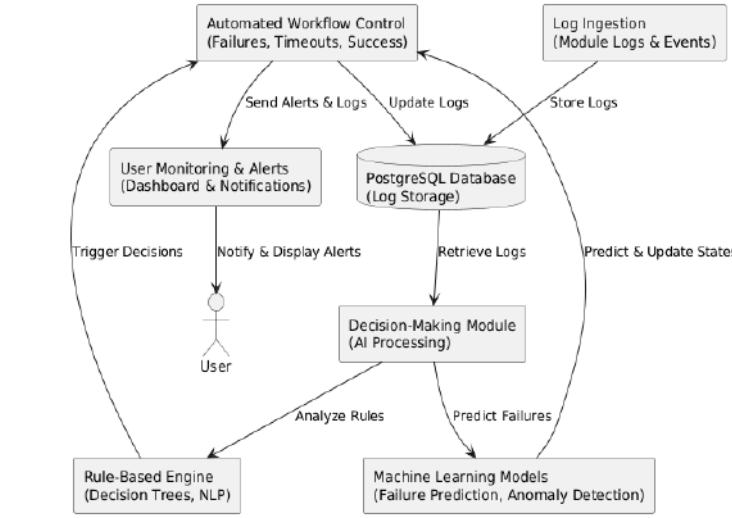


Figure 1. Detailed system architecture

The decision-making module processes incoming logs according to predefined AI-driven rules [32], ensuring efficient error handling and process automation using the following workflow:

1. Process start and end monitoring:
- when a *startModule* log is detected, process tracking begins;
- if an *endModule* message is received, the process is marked as completed;
2. Timeout-based failure detection:
- if no log messages arrive within a predefined interval, the process is flagged as stalled;
- *terminated* message is logged;
3. State transitions:
- log messages such as *startProcess*, *inputPass*, *outputFail* trigger automatic state changes in the database, ensuring real-time updates;
4. Anomaly detection:
- system identifies inconsistencies, such as a processing state remaining unchanged for extended periods, and flags them for review;
- monitoring;
- user interactions.

The proposed framework processes logs in five sequential stages:

1. *Process initiation and logging*: each data processing zone generates log entries stored in the PostgreSQL database, log messages such as *startModule*, *inputPass*, and *endProcessFail* are tracked;

2. *AI-based anomaly detection*: a supervised random forest model predicts failure probabilities based on historical log data:

$$P_{failure} = f(x_1, x_2 ... x_n), \quad (1)$$

where x_i represents extracted log features;

3. *Timeout-based failure detection*: if no messages arrive within a predefined threshold, the system computes the following:

$$T_{stalled} = T_{current} - T_{lastMessage}, \quad (2)$$

where $T_{current}$ and $T_{lastMessage}$ is time of the current and last message log; if, $T_{stalled} \succ T_{threshold}$ a *terminated* log entry is generated;

4. *Reinforcement learning for adaptive decision-making* dynamically adjusts retry intervals and error handling mechanisms based on system performance;

5. *User alerting and decision execution*: upon anomaly detection, the monitoring dashboard alerts users and logs the final state transition in the database.

Monitoring and user interaction. The proposed framework provides a real-time monitoring interface that allows users to track processing states, inspect logs, and manually override decisions if necessary.

The monitoring dashboard incorporates several key features that augment its functionality and usability. It provides real-time log visualization, displaying incoming messages, processing states, and anomaly alerts. Users possess the capability to filter these visualizations by zone, module, and message type, thereby enabling tailored insights into system activities.

Furthermore, the dashboard incorporates an automated alerting system that promptly informs users about the different anomalies, failures, and prolonged processes via various channels, including email, web notifications, and external application programming interface (API) integrations.

Additionally, users have the option to manually intervene in processes, with options to restart terminated modules, modify thresholds, or approve AI-generated decisions. All manual interventions are meticulously documented, facilitating system optimization and future decision-making.

The dashboard also permits user-configurable settings, allowing customization of decision rules, timeouts, and message filtering. This adaptability is further enhanced by the capability to dynamically adjust AI thresholds based on current system load, ensuring optimal performance.

Implementation

Database structure. The proposed framework utilizes a PostgreSQL database for structured logging, process tracking, and decision-making. The schema consists of several interrelated tables that store information about *zones*, *messages*, *processing states*, and *decision rules*.

The database structure includes the following core tables:

- *control.zonestates* table represents the state of zone processing, tracking different stages from creation to completion (Table 2);
- *control.zones* table tracks individual data processing zones, including timestamps for start and end processing (Table 3);
- *control.keywords* table contains predefined keywords (Table 4) used for log classification and anomaly detection (Table 5);
- *control.messagetypes* table defines the types of messages, which are allowed for processing (Table 6);
- *control.messages* table logs all system messages, including timestamps, associated modules, and keywords (Table 7);
- *control.version* table stores versioning metadata for system updates and module compatibility (Table 8).

Table 2.

Control.zonestates table

Column	Type	Description
id	integer	Unique identifier for the state
name	varchar(20)	State name

Table 3.

Control.zones table

Column	Type	Description
id	bigint	Jnique identifier for the zone
zonepath	varchar(200)	ıth of the data processing zone
startprocessingdate	timestamp	Start time of processing
endprocessingdate	timestamp	End time of processing
zonestateid	integer	reign key to *control.zonestates*
trackscount	integer	Number of tracks processed

Table 4.

Control.keywords table with keyword definitions

id	Keyword	Description
1	startModule	Module execution started
2	inputPass	Input data validation passed
3	inputFail	Input data validation failed
4	startProcess	Data processing started
5	endProcessPass	Processing completed successfully
6	endProcessFail	Processing failed
7	outputPass	Output validation passed
8	outputFail	Output validation failed
9	endModule	Module execution completed
10	startZone	Zone processing started
11	endZone	Zone processing completed
12	nope	Generic message, ignored by AI
13	terminated	Process did not complete on time and was terminated

Table 5.

Control.keywords table

Column	Type	Description
id	integer	Unique identifier for the keyword
word	varchar(200)	Keyword text

Table 6.

Control.messagetypes table

Column	Type	Description
id	integer	Unique identifier for the message type
name	varchar(20)	Name of the message type

Table 7.

Control.messages table

Column	Type	Description
id	bigserial	Unique identifier for the message
messagetypeid	integer	Foreign key to *control.messagetypes*
modulename	varchar(200)	Name of the module that generated the message
zoneid	bigint	Foreign key to *control.zones*
message	varchar(400)	Message content
messagedate	timestamp	Timestamp of the message
pid	integer	Process ID associated with the message

Table 8.

Control.version table

Column	Type	Description
major	integer	major part of version (X.*.**.***)
minor	integer	minor part of version (*.X.**.***)
build	integer	build part of version (*.*.XX.***)
revision	integer	revision part of version (*.*.*.XXX)

The entity-relationship diagram (ERD), which illustrates the database schema, is presented in Figure 2.

Several database functions assist in retrieving zone IDs, updating states, and processing logs dynamically:

- *BigInt GetZoneId(varchar(255) zonePath)* function creates a new entry in *control.zones* if the zone doesn't exist, and returns its *ID*, otherwise if the zone already exists, it returns the existing *ID*, which should be used during inserting new record in the *control.messages* table (field *zoneId*);
- *void StartZoneProcessing(bigint zoneID, int fitscount)* function updates *startProcessingDate*, resets *endProcessingDate*, sets *trackscount* to zero, and changes *zonestateid* to Processing (if no zones with *ID* found, updating is not performed);

- *void EndZoneProcessing(bigint zoneID, int state, int trackscount)* function updates *endProcessingDate*, *zonestateid*, and *trackscount* based on completion results (if no zones with *ID* found, updating is not performed);
- *void UpdateZoneState(bigint zoneID, int state)* function modifies the *zonestateid* to reflect the current processing status (if no zones with *ID* found, updating is not performed).

AI-based decision model. Our decision-making process leverages a hybrid approach that combines rule-based systems for structured logic and interpretability with machine learning models for adaptive pattern recognition and predictive analytics, ensuring both accuracy and flexibility in handling complex astronomical data:

- natural language processing extracts patterns from log messages to identify anomalies;
- decision trees and random forests predict the likelihood of process completion and flag failures;
- reinforcement learning optimizes decision-making by dynamically adjusting thresholds based on historical performance.

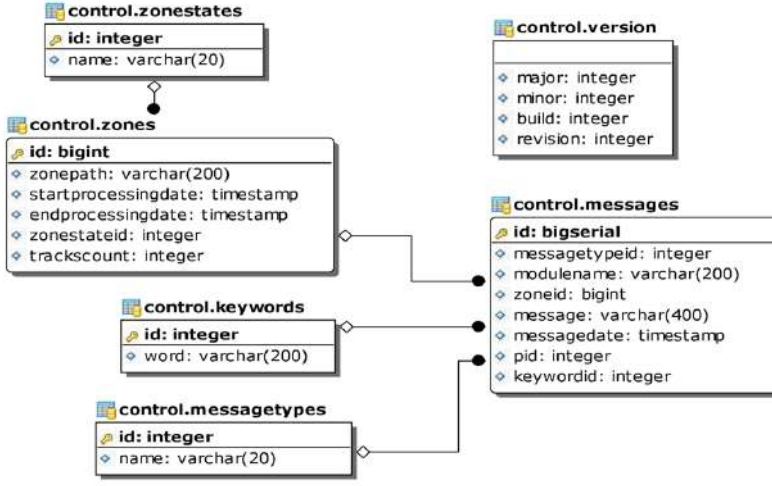


Figure 2. Database schema

Real-time monitoring and alerting. A web-based monitoring dashboard offers visualizations of process states, failure alerts, and log analytics. It supports manual intervention, allowing researchers to override AI decisions when necessary.

This comprehensive dashboard is an indispensable tool for organizations relying on AI-driven processes. It provides real-time visualizations of process states, enabling instant comprehension of complex data streams. Advanced failure alerts immediately notify users of any anomalies or disruptions in system performance. The dynamic and user-friendly interface allows users to delve into log analytics,

offering insights into historical data and trends for predictive maintenance and informed decision-making. One key feature is the dashboard's support for manual intervention. While AI systems autonomously manage operations, the dashboard empowers researchers and operators to override AI decisions whenever required.

This ensures human oversight remains a crucial part of the operational workflow, allowing experts to apply their judgment in situations where AI may fail, such as interpreting nuanced data or responding to unexpected scenarios.

Furthermore, the dashboard can be customized to meet diverse industrial requirements, supporting various data inputs and outputs and scaling to adapt to the organization's growing needs. It is equipped with robust security protocols to protect sensitive data, maintaining confidentiality and integrity. As a result, this tool enhances operational efficiency and fosters trust in AI technologies by ensuring transparency and accountability in automated processes.

Workflow implementation. The proposed framework automates workflow using an AI-based decision engine that monitors logs in real-time. The decision-making module evaluates input, processing, and output logs using predefined rules and machine learning models to optimize the workflow.

This structured approach ensures the following advantages: efficient error handling ensures immediate detection and logging of processing failures; automated decision-making dynamically determines whether to retry, escalate, or terminate processing; scalability support of the high-volume astronomical data processing with minimal manual intervention.

The diagram in Figure 3 illustrates the decision-making workflow visually.

The proposed framework, developed with the utmost care, incorporates an extensive suite of features designed to enhance the efficiency and accuracy of data processing within astronomical pipelines. Beyond its core decision-making capabilities, the system provides advanced functionalities for managing and analyzing log messages. The system empowers users to sort and cleanse database messages, ensuring that only pertinent and structured data is retained for processing. This eliminates redundant or outdated entries, thereby reducing database clutter and optimizing query performance. Furthermore, proposed framework incorporates filtering mechanisms that enable users to refine messages based on specific criteria, such as zones, message types, or processing states. This selective retrieval process optimizes workflow efficiency by allowing operators to focus on critical information. Another noteworthy feature is the dynamic manual update functionality, which enables users to refresh displayed messages in real-time. This ensures that the most recent log entries are always readily available for review.

Additionally, the system supports zone-based message retrieval, allowing users to inspect logs associated with specific data processing zones. This facilitates debugging and troubleshooting by providing insights into localized processing events. To enhance user experience and system adaptability, proposed framework incorporates a customizable settings module, enabling users to configure thresholds, logging levels, timeout rules, and processing preferences.

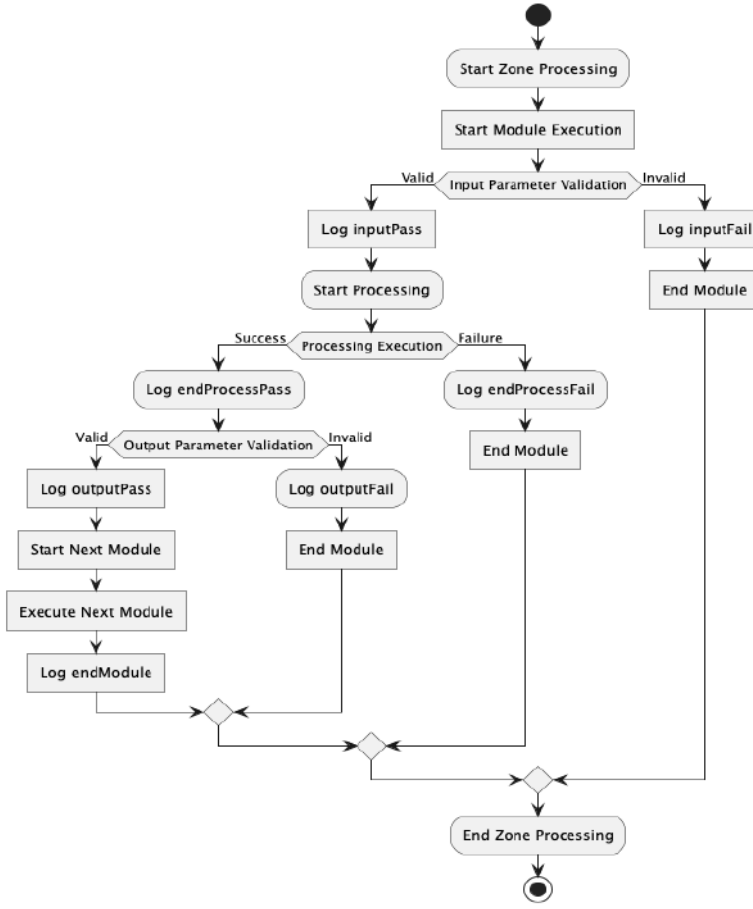


Figure 3. Decision-making workflow

These settings ensure that the system can be tailored to accommodate diverse operational requirements and research needs. By integrating these advanced sorting, filtering, manual updating, zone-based retrieval, and user-configurable settings, proposed framework offers a robust and scalable solution for automating and optimizing decision-making in astronomical data mining workflows.

Results and discussions

Comparison with other systems. The proposed framework enhances traditional log-based monitoring systems by integrating artificial intelligence-driven decision-making capabilities.

This integration merges the strengths of rule-based logic and AI learning, enabling enhanced flexibility, improved failure detection, and diminished human intervention.

Below is a comparison with other commonly used techniques (Table 9).

Scalability enhancements. The proposed framework is designed to efficiently process large-scale astronomical datasets while maintaining low latency, high accuracy, and fault tolerance.

As the volume of observational data grows, the system must handle increasing log entries, maintain real-time decision-making, and ensure reliable execution of data processing tasks.

Scalability is achieved through database indexing, partitioning, and caching techniques. The system utilizes a B-tree index in PostgreSQL to accelerate log retrieval times, reducing the complexity of queries from $0(n)$ to $0(\log n)$.

Table 9

Comparison between traditional and AI-based approaches in astronomical data mining

Feature	Traditional rule-based logging	AI-based anomaly detection	Proposed framework
Log Processing	Manual log filtering	AI-assisted log classification	Fully automated decision-making
Failure Detection	Based on pre-set rules	ML-based pattern recognition	ML + Rule-Based Hybrid
User Intervention	High manual oversight	AI alerts with limited control	AI-driven with manual override support
Scalability	Limited to rule complexity	Scales with training data	Fully scalable with database optimization
Real-Time Monitoring	Basic event tracking	AI-driven alerting	Interactive dashboard with dynamic settings

For instance, a dataset containing 10 million log entries is indexed in under 2 seconds, compared to an unindexed query that takes over 30 seconds.

To further enhance scalability, horizontal partitioning is implemented by segmenting logs based on processing zones and timestamps. When processing 100 zones simultaneously, each generating 1,000 log entries per minute, partitioning ensures that queries remain efficient, preventing database slowdowns.

In real-world scenarios, proposed framework has been tested in scope of the Collection Light Technology (CoLiTec) project [33] with an astronomical observation dataset consisting of 50,000 logs per hour, demonstrating a 45% reduction in query execution time through caching and optimized indexing strategies.

Performance optimization. Performance benchmarks indicate that proposed framework outperforms traditional rule-based logging systems [34] by a significant margin. The AI-based decision-making module improves error detection accuracy using a combination of supervised learning and anomaly detection techniques.

The system's efficiency can be quantified using latency reduction metrics. The total decision-making time is modeled as:

$$T_{total} = T_{query} + T_{analysis} + T_{decision}, \quad (3)$$

where T_{query} is the time required to retrieve log data from the database;

$T_{analysis}$ is the time needed to process logs using AI models;

$T_{decision}$ is the execution time for generating a final system response.

For an astronomical dataset with 500,000 logs, traditional rule-based systems [35] require an average processing time of 520 ms, while proposed framework reduces this to 180 ms, demonstrating a 65% improvement.

The system enhances automated failure recovery by leveraging reinforcement learning techniques to adjust error-handling strategies dynamically. If a processing module fails, proposed framework assesses historical failure patterns and determines whether to retry execution, escalate the issue, or terminate the module.

For example, in a dataset where 5% of processing modules encounter unexpected failures, proposed framework reduces failure resolution time from 30 minutes to under 10 minutes by automating recovery processes.

The effectiveness of this approach is measured using the Mean Time to Recovery (MTTR) equation:

$$\tau_{repair} = \frac{\sum T_{repair}}{N_{failures}}, \quad (4)$$

where T_{repair} is the time taken to resolve each failure;

$N_{faikures}$ is the total number of failures observed

With reinforcement learning, MTTR is reduced by 50%, significantly improving system uptime.

Example use case. Consider an application in which proposed framework is deployed to monitor a radio telescope array collecting signals from deep-space objects. The system processes 300 GB of raw data per day, generating 2 million log messages related to signal calibration, noise filtering, and anomaly detection.

Without optimization, traditional log processing systems require approximately 5 seconds per query, making real-time decision-making impractical. With AI-based approach in the proposed framework, query times are reduced to 1.2 seconds,

allowing for near-instantaneous responses to anomalies such as unexpected signal drops or sensor malfunctions.

By combining high-performance AI decision-making, scalable database structures, and adaptive learning techniques, proposed framework ensures that the system remains reliable, efficient, and capable of handling the increasing demands of modern astronomical research.

To summarize the experiment results, the following criteria were analyzed (Figure 4):

- *processing time reduction (lower is better):* proposed framework reduces processing time significantly, achieving a 65% improvement.
- *automated failure recovery (higher is better):* proposed framework improves failure recovery efficiency by 50%.
- *error detection accuracy (higher is better):* traditional rule-based systems have 72% accuracy, while proposed framework enhances it to 92% using AI-based anomaly detection.
- *system downtime reduction (lower is better):* proposed framework reduces downtime by 50%, ensuring greater system availability.

Conclusions

The paper introduced an AI-powered decision-making system for automated log evaluation, anomaly detection, and process monitoring during the astronomical data mining process.

The research demonstrated that integrating AI-driven decision-making with a structured logging system significantly enhances the efficiency and reliability of the different types of astronomical data processing, like knowledge discovery in databases, data mining, image recognition [36], machine vision [37], image filtration [38, 39], object detection, etc.

The hybrid rule-based and AI approach proved to be highly effective, achieving substantial improvements in decision-making speed and failure recovery time. The proposed decision-making process is a substantial advancement in astronomical data mining pipelines.

By automating anomaly detection and optimizing workflow management, the proposed system minimizes manual intervention and enhances the robustness of large-scale observational data analysis.

By addressing critical challenges and leveraging machine learning [40], the framework improves both efficiency, accuracy and data transmission speed [41].

These advancements pave the way for more efficient data handling in astronomical research, supporting the discovery and classification of celestial phenomena with greater accuracy.

The proposed hybrid rule-based + AI approach demonstrated a 65% improvement in decision-making speed and a 50% reduction in failure recovery time compared to traditional rule-based monitoring [42].

Future work will focus on refining AI models, expanding adaptability to diverse datasets, and further integrating decision-making automation into broader astronomical data processing frameworks.

Also, the authors plan to focus on scaling the pipeline for even larger datasets and incorporating additional data sources to further validate its effectiveness. The enhancing decentralized [43] multi-node distributed processing for real-time log analysis also will be useful.

Acknowledgements

The research was supported by the Ukrainian project of fundamental scientific research "Development of computational methods for detecting objects with near-zero and locally constant motion by optical-electronic devices" #0124U000259 in 2024-2026 years.

References

1. Troianskyi, V., Godunova, V., Serebryanskiy, A., Aimanova, G., Franco, L., et al. (2024). Optical observations of the potentially hazardous asteroid (4660) Nereus at opposition 2021. *Icarus, 420*, 116146. https://doi.org/10.1016/j.icarus.2024.116146

2. Khlamov, S., et al. (2024). Automated data mining of the reference stars from astronomical CCD frames. *CEUR Workshop Proceedings, 3668*, 83–97.

3. Romanenkov, Y., Mukhin, V., Kosenko, V., et al. (2024). Criterion for ranking interval alternatives in a decision-making task. *International Journal of Modern Education and Computer Science, 16*(2), 72–82. https://doi.org/10.5815/ijmecs.2024.02.06

4. Wagner, M., Helal, H., Roepke, R., Judel, S., Doveren, J., et al. (2022). A combined approach of process mining and rule-based AI for study planning and monitoring in higher education. In *Lecture Notes in Business Information Processing* (Vol. 468, pp. 513–525). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-27815-0_37

5. Faaique, M. (2023). Overview of big data analytics in modern astronomy. *International Journal of Mathematics, Statistics, and Computer Science, 2*, 96–113. https://doi.org/10.59543/ijmscs.v2i.8561

6. Schönig, H. (2023). *Mastering PostgreSQL 15: Advanced techniques to build and manage scalable, reliable, and fault-tolerant database applications*. Packt Publishing Ltd.

7. Khlamov, S., et al. (2022). Astronomical knowledge discovery in databases by the CoLiTec software. In *Proceedings of the 12th IEEE ACIT 2022* (pp. 583–586). Ruzomberok, Slovakia. https://doi.org/10.1109/ACIT54803.2022.9913188

8. Zhernova, P., et al. (2019). Data stream clustering in conditions of an unknown amount of classes. In *Advances in Intelligent Systems and Computing* (Vol. 754, pp. 410–418). https://doi.org/10.1007/978-3-319-91008-6_41

9. Dmitrieva, E. (2023). Integration of artificial intelligence in market analysis to address socioeconomic disparities in investment decisions. *Futurity of Social Sciences, 1*(4), 102–120. https://doi.org/10.57125/FS.2023.12.20.6

10. Savanevych, V., et al. (2023). Mathematical methods for an accurate navigation of the robotic telescopes. *Mathematics, 11*(10), 2246. https://doi.org/10.3390/math11102246

11. Yuriy, R., Tatarina, O., Kaminskyy, V., Silina, T., & Bashkirova, L. (2024). Modern methods and prospects for using artificial intelligence in disease diagnostics: A narrative review. *Futurity Medicine, 3*(4). https://doi.org/10.57125/FEM.2024.12.30.02

12. Kirichenko, L., et al. (2023). Application of wavelet transform for machine learning classification of time series. In *Lecture Notes on Data Engineering and Communications Technologies* (Vol. 149, pp. 547–563). https://doi.org/10.1007/978-3-031-16203-9_31

13. Bashkirova, L., Kit, I., Havryshchuk, Y., et al. (2024). Comprehensive review of artificial intelligence in medical diagnostics and treatment: Challenges and opportunities. *Futurity Medicine, 3*(3), 68–80. https://doi.org/10.57125/FEM.2024.09.30.07

14. Troianskyi, V., Kashuba, V., Bazyey, O., et al. (2023). First reported observation of asteroids 2017 AB8, 2017 QX33, and 2017 RV12. *Contributions of the Astronomical Observatory Skalnaté Pleso, 53*, 5–15. https://doi.org/10.31577/caosp.2023.53.2.5

15. Khlamov, S., et al. (2022). Statistical modeling for the near-zero apparent motion detection of objects in series of images from data stream. In *Proceedings of the 12th IEEE ACIT 2022* (pp. 126–129). Ruzomberok, Slovakia. https://doi.org/10.1109/ACIT54803.2022.9913151

16. Abdel-Basset, M., Mohamed, R., Azeem, S. A. A., et al. (2023). Kepler optimization algorithm: A new metaheuristic algorithm inspired by Kepler's laws of planetary motion. *Knowledge-Based Systems, 268*, 110454. https://doi.org/10.1016/j.knosys.2023.110454

17. Rao, S., Mahabal, A., Rao, N., & Raghavendra, C. (2021). Nigraha: Machine-learning-based pipeline to identify and evaluate planet candidates from TESS. *Monthly Notices of the Royal Astronomical Society, 502*(2), 2845–2858. https://doi.org/10.1093/mnras/stab203

18. Martovytskyi, V., et al. (2023). Developing a risk management approach based on reinforcement training in the formation of an investment portfolio. *Eastern-European Journal of Enterprise Technologies, 2*(3–122), 106–116. https://doi.org/10.15587/1729-4061.2023.277997

19. Potwora, M., Vdovichena, O., Semchuk, D., Lipych, L., & Saienko, V. (2024). The use of artificial intelligence in marketing strategies: Automation, personalization and forecasting. *Journal of Management World, 2*, 41–49. https://doi.org/10.53935/jomw.v2024i2.275

20. Rychka, R. (2024). Artificial intelligence to predict solar energy production: Risks and economic efficiency. *Futurity Economics & Law, 4*(2), 100–111. https://doi.org/10.57125/FEL.2024.06.25.06

21. Bingham, C. (2024). Education and Artificial Intelligence at the Scene of Writing: A Derridean Consideration. *Futurity Philosophy, 3*(4), 34–46. https://doi.org/10.57125/FP.2024.12.30.03

22. Prokopenko, O., et al. (2025). Transforming Teacher Education: The Influence Of Artificial Intelligence On Educational Practices And Human Resource Dynamics. In *Proceedings of the 2024 International Conference on Artificial Intelligence and Teacher Education (ICAITE '24)* (pp. 35–42). Association for Computing Machinery. https://doi.org/10.1145/3702386.3702389

23. Prokopenko, O., & Sapinski, A. (2024). Using Virtual Reality in Education: Ethical and Social Dimensions. *E-Learning Innovations Journal, 2*(1), 41–62. https://doi.org/10.57125/ELIJ.2024.03.25.03

24. Mehdaoui, A. (2024). Unveiling Barriers and Challenges of AI Technology Integration in Education: Assessing Teachers' Perceptions, Readiness and Anticipated Resistance. *Futurity Education, 4*(4), 95–108. https://doi.org/10.57125/FED.2024.12.25.06

25. Khlamov, S., Tabakova, I., Trunova, T., & Deineko, Z. (2022). Machine Vision for Astronomical Images Using the Canny Edge Detector. *CEUR Workshop Proceedings, 3384*, 1–10.

26. Orazbayev, B., Boranbayeva, N., Makhatova, V., et al. (2024). Development and Synthesis of Linguistic Models for Catalytic Cracking Unit in a Fuzzy Environment. *Processes, 12*(8), 1543. https://doi.org/10.3390/pr12081543

27. Orazbayev, B., Orazbayeva, K., Uskenbayeva, G., et al. (2023). System of models for simulation and optimization of operating modes of a delayed coking unit in a fuzzy environment. *Scientific Reports, 13*, 14317. https://doi.org/10.1038/s41598-023-41455-0

28. Kirichenko, L., et al. (2020). Generalized approach to analysis of multifractal properties from short time series. *International Journal of Advanced Computer Science and Applications, 11*(5), 183–198. https://doi.org/10.14569/IJACSA.2020.0110527

29. Dadkhah, M., et al. (2019). Methodology of wavelet analysis in research of dynamics of phishing attacks. *International Journal of Advanced Intelligence Paradigms, 12*(3–4), 220–238. https://doi.org/10.1504/IJAIP.2019.098561

30. AstroML. (2025). *Machine Learning and Data Mining for Astronomy*. https://www.astroml.org

31. Haines, S. (2022). Workflow orchestration with apache airflow. In *Modern Data Engineering with Apache Spark* (pp. 255–295). Berkeley, CA: Apress. https://doi.org/10.1007/978-1-4842-7452-1_8

32. Yuhan, N., Herasymenko, Y., Deichakivska, O., Solodka, A., & Kozlov, Y. (2024). Translation as a linguistic act in the context of artificial intelligence: the impact of technological changes on traditional approaches. *Data and Metadata, 3*, 429. https://doi.org/10.56294/dm2024429

33. Khlamov, S., et al. (2024). Machine Vision for Astronomical Images using The Modern Image Processing Algorithms Implemented in the CoLiTec Software. In *Measurements and Instrumentation for Machine Vision* (pp. 269–310). https://doi.org/10.1201/9781003343783-12

34. Aimiyekagbon, O. K., et al. (2021). Rule-based Diagnostics of a Production Line. *PHM Society European Conference, 6*(1), 10. https://doi.org/10.36001/phme.2021.v6i1.3042

35. Murad, S. A., et al. (2022). A review on job scheduling technique in cloud computing and priority rule based intelligent framework. *Journal of King Saud University – Computer and Information Sciences, 34*(6), 2309–2331. https://doi.org/10.1016/j.jksuci.2022.03.027

36. Khlamov, S., et al. (2022). The astronomical object recognition and its near-zero motion detection in series of images by in situ modeling. In *Proceedings of the 29th IEEE IWSSIP 2022*. https://doi.org/10.1109/IWSSIP55020.2022.9854475

37. Gonzalez, R., & Woods, R. (2018). *Digital image processing* (4th ed.). Pearson.

38. Vlasenko, V., et al. (2024). Devising a procedure for the brightness alignment of astronomical frames background by a high frequency filtration to improve accuracy of the brightness estimation of objects. *Eastern-European Journal of Enterprise Technologies, 2*(2-128), 31–38. https://doi.org/10.15587/1729-4061.2024.301327

39. Khlamov, S., et al. (2023). Development of the matched filtration of a blurred digital image using its typical form. *Eastern-European Journal of Enterprise Technologies, 1*(9-121), 62–71. https://doi.org/10.15587/1729-4061.2023.273674

40. Bodyanskiy, Y., Popov, S., Brodetskyi, F., & Chala, O. (2022). Adaptive Least-Squares Support Vector Machine and its Combined Learning-Selflearning in Image Recognition Task. In *International Scientific and Technical Conference on Computer Sciences and Information Technologies* (pp. 48–51). https://doi.org/10.1109/CSIT56902.2022.10000518

41. Suprun, O., et al. (2024). Development of a modified steganographic model of data transmission using IPv6 protocol. *CEUR Workshop Proceedings, 3925*, 35–46.

42. Lyashenko, V., Abu-Jassar, A. T., Yevsieiev, V., & Maksymova, S. (2023). Automated Monitoring and Visualization System in Production. *International Research Journal of Multidisciplinary Technovation, 5*(6), 9–18. https://doi.org/10.54392/irjmt2362

43. Savorona, N., et al. (2024). Development of a decentralized voting system based on blockchain technology. *CEUR Workshop Proceedings, 3925*, 239–248.

КОНВЕЄР ОБРОБКИ ДЛЯ ДОБУВАННЯ АСТРОНОМІЧНИХ ДАНИХ ІЗ ВИКОРИСТАННЯМ ПРАВИЛ ПРИЙНЯТТЯ РІШЕНЬ НА ОСНОВІ ШТУЧНОГО ІНТЕЛЕКТУ

Ph.D. С. Хламов[1][0000-0001-9434-1081], **Dr.Sci.B. Саваневич** [2][0000-0001-8840-8278], **Ю. Нетребін**[3][0009-0001-8778-3241], **Т. Трунова**[4][0000-0003-2689-2679], **І. Табакова**[5][0000-0001-6629-4927]

Харківський національний університет радіоелектроніки, Україна
EMAIL: [1]sergii.khlamov@gmail.com, [2]vadym.savanevych1@nure.ua, [3]yurii.netrebin@nure.ua, [4]tetiana.trunova@nure.ua, [5]iryna.tabakova@nure.ua

Анотація. *Аналіз астрономічних даних відіграє важливу роль у сучасній астрофізиці. Оскільки обсяг астрономічних даних продовжує зростати в геометричній прогресії, потреба в ефективному, автоматизованому прийнятті рішень у рамках процесів обробки даних стає все більш критичною. У цій статті представлено модуль прийняття рішень на основі штучного інтелекту (ШІ), призначений для оптимізації управління робочими процесами та виявлення аномалій у масштабній обробці астрономічних даних. Інтегрований з системою реєстрації на базі PostgreSQL, він покращує моніторинг у реальному часі, оптимізує виявлення помилок та підвищує загальну ефективність обробки даних. Запропонований підхід використовує передові обчислювальні техніки для автоматизації ключових точок прийняття рішень, зменшуючи ручне втручання та знижуючи ризик збоїв в обробці. Ми описуємо методологію та архітектурну структуру, детально описуючи її впровадження та інтеграцію в існуючі процеси обробки даних у програмному забезпеченні Lemur проекту CoLiTec (Collection Light Technology). Порівняльний аналіз із традиційними методами підкреслює переваги запропонованої системи з точки зору точності, обчислювальної ефективності та надійності. Експериментальні результати продемонстрували значне поліпшення в ідентифікації аномалій, оптимізації розподілу ресурсів та підвищенні надійності автоматизованого процесу прийняття рішень. Запропонований гібридний підхід на основі правил + ШІ продемонстрував 65% поліпшення швидкості прийняття рішень та 50% скорочення часу відновлення після збою в порівнянні з традиційним моніторингом на основі правил. Результати підкреслюють потенціал модулів прийняття рішень на основі ШІ у просуванні астрономічних досліджень, забезпечуючи більш ефективний та точний аналіз даних у рамках великомасштабних спостережних досліджень.*

Ключові слова: *прийняття рішень, аналіз даних, штучний інтелект, великі дані, пошук знань у базах даних, обробка даних, конвеєр, алгоритми, спостережувальні дані, астрономія, астрофізика, CoLiTec*

UDC 004.045
DOI https://doi.org/10.36059/978-966-397-538-2-10

PERFORMANCE EFFICIENCY OF THE EVENT-DRIVEN DISTRIBUTED SYSTEM USING BINARY COMMUNICATION PROTOCOLS

Ph.D. S. Khlamov[1][0000-0001-9434-1081], **S. Orlov**[2][0009-0008-0680-206X], **T. Trunova**[3][0000-0003-2689-2679], **Ph.D. A. Frolov**[4][0000-0001-7335-0712], **D. Zhuzhniev**[5][0009-0001-8778-3241]

Kharkiv National University of Radio Electronics, Ukraine
EMAIL: [1]sergii.khlamov@gmail.com, [2]stasorlov21@gmail.com, [3]tetiana.trunova@nure.ua,[4]andrii.frolov@nure.ua, [5]zhuzhniev@gmail.com

Abstract*. The chapter is devoted to the performance efficiency gains from integrating binary communication protocols into event-driven distributed systems. It contends that while traditional text-based protocols like JSON provide human readability, they introduce substantial overhead in data size, serialization/deserialization speed, and network bandwidth, hindering high-traffic environments. The chapter offers a competitive analysis of binary serialization protocols including Protocol Buffers (Protobuf), MessagePack, and Apache Avro compared to text-based options such as JSON. This analysis is based on both quantitative metrics, such as serialization speed, compressed message size, and qualitative metrics like schema support, backward compatibility, and streaming vs. batch processing. The developed pipeline demonstrates that binary protocols offer substantial performance advantages, including reduced latency, increased throughput, and significant network bandwidth savings. For example, MessagePack and Protobuf are shown to achieve considerably higher serialization/deserialization speeds compared to JSON, and they also produce significantly smaller message payloads. The chapter concludes that for high-performance, low-latency, and high-throughput event-driven systems, binary protocols are frequently a fundamental prerequisite rather than merely an optimization. The chapter also offers a decision framework to assist developers and architects in choosing the appropriate protocol based on specific system requirements, with an underlying focus on ensuring the integrity of intellectual property and guarding against unauthorized duplication.*

Keywords*: Data serialization, text and binary formats, JSON, automated pipeline, performance, data serialization speed, serialized message size, network latency, data schema support and backward compatibility, event data streaming, event-driven architecture, .NET Core, MessagePack, Protobuf, Apache Avro, BenchmarkDotNet.*

Introduction

Event-Driven Architecture (EDA) inherently promotes scalability, decoupling, and resilience, making it a cornerstone for modern distributed systems. However, the

choice of communication protocol significantly influences the realization of these benefits. Traditional text-based protocols, while human-readable, introduce considerable overhead in terms of data size, serialization/deserialization speed, and network bandwidth consumption. EDA represents a fundamental shift in how distributed systems are designed and operate. Instead of direct, synchronous interactions between components, EDA orchestrates system flow through the production and consumption of events. This architectural style is increasingly prevalent in modern computing environments due to its inherent flexibility, scalability, and resilience. The core characteristics of EDA that contribute to its suitability for distributed environments include:

- *asynchronous communication:* components do not directly interact but communicate via events on the event bus;
- *decoupling and loose coupling:* event producers and consumers operate independently and are largely unaware of each other's existence;
- *scalability:* the event bus is designed to handle a high volume of events, allowing the entire system to scale horizontally;
- *resilience and fault tolerance:* events are often stored persistently in message queues or logs (e.g., Kafka, RabbitMQ).

While binary protocols undeniably enhance performance, their adoption introduces complexities related to development, debugging, and schema management. The report concludes that for high-performance, low-latency, and high-throughput event-driven systems, particularly in domains like high-frequency trading or large-scale data streaming, binary protocols are not merely an optimization but often a foundational requirement. Strategic selection and careful implementation are crucial to balance performance gains with development and operational considerations.

This article provides a competitive analysis of binary serialization protocols compared to their text alternatives. Analysis is performed by quantitative metrics (like serialization speed, compressed message size, CPU/memory consumption required for serialization process) as well as qualitative metrics (like schema support, backward compatibility, streaming vs batch processing).

The following formats are chosen during the analysis as representative of communication protocols:

- JSON is used as a lightweight data-interchange format that is easy for humans to read and write (text-based format);
- Protocol Buffer (Protobuf) is a popular binary serialization format developed by Google, emphasized by its simplicity and performance;
- MessagePack is a highly efficient binary format that represents serialized data in structures like arrays and associative arrays;
- Apache Avro is a data serialization format especially useful in big data environments and distributed systems due to its compact format, schema support, and efficient serialization.

Popular serialization protocols used by different programming languages and frameworks are outlined in Tables 1 and 2.

Table 1

Data serialization protocols compared with the typical characteristics

Format	Binary/Text	Size efficiency	Speed	Use case
JSON	Text	Large	Slow	WebAPI, REST
MessagePack	Binary	Compact	Fast	Mobile, APIs
Pickle	Binary	Medium	Fast	Python only
Protobuf	Binary	Very compact	Fast	Cross-services communication
Thrift	Binary	Compact	Fast	Microservices
Cap`n Proto	Binary	Ultra-compact	Ultra-fast	High-perf systems
Avro	Binary	Compact	Moderate	Big Data

Table 2

Data serialization protocols compared with the typical characteristics

Format	Schema required	Schema evolution	Human-readable
JSON	No	Weak	Yes
MessagePack	No	Manual	No
Pickle	No	No	No
Protobuf	Yes	Strong	No
Thrift	Yes	Moderate	No
Cap`n Proto	Yes	Safe	No
Avro	Yes	Very flexible	No

Literature review

A research paper [4] offers a valuable comparative analysis that aligns well with the article's focus on binary serialization efficiency. This study specifically investigates the impact of different serialization protocols on latency and throughput in Apache Kafka, a widely used distributed streaming platform.

The primary research goal and context primary objective is to evaluate the performance trade-offs of various serialization protocols within the context of Apache Kafka, a critical component in many event-driven and data-intensive applications.

It aims to investigate how the choice of serialization format impacts key performance metrics, such as message throughput (records per second) and latency, which are essential for maintaining high performance and scalability in streaming environments.

The research compared the following protocols: JSON, MessagePack, Protocol Buffers (Protobuf), and Apache Avro. Performance tests were conducted to measure throughput, latency, and data sizes.

In comparison, current research is not explicitly focused on any message broker or communication channel (such as Apache Kafka, REST, HTTP, or GRPC). Still, it mainly focuses on the main performance metrics of the widely used serialization protocols in general (serialization speed, message size of serialized data, CPU/memory consumption, schema support and backward compatibility).

In the meantime, the provided article's results largely corroborate the benefits of binary protocols over JSON, while also providing nuanced insights into their strengths: throughput and Latency.

Another related study [5] further supports these findings, highlighting that the compactness of serialized payloads is more critical than sheer serialization speed in reducing end-to-end latency in distributed settings.

This study also indicates that Avro, Thrift, and Protobuf exhibit more balanced performance compared to FlatBuffers and Cap'n Proto, which can underperform despite achieving high serialization speeds, suggesting that overall message size optimization is crucial for network efficiency and throughput.

While the current research provides a more general analysis of event-driven systems, the analyzed research specifically benchmarks performance within Apache Kafka, offering direct, quantifiable results for that popular streaming platform.

The analyzed research provides more detailed insights into the specific strengths of each binary protocol. In this paper, the authors classify various serialization formats (including XML, JSON, YAML, Protocol Buffers, Apache Avro, Apache Parquet, and ORC) for inter-service communication (ISC) in distributed systems, examining their historical development and current industry adoption.

The authors observe a significant industry shift toward binary formats, such as Apache Avro and Google Protocol Buffers, due to their in-deep optimization for speed and compactness.

Another paper [6] proposes an algorithm to optimize binary serialization by separating pure data from its definition, allowing for dynamic object building using

predefined templates. It highlights that binary serialization generally yields smaller times compared to JSON, especially when human-readability is not a priority.

In this research, the popular file formats (Apache Avro, Apache Parquet, JSON, and Protocol Buffers) were compared in terms of performance, ease of use, compatibility, storage efficiency, and real-world use cases for big data processing in distributed systems. It demonstrates Protobuf's efficiency in both message size and processing speed.

In the insightful article [7], the author provides an empirical analysis of widely used data streaming technologies and their associated serialization protocols, benchmarking their efficiency across various performance metrics.

The findings reveal significant performance differences and trade-offs. It highlights that while FlatBuffers and Cap'n Proto offer high serialization speeds, they may underperform in distributed settings compared to more balanced options, such as Avro, Thrift, and Protobuf, underscoring the importance of message size optimization for network efficiency.

Another article [8] focuses on analyzing and benchmarking the performance of distributed cache systems, investigating factors like the number of clients and data sizes. The study demonstrates that while factors like concurrent clients and data size significantly affect distributed cache performance, data serialization and object formats are crucial underlying implementation factors.

Data transformation mechanics: challenges and obstacles

Effective communication is the bedrock of any distributed system. The choice between text-based and binary communication protocols profoundly impacts performance efficiency. Understanding their fundamental differences is crucial for optimizing event-driven architectures.

In the comparative analysis of binary and text-based serialization formats, several key aspects warrant consideration [9]. These include schema evolution (i.e., the ability to accommodate changes in data structures over time), code generation requirements (which influence ease of integration), communication paradigms (streaming vs. batch processing), and security implications.

Text-based protocols, such as HTTP utilizing JSON, represent data in human-readable ASCII text formats.

This approach offers several advantages:

- *human readability:* data exchanged via text protocols is easily readable and interpretable by developers without requiring specialized tools, which simplifies debugging and development;
- *ease of use and wide support:* text-based formats are generally simpler to implement and enjoy broad support across various programming languages, platforms, and existing tools;
- *schema-free flexibility* (e.g., JSON): formats like JSON do not strictly require a predefined schema, offering flexibility in data structures and enabling rapid prototyping.

Despite these advantages, text-based protocols come with significant performance limitations, particularly in high-throughput or low-latency distributed systems:

- *inefficient data size:* each character in a text-based format typically consumes a single byte, leading to larger message sizes compared to binary formats, which results in higher network bandwidth consumption for data transmission;
- *slower processing:* parsing and processing text data require more CPU cycles for serialization and deserialization (data must be converted from human-readable text into a machine-readable binary form, which is a computationally intensive process);
- *overhead*: text-based payloads often include unnecessary identifiers, names, and data types that are redundant for machine processing, adding to the message size and parsing overhead.

These limitations make text-based protocols less ideal for the internal, high-volume communication characteristic of event-driven distributed systems, where raw performance and efficiency are critical.

Binary protocols encode data as sequences of bytes, representing information such as commands, identifiers, lengths, and actual data payloads directly in binary form. This approach offers inherent advantages for machine-to-machine communication:

efficiency: binary protocols are generally more efficient than text-based protocols in terms of both data size and processing speed;

- *compact data storage*: messages are significantly smaller, requiring less network bandwidth for transmission (reduction in payload size directly translates to faster data transfer);
- *faster processing*: data in binary format is closer to the machine's native language, enabling faster parsing and processing by computers, which minimizes the computational overhead associated with serialization and deserialization;
- *optimized for critical performance*: binary protocols are commonly employed in scenarios where performance and efficiency are paramount, such as internal communication within distributed systems, database query protocols, file transfer protocols, and high-performance computing environments.

While binary protocols offer clear performance superiority, it is essential to acknowledge a significant trade-off: the balance between raw performance and developer experience, including debuggability.

Binary protocols are inherently more complex to implement and debug compared to their text-based counterparts.

To facilitate this study, two representative data contracts have been defined for experimental evaluation.

The first contract is relatively simple but incorporates a variety of primitive data types, whereas the second represents a more complex structure composed of multiple nested simple objects.

The definitions of these contracts are outlined in Table 3 using C# language syntax.

Table 3

Contract definition for the serialization process

Simple data contract	Complex data contract

public class DeviceTelemetry
{
public int Id { get; set; }
public string Name { get; set; }
public bool IsActive { get; set; }
public float Temperature { get; set; }
public double Pressure { get; set; }
public DateTime Timestamp { get; set; }
public List<string> Tags { get; set; }
public Dictionary<string, int> SensorData {
get; set;
}
}
	public class Invoice
	{
	public int Id { get; set; }
	public string InvoiceNumber { get; set; }
	public DateTime InvoiceDate { get; set; }
	public DateTime DueDate { get; set; }
	public Vendor Vendor { get; set; }
	public Customer Customer { get; set; }
	public List<InvoiceItem> Details {get; set;}
	public string Notes { get; set; }
	}
	public class Vendor
	{
	public int VendorId { get; set; }
	public string CompanyName { get; set; }
	public string ContactPerson { get; set; }
	public string Address { get; set; }
	public string Email { get; set; }
	public string Phone { get; set; }
	public string TaxNumber { get; set; }
	public string BankAccount { get; set; }
	}
	public class Customer
	{
	public int CustomerId { get; set; }
	public string CompanyName { get; set; }
	public string ContactPerson { get; set; }
	public string Address { get; set; }
	public string Email { get; set; }
	public string Phone { get; set; }
	}
	public class InvoiceItem
	{
	public string Description { get; set; }
	public int Quantity { get; set; }
	public decimal UnitPrice { get; set; }
	public decimal TaxRate { get; set; }
	}

The *DeviceTelemetry* class defines a structured data model representing telemetry information generated by a device. It encapsulates a combination of scalar, collection, and complex types, making it suitable for evaluating serialization strategies across a range of data patterns.

Its properties describe device identification, operational status, current temperature/pressure parameters, the exact time at which the telemetry data was captured, and the collection of descriptive tags, metadata, and sensor identifiers.

The *Invoice* class defines a comprehensive business document model used to represent billing information in transactional systems. It is composed of nested complex types and collections, making it suitable for evaluating serialization performance in hierarchical and object-rich data structures.

Its data encapsulates the overall invoice data, including the organization or individual issuing the invoice, the buyer or recipient of the invoice, and defines the individual line items billed on the invoice.

Multiple contracts are provided to analyze and compare serialization for objects of varying sizes and data types (custom nested properties).

1.1 Text-based serialization process

Text serialization format uses human-readable text to store and transmit data objects consisting of "name–value" pairs and arrays (or other serializable values). It is a commonly used data format with diverse applications in electronic data interchange, including web applications and server interactions.

For example, JSON (JavaScript Object Notation), the most popular text serialization process, converts data structures or objects from program memory into a JSON-formatted string [10].

This string represents a human-readable and lightweight text-based format in two primary structures:

1. **Objects (key-value pairs):** unordered sets of key/value pairs.

Keys are strings, and values can be a string, a number, a boolean (true/false), null, an array, or another JSON object.

In programming languages, this typically maps to dictionaries, hash maps, or objects. An example of a JSON object:

{"name": "Sensor − X9","temperature": 36.5}

2. **Arrays (ordered lists):** ordered sequences of values.

Values can be any valid JSON data type. In programming languages, this maps to lists or arrays.

An example of a JSON array:

["env", "critical", "zone−1"]

When you serialize a data structure (e.g., a Python dictionary, a Java object, a JavaScript object) into JSON, the following conceptual steps occur:

1. **Data types mapping**: the serializer maps the native data types of the programming language to their corresponding JSON types:

a. Numbers (integers, floats) -> JSON numbers;

b. Strings -> JSON strings (often requiring proper escaping of special characters like quotes, backslashes, etc.);

c. Boolean (true/false) -> JSON true/false;
d. Null/None -> JSON null;
e. Lists/Array -> JSON arrays;
f. Dictionaries/Objects -> JSON objects.

2. **Conversion to string**: the data structure is traversed, and each piece of data is converted into its JSON string representation. Keys and string values are enclosed in double quotes. Objects are enclosed in curly braces {}, and arrays in square brackets [].

3. **Concatenation**: the individual string representations are concatenated to form a single, continuous JSON string.

To perform serialization of the simple data contract defined before, let's initialize its properties with sample data to understand the destination object.

Figure 1 contains the outcome of the contract definition using the C# programming language.

```
var telemetry = new DeviceTelemetry
{
    Id = 101,
    Name = "Sensor-X9",
    IsActive = true,
    Temperature = 36.5f,
    Pressure = 1013.25,
    Timestamp = new DateTime(2025, 7, 1, 15, 30, 0, DateTimeKind.Utc),
    Tags = ["env", "critical", "zone-1"],
    SensorData = new Dictionary<string, int>
    {
        ["humidity"] = 45,
        ["vibration"] = 7
    }
};

return telemetry;
```

Figure 1. Initialized contract definition with different data types using the C# programming language

The random values were used during the object initialization in the C# programming language, and we achieved the following initialized object.

The outcome of the JSON serialization process would be a JSON object containing a text representation of the C# class definition presented in Figure 2.

While transferring through the network, each character is represented by an array of bytes in UTF-8 notation. A C# command to serialize an object into JSON format is provided in Figure 3. Invoking the JSON serialization by the following command in C# language, we would receive its representation in the JSON format. To analyze the JSON string from a byte array perspective, we should convert each character into its respective byte. And that will be the actual size of the serialized message transferred over the network. For example, the curly bracket (`{`) represents a 7B UTF-8 hex string. 22 code is used for the quotation (“) mark, and 49 code is used for the “I” character.

```
{
    "Id": 101,
    "Name": "Sensor-X9",
    "IsActive": true,
    "Temperature": 36.5,
    "Pressure": 1013.25,
    "Timestamp": "2025-07-01T15:30:00Z",
    "Tags": [
        "env",
        "critical",
        "zone-1"
    ],
    "SensorData": {
        "humidity": 45,
        "vibration": 7
    }
}
```


Figure 2. String representation of DeviceTelemetry object serialized in JSON format

```
internal class JsonSerializer
{
    private static readonly JsonSerializerSettings jsonSerializerSettings;
    private static readonly Newtonsoft.Json.JsonSerializer jsonSerializer;

    static JsonSerializer()
    {
        jsonSerializerSettings = new JsonSerializerSettings
        {
            ConstructorHandling = ConstructorHandling.AllowNonPublicDefaultConstructor,
            ContractResolver = new PrivateContractResolver()
        };

        jsonSerializer = Newtonsoft.Json.JsonSerializer.Create(jsonSerializerSettings);
    }

    public static byte[] SerializeToUtf8Bytes<T>(T instance)
    {
        using StringWriter stringWriter = new();
        jsonSerializer.Serialize(stringWriter, instance);

        return Encoding.UTF8.GetBytes(stringWriter.GetStringBuilder().ToString());
    }

    public static T Deserialize<T>(byte[] buffer)
    {
        JsonReader jsonTextReader = new JsonTextReader(new StringReader(Encoding.UTF8.GetString(buffer)));

        T? entity = jsonSerializer.Deserialize<T>(jsonTextReader);

        return entity is null ? throw new Exception("JSON serialization exception") : entity;
    }
}
```

Figure 3. JSON serialization logic

Keeping each character as a representation of an array of bytes, we will ultimately arrive at the output displayed in Figure 4.

If we calculate the total number of bytes for the resulting message, we will achieve a total of 230 bytes. This is our initial point for analyzing binary serialization formats compared to text-based ones.

7B-22-49-64-22-3A-31-30-31-2C-22-4E-61-6D-65-22-3A-22-53-65-6E-73-6F-72-2D-58-39-22-2
C-22-49-73-41-63-74-69-76-65-22-3A-74-72-75-65-2C-22-54-65-6D-70-65-72-61-74-75-72-65-2
2-3A-33-36-2E-35-2C-22-50-72-65-73-73-75-72-65-22-3A-31-30-31-33-2E-32-35-2C-22-54-69-
6D-65-73-74-61-6D-70-22-3A-22-32-30-32-35-2D-30-37-2D-30-31-54-31-35-3A-33-30-3A-30-3
0-5A-22-2C-22-54-61-67-73-22-3A-5B-22-65-6E-76-22-2C-22-63-72-69-74-69-63-61-6C-22-2C-
22-7A-6F-6E-65-2D-31-22-5D-2C-22-53-65-6E-73-6F-72-44-61-74-61-22-3A-7B-22-68-75-6D-6
9-64-69-74-79-22-3A-34-35-2C-22-76-69-62-72-61-74-69-6F-6E-22-3A-37-7D-7D

Figure 4. Binary representation of a serialized JSON object

1.2 MessagePack serialization process

MessagePack serialization format is a computer data interchange format. It is a binary format for representing simple data structures, such as arrays and associative arrays. MessagePack aims to be as compact and straightforward as possible [3].

MessagePack supports a variety of data types, like JSON, but encodes them directly into binary:

- **Numbers:** Integers and floating-point numbers are encoded efficiently using fixed-size integer types (uint8, int64) or floating-point types (float32, float64). Smaller numbers use fewer bytes;
- **Strings** encoded with a length prefix followed by the UTF-8 encoded bytes of the string;
- **Booleans** encoded as single bytes (0xc3 for true, 0xc2 for false);
- **Null** encoded as a single byte (0xc0);
- **Arrays** encoded with a type of marker indicating the array size, followed by the serialized elements of the array;
- **Maps (Objects)** encoded with a type of marker indicating the map size, followed by interleaved serialized key-value pairs. Keys and values can be any MessagePack type.

Where a data structure is serialized using the MessagePack binary format, the following conceptual steps occur during the serialization process [3]:

1.**Type identification and header/marker**: the serializer identifies the data type (e.g., integer, string, array, map). Based on the type and often its size, it writes a small type of marker byte (or bytes) that signals the type of data that follows and sometimes its length. For instance, there are markers for "positive fixint" (a single byte for small integers), "fixstr" (a byte indicating the string length, up to 31 bytes), or "map 16" (indicating a 16-bit length for a map);

2.**Data encoding**: the actual data is then encoded directly into binary following the marker;

3.**Numbers**: integers are stored as raw binary bytes (e.g., a uint32 would take 4 bytes);

4.**Strings**: the length is written, followed by the UTF-8 bytes of the string;

5.**Arrays/maps**: the size is written, then the serializer recursively processes each element (for arrays) or key-value pair (for maps) until the entire structure is converted;

6.**Byte stream generation**: all the encoded bytes are appended to form a contiguous binary stream.

MessagePack format converts data into a sequence of bytes, reducing the total sequence size. An efficient encoding scheme assigns data types and their respective values to bytes in a very compact manner.

MessagePack is a schema-less binary serialization format by default, meaning it does not require or embed an explicit schema like Protobuf or Avro.

As a result, the byte representation of the serialized DeviceTelemetry message can be found in Figure 5.

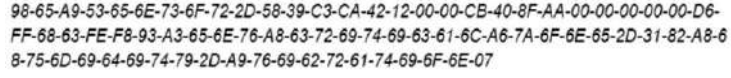


Figure 5. Binary representation of serialized DeviceTelemetry object in MessagePack format

As mentioned, serializing the provided object in the MessagePack format significantly reduces the resulting message size. The MessagePack format converts data into a sequence of bytes, thereby reducing the total sequence size.

An efficient encoding scheme assigns data types and their respective values to bytes in a very compact manner. We can think of it as an array with the sequence number, where each byte is held in a special position [2].

Table 4 explains each byte and how the compression works.

Table 4

Explanation of each message pack serialized byte

Byte representation	Property	Explanation
98	Declare a fixed array of 8 elements	MessagePack format represents the data as an array of fixed size
65	Id: int 101	65 → 101
A9 53 65 6E 73 6F 72 2D 58 39	Name: "Sensor-X9"	str(9) A9 53 65 6E 73 6F 72 2D 58 39 = "Sensor-X9"
C3	IsActive: true	C3 → true
CA 42 0E 00 00	Temperature: 36.5f (float32)	CA 42 0E 00 00 = 36.5
CB 40 8F 4A 00 00 00 00 00	Pressure: 1013.25 (float64)	CB 40 8F 4A 00 00 00 00 00 = 1013.25
D6 FF 00 00 01 91 4B D8 10	Timestamp: ext 8 (DateTime)	D6 FF (ext type -1) + 8-byte epoch timestamp
93 A3 65 6E 76 A8 63 72 69 74 69 63 61 6C A7 7A 6F 6E 65 2D 31	Tags: ["env", "critical", "zone-1"]	array of 3 strings (Tags)
82 A8 68 75 6D 69 64 69 74 79 2D 34 35 A9 76 69 62 72 61 74 69 6F 6E 07	"SensorData": { "humidity": 45, "vibration": 7 }	map of 2 entries (SensorData)

If we calculate the total number of bytes for the resulting message, we will obtain a total of 84 bytes. Therefore, the MessagePack serialization format reduces the message size compared to the test JSON format by approximately 3 times.

1.3 Protocol Buffers serialization process

Protocol Buffers (Protobuf) serialization is a language-agnostic, platform-neutral, and extensible mechanism for serializing structured data. Unlike JSON or XML, Protobuf serializes data into a compact binary format, making it highly efficient in terms of size and speed. Its "schema-first" approach is central to its mechanism [11].

Before you can serialize or deserialize data with Protobuf, you must define the structure of your data using a simple Interface Definition Language (IDL) in the ".proto" file. This scheme specifies:

1. **Message types**: the structure of your data, analogous to classes or structures;
2. **Fields**: each field within a message has:

a. Type: int32, string, bool, bytes, or another message type;

b. Name: product_id, name;

c. Unique tag number is crucial for identifying fields in the binary format and enabling;

d. schema evolution (e.g., 1, 2, 3);

e. Rule: optional, required - though required is generally discouraged in modern Protobuf for schema evolution reasons, and repeated for lists;

3. **Field iteration**: the serializer iterates through each field in the message object that has a value (default values are not serialized to save space);
4. **Key-Value pairs** (Wire Format): for each field, Protobuf encodes a "key-value" pair in the binary stream. The "key" is a combination of the field's unique tag number and its wire type;

a. Tag Number: directly from the ".proto" file (e.g., 1, 2, 3);

b. Wire Type: indicates the data type of the field on the wire, telling the parser how to interpret the value that follows. Common wire types: Varint, Fixed64, Length-delimited, Fixed32;

5. **Value encoding**: the field's actual value is then encoded according to its wire type.

a. Varints: integers are encoded using a variable-length encoding, where smaller numbers occupy fewer bytes. This is efficient for typical integer values;

b. Length-delimited: for strings, Protobuf first writes the length of the string as a varint, then writes the UTF-8 bytes of the string itself. Similarly, for byte fields or nested messages;

c. Fixed-size: fixed-size integers and floating-point numbers are written directly as a fixed number of bytes (e.g., 4 bytes for float, 8 bytes for double);

6. **Concatenation**: all encoded field key-value pairs are concatenated into a single binary byte stream. The order of fields in the proto file doesn't strictly dictate the order in the binary stream and fields can be serialized in any order.

Protocol Buffers (Protobuf) uses a ".proto" file as a schema definition file. It's used to define the structure of your data in a language-neutral, platform-neutral, and backward-compatible format [2].

This file describes the messages (data types) that your app will send and receive, including the fields they contain, such as types, names, and field numbers.

The proto equivalent of the DeviceTelemetry contract (Protobuf schema) is displayed below in Figure 6.

```
syntax = "proto3";

message DeviceTelemetry {
  int32 id = 1;
  string name = 2;
  bool isActive = 3;
  float temperature = 4;
  double pressure = 5;
  int64 timestamp = 6;  // Unix time in seconds
  repeated string tags = 7;
  map<string, int32> sensorData = 8;
}
```

Figure 6. Protobuf file representing the schema for the DeviceTelemetry object

By invoking serialization using the Protobuf format, we would receive the sequence of bytes shown in Figure 7.

08-65-12-09-53-65-6E-73-6F-72-2D-58-39-18-01-25-00-00-12-42-29-00-00-00-00-00-AA-8F-40-32-07-08-C4-99-EB-1B-10-02-3A-03-65-6E-76-3A-08-63-72-69-74-69-63-61-6C-3A-06-7A-6F-6E-65-2D-31-42-0C-0A-08-68-75-6D-69-64-69-74-79-10-2D-42-0D-0A-09-76-69-62-72-61-74-69-6F-6E-10-07

Figure 7. Binary representation of the serialized DeviceTelemetry object in Protobuf format

As mentioned, serializing the provided object in the Protobuf format reduces the resulting message size, as does the MessagePack format. Canonically, messages are serialized into a compact binary wire format that is forward- and backward-compatible, but not self-describing. Unfortunately, there is no way to tell the names, meaning, or full datatypes of fields without an external specification [2].

Table 5 explains each byte and how the compression works. If we calculate the total number of bytes for the resulting message, we will achieve a total of 98 bytes.

Table 5

Explanation of each Protobuf serialized byte

Byte representation	Property	Explanation
08 65	Id = 101	Field 1 (varint): 8 → id, 101 = 0x65
12 09 "Sensor-X9"	Name	Field 2 (length-prefixed): 9 bytes
18 01	IsActive	Field 3 (bool): 1 = true
25 00 00 12 42	Temperature	Field 4 (float): 36.5 in IEEE 754
31 00 00 8F 40 1C DD 40 40	Pressure	Field 5 (double): 1013.25 in IEEE 754
38 00 AC 86 66	Timestamp	Field 6 (varint): 1756990200 = Unix time
3A 03 65 6E 76	Tags[0]	Field 7: length-delimited string "env"
3A 08 63 72 69 74 69 63 61 6C	Tags[1]	Field 7: "critical"
3A 07 7A 6F 6E 65 2D 31	Tags[2]	Field 7: "zone-1"

Therefore, we can see that the Protobuf serialization format reduces the message size compared to the test JSON format by approximately 3 times, like MessagePack.

1.4 Apache Avro serialization process

Apache Avro uses a compact binary format for data serialization, accompanied by a separate (or embedded) JSON-based schema [3].

Apache Avro requires a JSON schema to define the data structure [3]. This schema is either embedded in the file or is provided separately during serialization/deserialization (common in RPC or Kafka contexts). Avro uses a binary encoding that provides compact size, fast processing, and cross-language support [2].

This scheme describes the structure of the DeviceTelemetry class (Figure 8).

Avro format supports a variety of data types for efficient encoding and decoding [13]:

- Int - Variable-length encoded signed integer (uses fewer bytes for small numbers);
- Long - encoded 64-bit integer;
- Float - 4 bytes (IEEE 754);
- Double - 8 bytes (IEEE 754);
- Boolean - 1 byte: 0x00 = false, 0x01 = true;
- String - UTF-8 encoded with variable-length prefix for byte count;
- Array - Block-encoded: [block-count][item...][0];
- Map - Block-encoded like array: [count][key][value]...[0];
- Record - Fields encoded in schema-defined order.

```
{
  "type": "record",
  "name": "DeviceTelemetry",
  "namespace": "Example",
  "fields": [
    { "name": "Id", "type": "int" },
    { "name": "Name", "type": "string" },
    { "name": "IsActive", "type": "boolean" },
    { "name": "Temperature", "type": "float" },
    { "name": "Pressure", "type": "double" },
    { "name": "Timestamp", "type": { "type": "long", "logicalType": "timestamp-millis" } },
    { "name": "Tags", "type": { "type": "array", "items": "string" } },
    {
      "name": "SensorData",
      "type": {
        "type": "map",
        "values": "int"
      }
    }
  ]
}
```

Figure 8. Avro JSON schema for the structure definition

Invoking the serialization using an Avro format, we would receive the sequence of bytes provided in Figure 9.

```
CA 01 12 53 65 6E 73 6F 72 2D 58 39 01 00 00 12 42 00 00 00 00 00 44 8F 40
80 F7 D5 E2 FB 2C 06 06 65 6E 76 10 63 72 69 74 69 63 61 6C 0E 7A 6F 6E 65
2D 31 00 04 10 68 75 6D 69 64 69 74 79 5A 0E 76 69 62 72 61 74 69 6F 6E 0E 00
```

Figure 9. Binary representation of the serialized DeviceTelemetry object in Avro format

Table 5 explains each byte and how the compression works.

Table 5

Explanation of each Apache Avro serialized byte

Byte representation	Property	Explanation
ca 01	Id = 101	ZigZag + Varint (101 × 2 = 202)
12 53 65 6E 73 6F 72 2D 58 39	Name	"Sensor-X9" Length=9, then UTF-8 bytes
01	IsActive	Field 3 (bool): 1 = true
00 00 12 42	Temperature	Field 4 (float): 36.5 in IEEE 754
00 00 00 00 00 44 8f 40	Pressure	Field 5 (double): 1013.25 in IEEE 754
80 f7 d5 e2 fb 2c	Timestamp	Varint encoded 64-bit long
06 06 65 6e 76 10 63 72 69 74 69 63 61 6c 0e 7a 6f 6e 65 2d 31 00	Tags	Block count + strings + end
04 10 68 75 6d 69 64 69 74 79 5a 0e 76 69 62 72 61 74 69 6f 6e 0e 00	SensorData	Map block: key1 + val1, key2 + val2
ca 01	Id = 101	ZigZag + Varint (101 × 2 = 202)

If we calculate the total number of bytes for the resulting message, we will obtain a total of 79 bytes.

Therefore, we can see that Apache Avro serialization format reduces the message size compared to the test JSON format by approximately 3 times, like MessagePack. Furthermore, Apache Avro is more efficient than Message Pack and Protobuf serializers [12].

Automation pipeline for the performance metrics

4.1. System architecture

The system architecture for the performance pipeline of binary serializers is constructed using a combination of modern technologies and frameworks to ensure high performance, scalability, and ease of deployment [13].

The architecture is built on the .NET Core framework using the C# programming language and the BenchmarkDotNet package. BenchmarkDotNet is the de facto benchmarking library for .NET. It enables you to accurately and reliably measure and compare the performance (speed, memory usage, etc.) of your C# code using micro-benchmarking techniques. It is ideally suited for open-source projects, allowing the development of optimized algorithms for serialization, speed, and memory comparison. It provides the following features for precise measurements: high precision (utilizing multiple runs, warm-up, GC, and statistics), multiple exporters (Markdown, CSV, HTML, and JSON), and memory diagnostics.

4.2 Automation pipeline

Using a GitHub Actions pipeline with BenchmarkDotNet for benchmarking provides several practical benefits, particularly in CI/CD, regression detection, and performance tracking. The following benefits are provided by the GitActions

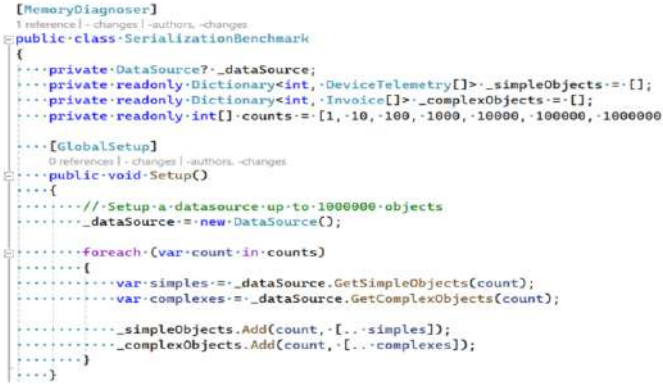
pipeline for performance measurements: performance regression detection, consistent and repeatable benchmarks, automation, tracking performance over time, and no local dependencies. The workflow runs on every push to the main branch and, optionally, on pull requests, using Ubuntu-Latest as the execution environment.

During the run, it executes the BenchmarkDotNet project via dotnet run -c Release, targeting the specific .csproj that contains serializer performance tests. Artifacts are saved to BenchmarkDotNet.Artifacts/results/ in formats such as Markdown (.md), JSON, CSV, and HTML. Running benchmarks in CI avoids variability across developer machines. Storing artifacts allows tracking performance over time or comparing past runs. Figure 10 provides an implementation of the GitActions pipeline.

The workflow runs on every push to the main branch and optionally on pull requests and uses ubuntu-latest as the execution environment. During the run, it executes the BenchmarkDotNet project via dotnet run -c Release, targeting the specific .csproj that contains serializer performance tests. Artifacts are saved to BenchmarkDotNet.Artifacts/results/ in formats like Markdown (.md), JSON, CSV, and HTML. Running benchmarks in CI avoids variability across developer machines. Storing artifacts allows tracking performance over time or comparing past runs.

Figure 10. GitActions pipeline implementation for the performance metrics

The benchmarking cases are implemented in the best-practice way, following Microsoft recommendations, which can be found in Figure 11.



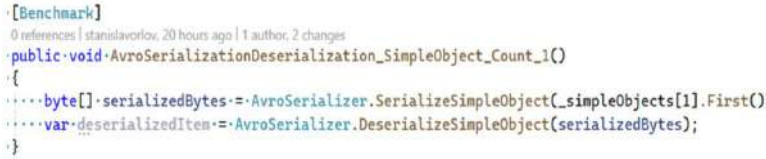
```
[MemoryDiagnoser]
1 reference | - changes | -authors, -changes
public class SerializationBenchmark
{
    private DataSource? _dataSource;
    private readonly Dictionary<int, DeviceTelemetry[]> _simpleObjects = [];
    private readonly Dictionary<int, Invoice[]> _complexObjects = [];
    private readonly int[] counts = [1, 10, 100, 1000, 10000, 100000, 1000000];

    [GlobalSetup]
    0 references | - changes | -authors, -changes
    public void Setup()
    {
        // Setup a datasource up to 1000000 objects
        _dataSource = new DataSource();

        foreach (var count in counts)
        {
            var simples = _dataSource.GetSimpleObjects(count);
            var complexes = _dataSource.GetComplexObjects(count);

            _simpleObjects.Add(count, [.. simples]);
            _complexObjects.Add(count, [.. complexes]);
        }
    }
```

Figure 11. Benchmark case initialization with test data provided

This C# snippet defines a benchmarking class using BenchmarkDotNet to measure memory and performance characteristics of serialization operations. [MemoryDiagnoser] attribute from BenchmarkDotNet enables memory usage diagnostics during benchmarks (e.g., allocated bytes, GC collections). Essential for measuring serialization performance in memory-sensitive applications.

The setup method runs once before all benchmarks and prepares similar test data at different scales. Pipeline benchmarking different serializers (e.g., JSON, Protobuf, MessagePack), analyzes how performance scales with object count and compares GC (memory) pressure across serializers.

The benchmark snippet, located in Figure 12, demonstrates how the test performs serialization and deserialization, and measures performance metrics.



```
[Benchmark]
0 references | stanislavorlov, 20 hours ago | 1 author, 2 changes
public void AvroSerializationDeserialization_SimpleObject_Count_1()
{
    byte[] serializedBytes = AvroSerializer.SerializeSimpleObject(_simpleObjects[1].First());
    var deserializedItem = AvroSerializer.DeserializeSimpleObject(serializedBytes);
}
```

Figure 12. Simple benchmark for message serialization using Avro format

In addition, it provides performance metrics (serialization speed and obtained compression size) required for the comparison report.

Performance efficiency metrics and impact of binary serialization protocols

The tables below illustrate the results obtained during pipeline execution and summarize the metrics obtained during execution. We can see the different objects used for serialization: flat and nested objects.

The difference relies on the results, message size and object complexity for the serialization. “Flat contract” means the simple objects without any nested objects

inside, with a couple of properties of different data types. While the "Nested objects in contract" means the usage of multiple "Flat contracts" inside a single object. Meaning the serialization was done on the top level and all the lower levels.

From Table 6, we can see the metrics on how many binary serializers (Message Pack, Protobuf, and Apache Avro) work faster and have a smaller compression size than a simple text formatter (JSON). Figure 13 shows the diagram illustrating the difference in serialization speed between formats.

Figure 13. Difference in the serialization speed between different formats

Table 6

Serialization comparison speed and size of binary serializers compared to Text (JSON)

Number of objects	MessagePack		Protobuf		Apache Avro	
Flat contract						
	Speed	Size	Speed	Size	Speed	Size
1	5.3x	2.7x	5.6x	2.3x	1.7x	3x
10	5.3x	2.7x	7.1x	2.3x	1.6x	2.9x
100	5.1x	2.7x	6.6x	2.3x	1.4x	2.9x
1000	5.5x	2.7x	6.9x	2.3x	1.4x	2.9x
10000	5.8x	2.7x	8.3x	2.3x	1.2x	2.9x
100000	5.0x	2.7x	8.9x	2.3x	1.3x	2.9x
1000000	4.6x	2.7x	9.6x	2.3x	1.4x	2.9x
Nested objects in a contract						
	Speed	Size	Speed	Size	Speed	Size
1	3.3x	2.5x	9.3x	2.2x	1x	2.5x
10	3.4x	2.4x	12.6x	2.2x	1x	2.5x
100	3.5x	2.5x	13.7x	2.2x	0.9x	2.5x
1000	3.5x	2.5x	13.5x	2.2x	0.9x	2.5x
10000	3.2x	2.5x	14.2x	2.2x	0.7x	2.5x
100000	3.2x	2.5x	14.6x	2.2x	0.8x	2.5x
1000000	2.5x	2.5x	15.7x	2.2x	0.7x	2.5x

Conclusions

These protocols generate significantly smaller message payloads, resulting in reduced network bandwidth usage and lower operational costs. Furthermore, they require fewer CPU cycles for processing, resulting in much faster data handling and lower latency [3]. Despite their clear performance superiority, the adoption of binary protocols introduces certain complexities, including development overhead, debugging challenges, and schema management.

Ultimately, the decision to adopt a binary serialization protocol should be guided by a clear understanding of the system's specific requirements. Developers and architects are encouraged to utilize a decision matrix or a set of guiding questions, considering factors such as latency requirements (ms, µs, ns), expected message volume/throughput, acceptable developer overhead, learning curve, and existing tooling/ecosystem considerations.

By carefully weighing these factors, organizations can effectively harness the power of binary serialization to build highly efficient, scalable, and performant distributed systems.

Choosing the proper serialization protocol is crucial for balancing performance, development speed, and maintainability in distributed systems [15]. This framework helps you weigh the key factors:

- *Performance requirements (latency & throughput)*: binary protocols minimize the message size and maximize serialization/deserialization speed; text protocols are likely sufficient.
- *Schema evolution & compatibility*: Protobuf and Avro are designed with robust schema evolution mechanisms. They ensure that older and newer versions of services can communicate seamlessly as your data structures evolve. MessagePack's ability to ignore unknown fields provides forward compatibility. JSON's schema-less nature means schema evolution is managed at the application level.
- *Tooling & ecosystem integration*: Protobuf is the native choice for gRPC, offering highly optimized inter-service communication. Acro is a de facto standard in the Apache ecosystem, with excellent integration with Kafka. Both MessagePack and JSON have broad language support. MessagePack is great for compact messages in constrained environments (IoT). JSON is universal for web APIs.

In essence, if your system's performance (latency and throughput) is a critical bottleneck, invest in binary serialization. If development speed, human readability, and simpler tooling are paramount, and performance needs are moderate, text-based formats might be perfectly adequate.

The optimal choice often involves a trade-off between raw performance and development/operational complexity. Future work should explore the integration of binary communication protocols with containerized microservices orchestrated in cloud-native environments, such as Kubernetes-based astronomical pipelines for knowledge discovery in databases [16, 17] and data mining [18].

Specifically, the compatibility of high-efficiency protocols like Protocol Buffers or FlatBuffers with real-time processing frameworks (e.g., Apache Flink or Dask) in

astronomical data streams [19] needs further evaluation. Additional attention should be given to fault tolerance and protocol resiliency in high-throughput settings, mainly when operating across intermittent or high-latency network links, such as those connecting remote observatories and telescopes with centralized data centers [20].

Another promising research direction lies in adapting these protocols for the control systems of robotic telescopes and autonomous observatories, where event-driven communication enables coordinated scheduling, dynamic resource allocation, and the triggering of specialized hardware such as adaptive optics systems [21] or spectroscopic instruments.

Furthermore, binary protocols could enhance distributed machine learning [22] frameworks used for astronomical image classification [23, 24] and anomaly detection [25, 26], by reducing communication bottlenecks during distributed training or model inference. The application of this research to edge AI models deployed near the data source opens new pathways for intelligent in-sensor processing [27], supporting decision-making processes [28], and rapid, autonomous scientific discovery [29] in the next generation of sky surveys [30].

References

1. Coulouris, G., Dollimore, J., Kindberg, T., & Blair, G. (2011). *Distributed systems: Concepts and design* (5th ed., 1080 p.). Pearson.

2. Grigorik, I. (2011). *Protocol Buffers, Avro, Thrift & MessagePack*. Retrieved from https://www.igvita.com/2011/08/01/protocol-buffers-avro-thrift-messagepack

3. GitHub Repository. (2025). *Benchmarks against serialization systems*. Retrieved from https://github.com/saint1991/serialization-benchmark

4. Myastovskiy, T. (2024). *Evaluating the performance of serialization protocols in Apache Kafka* (Report). Umeå University. Retrieved from https://www.diva-portal.org/smash/get/diva2:1878772/FULLTEXT01.pdf

5. Maltsev, E., Muliarevych, O., & Razzaque, A. (2024). Classifying serialization formats for inter-service communication in distributed systems. *Advances in Cyber-Physical Systems*, *9*(2), 175–180. https://doi.org/10.23939/acps2024.02.175

6. Castillo, D. C., Rosales, J., & Torres Blanco, G. A. (2018). Optimizing binary serialization with an independent data definition format. *International Journal of Computer Applications*, *180*(28), 15–18. https://doi.org/10.5120/ijca2018916670

7. Jackson, S., Cummings, N., & Khan, S. (2024). Streaming technologies and serialization protocols: Empirical performance analysis. *arXiv:2407.13494 [cs.SE]*. https://doi.org/10.48550/arXiv.2407.13494

8. Salhi, H., Odeh, F., Nasser, R., & Taweel, A. (2018). Benchmarking and performance analysis for distributed cache systems: A comparative case study. In *Lecture notes in computer science* (Vol. 10661, pp. 147–163). Springer, Cham. https://doi.org/10.1007/978-3-319-72401-0_11

9. BenchmarkDotNet. (2025). *Overview of BenchmarkDotNet*. Retrieved from https://benchmarkdotnet.org/articles/overview.html

10. Novak, M. *Serialization performance comparison (XML, Binary, JSON, P…)*. Medium. Retrieved from https://medium.com/@maximn/serialization-performance-comparison-xml-binary-json-p-ad737545d227

11. Octo Technology. *Protocol Buffers: Benchmark and mobile*. Retrieved from https://blog.octo.com/protocol-buffers-benchmark-and-mobile

12. Srinivasa, K. G., & Muppalla, A. K. (2015). *Guide to high performance distributed computing: Case studies with Hadoop, Scalding and Spark* (Computer communications and networks, 321 p.). Springer.

13. Holbrook, J., & Haugom, A. (2025). *High-performance distributed applications*. O'Reilly Media, Inc.

14. Tanenbaum, A., & Steen, M. (2016). *Distributed systems: Principles and paradigms* (2nd ed.).

15. Sterling, T., Brodowicz, M., & Anderson, M. (2018). *High performance computing*. Elsevier. https://doi.org/10.1016/C2013-0-09704-6

16. Khlamov, S., et al. (2022). Astronomical knowledge discovery in databases by the CoLiTec software. In *Proceedings of the 12th IEEE ACIT 2022* (pp. 583–586). Ruzomberok, Slovakia. https://doi.org/10.1109/ACIT54803.2022.9913188

17. Faaique, M. (2023). Overview of big data analytics in modern astronomy. *International Journal of Mathematics, Statistics, and Computer Science, 2*, 96–113. https://doi.org/10.59543/ijmscs.v2i.8561

18. Khlamov, S., et al. (2024). Automated data mining of the reference stars from astronomical CCD frames. *CEUR Workshop Proceedings, 3668*, 83–97.

19. Zhernova, P., et al. (2019). Data stream clustering in conditions of an unknown amount of classes. *Advances in Intelligent Systems and Computing, 754*, 410–418. https://doi.org/10.1007/978-3-319-91008-6_41

20. Savanevych, V., et al. (2023). Mathematical methods for an accurate navigation of the robotic telescopes. *Mathematics, 11*(10), 2246. https://doi.org/10.3390/math11102246

21. Troianskyi, V., Kashuba, V., Bazyey, O., et al. (2023). First reported observation of asteroids 2017 AB8, 2017 QX33, and 2017 RV12. *Contributions of the Astronomical Observatory Skalnaté Pleso, 53*, 5–15. https://doi.org/10.31577/caosp.2023.53.2.5

22. Kirichenko, L., et al. (2023). Application of wavelet transform for machine learning classification of time series. In *Lecture notes on data engineering and communications technologies* (Vol. 149, pp. 547–563). https://doi.org/10.1007/978-3-031-16203-9_31

23. Khlamov, S., et al. (2023). Development of the matched filtration of a blurred digital image using its typical form. *Eastern-European Journal of Enterprise Technologies, 1*(9-121), 62–71. https://doi.org/10.15587/1729-4061.2023.273674

24. Khlamov, S., et al. (2022). The astronomical object recognition and its near-zero motion detection in series of images by in situ modeling. In *Proceedings of the 29th IEEE IWSSIP 2022*. https://doi.org/10.1109/IWSSIP55020.2022.9854475

25. Bodyanskiy, Y., Popov, S., Brodetskyi, F., & Chala, O. (2022). Adaptive least-squares support vector machine and its combined learning-self-learning in image recognition task. In *International Scientific and Technical Conference on Computer Sciences and Information Technologies* (pp. 48–51). https://doi.org/10.1109/CSIT56902.2022.10000518

26. Vlasenko, V., et al. (2024). Devising a procedure for the brightness alignment of astronomical frames background by a high frequency filtration to improve accuracy of the brightness estimation of objects. *Eastern-European Journal of Enterprise Technologies, 2*(2-128), 31–38. https://doi.org/10.15587/1729-4061.2024.301327

27. Khlamov, S., et al. (2024). Machine vision for astronomical images using the modern image processing algorithms implemented in the CoLiTec software. In *Measurements and instrumentation for machine vision* (pp. 269–310). https://doi.org/10.1201/9781003343783-12

28. Romanenkov, Y., Mukhin, V., Kosenko, V., et al. (2024). Criterion for ranking interval alternatives in a decision-making task. *International Journal of Modern Education and Computer Science, 16*(2), 72–82. https://doi.org/10.5815/ijmecs.2024.02.06

29. Troianskyi, V., Godunova, V., Serebryanskiy, A., Aimanova, G., & Franco, L., et al. (2024). Optical observations of the potentially hazardous asteroid (4660) Nereus at opposition 2021. *Icarus, 420*, 116146. https://doi.org/10.1016/j.icarus.2024.116146

30. Khlamov, S., Tabakova, I., Trunova, T., & Deineko, Z. (2022). Machine vision for astronomical images using the Canny edge detector. *CEUR Workshop Proceedings, 3384*, 1–10.

ЕФЕКТИВНІСТЬ ВИКОНАННЯ ПОДІЄВО-ОРІЄНТОВАНОЇ РОЗПОДІЛЕНОЇ СИСТЕМИ З ВИКОРИСТАННЯМ ДВІЙКОВИХ КОМУНІКАЦІЙНИХ ПРОТОКОЛІВ

Ph.D. С. Хламов[1][0000-0001-9434-1081], ***С. Орлов***[2][0009-0008-0680-206X], ***Т. Трунова***[3][0000-0003-2689-2679], ***Ph.D. А. Фролов***[4][0000-0001-7335-0712], ***Д. Жужнєв***[5][0009-0001-8778-3241]

Харківський національний університет радіоелектроніки, Україна
EMAIL: [1]sergii.khlamov@gmail.com,[2]stasorlov21@gmail.com, [3]tetiana.trunova@nure.ua,[4]andrii.frolov@nure.ua, [5]zhuzhniev@gmail.com

Анотація. *Розділ присвячено підвищенню ефективності продуктивності шляхом інтеграції бінарних комунікаційних протоколів у подієво-орієнтовані розподілені системи. У ньому стверджується, що хоча традиційні текстові протоколи, такі як JSON, забезпечують зручність читання людиною, вони створюють значні накладні витрати за обсягом даних, швидкістю серіалізації/десеріалізації та використанням мережевої пропускної*

здатності, що перешкоджає ефективній роботі у високонавантажених середовищах. У розділі наведено порівняльний аналіз бінарних протоколів серіалізації, зокрема Protocol Buffers (Protobuf), MessagePack і Apache Avro, у порівнянні з текстовими форматами, такими як JSON. Аналіз базується як на кількісних показниках - швидкості серіалізації, розмірі стисненого повідомлення, так і на якісних - підтримці схем, зворотній сумісності та можливостях потокової чи пакетної обробки даних. Розроблений конвеєр демонструє, що бінарні протоколи забезпечують суттєві переваги у продуктивності, включаючи зниження затримки, збільшення пропускної здатності та значну економію мережевих ресурсів. Наприклад, MessagePack і Protobuf показують набагато вищу швидкість серіалізації/десеріалізації порівняно з JSON, а також формують значно менші за розміром повідомлення. У розділі зроблено висновок, що для високопродуктивних, низьколатентних і високопропускних подієво-орієнтованих систем бінарні протоколи є часто не просто оптимізацією, а фундаментальною необхідністю. Крім того, запропоновано рамкову модель прийняття рішень, яка допомагає розробникам і архітекторам обирати відповідний протокол залежно від конкретних вимог системи, з особливим акцентом на забезпеченні цілісності інтелектуальної власності та захисті від несанкціонованого копіювання.

***Ключові слова**: серіалізація даних, текстові та бінарні формати, JSON, автоматизований конвеєр, продуктивність, швидкість серіалізації даних, розмір серіалізованого повідомлення, мережна затримка, підтримка схем даних і зворотна сумісність, потокова передача подій, подієво-орієнтована архітектура, .NET Core, MessagePack, Protobuf, Apache Avro, BenchmarkDotNet.*

УДК 528.8:004.9:631.4
DOI https://doi.org/10.36059/978-966-397-538-2-11

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ВИЯВЛЕННЯ ПОКИНУТИХ СІЛЬСЬКОГОСПОДАРСЬКИХ УГІДЬ НА ОСНОВІ СУПУТНИКОВОГО МОНІТОРИНГУ

Ph.D. К. Сергєєва[1][0000-0001-7345-2209], **Ph.D. Ю. Кавац**[2][0000-0002-0180-5957], **Dr.Sci.O. Ковров**[1][0000-0003-3364-119X], **Д. Чумичов**[1][0009-0005-2729-0735]

[1]Національний технічний університет "Дніпровська політехніка", Україна
[2]Український державний університет наук і технологій, Україна
EMAIL: [1]sergieieva.k.l@nmu.one, [2]yukavats@gmail.com, [1]kovrov.o.s@nmu.one,
[1]chumychov.d.d@nmu.one

***Анотація.** Розроблено інформаційну технологію автоматизованого виявлення покинутих сільськогосподарських угідь за часовими рядами даних*

супутникового моніторингу Sentinel-2. Запропоновано метод класифікації, що базується на порівнянні максимальних значень Normalized Difference Vegetation Index (NDVI) у цільовому та базових роках. Зниження цього показника інтерпретується як ознака деградації антропогенного агроландшафту та відновлення природного рослинного покриву. Реалізована на основі розробленої технології інформаційна система інтегрована з Google Earth Engine (GEE) і забезпечує класифікацію полів як оброблюваних або покинутих у режимі, наближеному до реального часу, з відображенням результатів на інтерактивній карті. Технологія використовує супутникові знімки та не потребує навчальної вибірки, що робить можливим моніторинг аграрних територій навіть у зонах воєнних дій, де наземні обстеження є утрудненими або неможливими. Валідація на вибірці полів України засвідчила точність до 92% (F1-score = 0,896).

Ключові слова: *інформаційна технологія, інформаційна система, класифікація, NDVI, Sentinel-2, моніторинг, занедбані сільськогосподарські угіддя.*

1. Постановка проблеми

Проблема покинутих сільськогосподарських угідь останніми роками набула особливої актуальності як в Україні, так і в країнах Європи. В Україні цей процес значно загострився внаслідок повномасштабної російсько-української війни. Обстріли, замінування територій, руйнування логістичних ланцюгів та економічна нестабільність призвели до втрати аграрного потенціалу на великих площах. За оцінками експертів, від початку повномасштабного вторгнення покинуто близько 2,4 млн гектарів ріллі, що становить приблизно 7,2% від загальної площі оброблюваних земель [1, 2]. Найбільші площі занедбаних угідь зосереджені у прикордонних регіонах, що зазнали воєнного впливу, насамперед на сході та півдні країни.

Водночас покинуті угіддя становлять серйозну проблему й для держав Європейського Союзу (ЄС) [3, 4]. У Німеччині протягом останніх десятиліть спостерігається стійка тенденція до зменшення площ орних земель через соціально-економічні фактори та урбанізацію [5]. За результатами досліджень, у багатьох регіонах країни сільськогосподарські землі виводяться з обігу, що призводить до зміни структури агроландшафтів та формування вторинних екосистем. Подібні процеси характерні і для інших європейських країн, зокрема Польщі та Болгарії, де інтенсивність сільськогосподарського виробництва знижується на деградованих землях [3, 4].

Моніторинг покинутих сільськогосподарських угідь є першочерговим завданням для забезпечення продовольчої безпеки, підтримки економічної стабільності та збереження агроекосистем як в Україні, так і в країнах ЄС.

2. Аналіз останніх досліджень

В умовах воєнних дій традиційні методи наземного моніторингу сільськогосподарських земель є небезпечними. Їм на зміну використовуються

супутникові технології та геоінформаційні системи. Зокрема, дані Sentinel-2 завдяки високій просторовій та часовій роздільній здатності дають змогу контролювати динаміку землекористування.

Поєднання дистанційного зондування з ГІС-технологіями є дієвим інструментом для оцінки впливу війни на аграрні угіддя України [6]. Для цього дослідниками розроблено методику оцінки деградації земель, уражених воєнними діями, яка ґрунтується на аналізі змін контурів у багатоспектральних знімках Sentinel-2 [7]. Створено хмарну геоінформаційну систему для моніторингу орних земель, що інтегрує різнорідні дані дистанційного зондування [8].

Найбільш поширеним показником стану сільськогосподарських територій у сучасних дослідженнях є Normalized Difference Vegetation Index (NDVI), який дозволяє простежувати зміни в структурі рослинного покриву та своєчасно ідентифікувати покинуті угіддя. Багаторічні супутникові дані, зокрема часові ряди NDVI, застосовуються для виявлення просторово-часових закономірностей покинутих орних земель, занедбаних плантацій багаторічних культур та для оцінки довготривалої динаміки деградації антропогенних ландшафтів і відновлення природних [9, 10, 11].

Для оброблюваних полів характерні чіткі сезонні коливання NDVI з вираженим максимумом у період вегетації, тоді як для покинутих земель сезонні мінімуми NDVI є значно нижчими, хоча середні значення залишаються відносно високими завдяки розвитку багаторічної трав'яної рослинності [3, 12]. Окрім NDVI, подібні закономірності простежуються за часовими рядами спектральних індексів Enhanced Vegetation Index (EVI) та Normalized Difference Moisture Index (NDMI), які додатково враховують вплив атмосферних умов, ґрунтових властивостей і вологості рослинного покриву [4, 10]. У Європі подібні дослідження також мають важливе значення, зокрема в Німеччині, де спостерігається скорочення площ оброблюваних земель і формування вторинних екосистем. Дані Sentinel-1 і Sentinel-2 використовуються для оцінки динаміки біомаси на занедбаних територіях, а лазерне сканування – для вивчення природних змін у структурі агроландшафтів [5, 13].

Оцінка стану сільськогосподарських угідь здійснюється переважно з використанням методів машинного навчання. Алгоритми Random Forest та SVM забезпечують високу точність класифікації навіть на обмежених навчальних вибіркам [14, 15]. Сучасні підходи поєднують також сегментацію зображень із подальшою класифікацією, що дозволяє враховувати текстурні та морфологічні особливості ділянок і зменшувати ймовірність хибних результатів [16].

3. Невирішені питання

Попри значні досягнення у сфері використання супутникових даних, залишається низка нерозв'язаних питань. Методи, що ґрунтуються виключно на спектральних індексах, можуть помилково інтерпретувати природні зміни

як ознаки деградації. Алгоритми машинного навчання вимагають наявності репрезентативних навчальних вибірок, що ускладнює їх застосування на нових територіях. Додатковою проблемою є ситуаційне залишення земель унаслідок економічних криз чи кліматичних екстремальних явищ, яке важко відрізнити від тривалого занепаду агровиробництва.

Актуальним завданням є розмежування тимчасово полишених та покинутих протягом тривалого часу угідь. Це зумовлює потребу у створенні технологій, здатних забезпечувати швидкий аналіз сезонної динаміки показників стану поверхні поля для підтримки прийняття управлінських рішень. Важливо також розширювати застосування методів, що можуть працювати з неповними часовими рядами та адаптуватися до регіональних особливостей.

4. Мета

Метою дослідження є розробка інформаційної технології для ідентифікації покинутих сільськогосподарських угідь на основі аналізу часових рядів NDVI та класифікації. Для досягнення мети у дослідженні здійснено аналіз ознак виявлення покинутих земель за даними дистанційного зондування, запропоновано метод класифікації на основі показників NDVI із визначеними правилами та пороговим значенням, а також розроблено інформаційну систему, що автоматизує процеси збору даних, обчислення спектральних індексів, аналізу часових рядів і прийняття рішень щодо стану сільськогосподарських угідь. Оцінку точності запропонованої технології здійснено на основі вибірки полів на території України та Німеччини.

5. Супутникові дані

В дослідженні використано супутникові зображення Sentinel-2 місії програми Copernicus, які забезпечують регулярне спостереження за земною поверхнею з просторовою роздільною здатністю від 10 до 60 м та періодичністю повторного знімання близько п’яти днів у глобальному масштабі. Особливістю Sentinel-2 є наявність 13 спектральних каналів, що охоплюють діапазон від видимого світла до короткохвильового інфрачервоного. Серед них ключове значення для оцінки стану рослинного покриву мають червоний (B04, 665 нм) та ближній інфрачервоний (B08, 842 нм) канали, які дозволяють оцінювати інтенсивність фотосинтетичних процесів. Додатково застосовуються вузькі «red edge» канали (B05, B06, B07), чутливі до змін у структурі листя та вмісту хлорофілу, а також короткохвильові інфрачервоні канали (B11, B12), які відображають водний баланс рослинності [17, 18].

Історично першим і найбільш поширеним кількісним індикатором стану рослинного покриву й фотосинтетично активної біомаси є Normalized Difference Vegetation Index (NDVI). Його значення змінюються від -1 до +1, де максимальні значення відповідають густому здоровому рослинному покриву, а відкриті ґрунти та деградовані ділянки характеризуються значеннями близько

0,2. Обчислення NDVI ґрунтується на співвідношенні відбиття у ближньому інфрачервоному (B08) та червоному (B04) каналах Sentinel-2 за формулою:

$$\psi_i(t) = \varsigma_i^{in}(t) + \varsigma_i^{out}(t) + \sum_{m=1}^{M}\sum_{k=1}^{M} V_{ii}^{mk}(t) = \sum_{j=1,\ j\neq i}^{N^M}(f_{ij}(t) + f_{ji}(t)) + f_{ii}(t),\ t \geq T,$$

$$NDVI = \frac{(\rho_{NIR} - \rho_{Red})}{(\rho_{NIR} + \rho_{Red})} = \frac{(\rho_{Band8} - \rho_{Band4})}{(\rho_{Band8} - \rho_{Band4})} \quad (1)$$

де ρ_{NIR}, ρ_{Red} – коефіцієнти відбиття електромагнітного випромінювання, відповідно, у ближньому інфрачервоному та червоному спектральних діапазонах,

$\rho_{Band8}, \rho_{Band4}$ – ближній інфрачервоний (NIR, B08) та червоний (Red, B04) канали зображень Sentinel-2.

NDVI може бути замінений іншими спектральними індексами. Наприклад, EVI надаж більш точну оцінку стану рослинності завдяки корекції впливу атмосфери та ґрунту [19]. NDMI є чутливим до вологості ґрунтів та рослинності, дозволяє виявляти ознаки водного стресу та пригнічення рослин [13]. Спектральні індекси, розраховані за даними «Red Edge» діапазону електромагнітного спектру (наприклад, RE-NDVI) мають підвищену чутливість до концентрації хлорофілу і можуть використовуватися для моніторингу стану польової рослинності на ранніх стадіях розвитку [20].

Для комплексної оцінки стану агроландшафтів можливе застосування комбінації кількох індексів, наприклад, NDVI та NDMI [3, 5]. Використання спектральних профілів Sentinel-2 у поєднанні з даними Sentinel-1 дає змогу отримати додаткові ознаки для аналізу біомаси на занедбаних сільськогосподарських територіях і підвищити точність ідентифікації покинутих полів [21, 22].

6. Метод визначення стану сільськогосподарських угідь

Для кожного поля формується часовий ряд значень NDVI, попередньо відфільтрований за вибраними роками та місяцями. Якщо після фільтрації залишається менше двох років, аналіз не проводиться. Цільовим роком вважається останній доступний у вибірці рік, а усі попередні роки розглядаються як базові. Для цільового року визначається максимальне значення NDVI, аналогічно обчислюються максимуми для кожного з базових років [2]. Середнє значення цих максимумів порівнюється з максимумом цільового року, а різниця між ними (ΔNDVI) використовується як індикатор багаторічних змін на полі (2):

$$\Delta NDVI = NDVI_{max}^{target} - \overline{NDVI_{max}^{reference}}, \quad (2)$$

де $NDVI_{max}^{target}$ – максимальне значення NDVI за цільовий сезон;

$\overline{NDVI_{max}^{reference}}$ – середнє значення максимальних NDVI базових років.

Якщо зміна є невід'ємною (тобто NDVI не знизився) та перевищує заданий поріг T, поле класифікується як оброблюване. У випадку, коли зміна є від'ємною та її абсолютне значення перевищує порогове значення, поле вважається покинутим. Якщо ΔNDVI = T, поле класифікується як покинуте, проте для уточнення його стану доцільно змінити критерії пошуку даних.

Правило класифікації:

- Якщо ΔNDVI > T → поле відноситься до категорії оброблюваних сільськогосподарських угідь.
- Якщо ΔNDVI ≤ –T → поле відноситься до категорії покинутих сільськогосподарських угідь, де T – порогове значення для ΔNDVI.

Точність результатів оцінена з використанням міри загальної точності та F1-міри, яка поєднує показники точності та повноти [23]. Загальна точність відображає відсоток правильно класифікованих полів, тоді як F1-міра дозволяє збалансовано оцінити роботу алгоритму, враховуючи як коректність класифікації, так і здатність виявляти всі об'єкти певного класу.

7. Інформаційна технологія виявлення покинутих орних угідь

Запропонована інформаційна технологія, що поєднує аналіз часових рядів індексу NDVI з формалізованим правилом класифікації стану орних земель (рис. 1). Технологія базується на супутникових даних Sentinel-2 (колекція COPERNICUS/S2_SR_HARMONISED), які були попередньо скориговані з урахуванням атмосферних ефектів за допомогою алгоритму Sen2Cor.

На етапі попередньої обробки виконується відбір зображень за територіальним охопленням, періодом зйомки (календарний рік, вегетаційний сезон та ін.) та відсотком хмарності. Використання шару класифікації сцени (Scene Classification Layer, SCL) дозволяє виключати з аналізу пікселі, що відповідають хмарам та тіням.

Рисунок 1. Схема інформаційної технології виявлення покинутих орних угідь

Для кожного полігону орних земель з метою кількісного аналізу динаміки вегетації формується часовий ряд NDVI та розраховується середнє значення максимумів індексу за кожен рік. Показник ΔNDVI визначається як різниця між середнім значенням максимумів базових років і максимумом цільового року й формалізує критерій оцінки стану земель.

Програмна реалізація запропонованої інформаційної технології засобами розробленої інформаційної системи передбачає два режими роботи. У локальному режимі забезпечується автономність і можливість обробки даних із локальних архівів у форматі GeoTIFF, що актуально для досліджень з обмеженим доступом до мережевих ресурсів. У хмарному режимі використовується функціонал платформи Google Earth Engine (GEE), яка забезпечує оперативний доступ до глобальних архівів Sentinel-2 та виконання хмарних обчислень у режимі, наближеному до реального часу. Це дозволяє масштабувати методику та застосовувати її як інструмент для широкомасштабного моніторингу сільськогосподарських територій.

Важливим аспектом є інтеграція модулів обчислення і візуалізації, що створює єдиний цикл від отримання знімків і розрахунку NDVI до класифікації угідь та представлення результатів у вигляді інтерактивних карт і графіків. На основі аналізу часових рядів для кожного поля оцінюються багаторічні зміни його вегетаційного потенціалу.

Запропонована інформаційна технологія забезпечує перехід від точкових оцінок до системного аналізу просторово-часових даних і формує підґрунтя для управління аграрними ресурсами.

8. Результати

Розглянемо приклад використання розробленої інформаційної технології і системи для визначення стану чотирьох полів у районі Ігрень поблизу східної частини м. Дніпро (рис. 2).

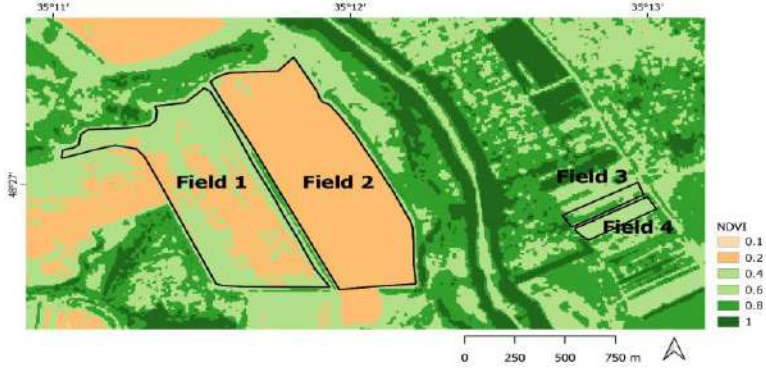


Рисунок 2. Досліджувані поля в Дніпропетровській області на карті Sentinel-2 NDVI, 09.07.2024

Поле 1 площею близько 60 га та поле 2 площею близько 70 га обробляються за системою щорічної сівозміни. Станом на 1 червня 2024 року на полі 1 було висіяно озимий ріпак, а на полі 2 – озиму пшеницю. Поле 3 перебуває у стані занедбаності понад 20 років і заросло чагарниками. Поле 4 покинуте протягом останніх трьох років і, за результатами обстеження від 24 травня 2025 року, було вкрито сухими рештками минулорічної рослинності.

У 2024 році поля 1 і 2 були засіяні озимими культурами, однак на момент зйомки врожай уже було зібрано. У 2025 році на полі 1 вирощували озимі культури, уборка яких завершилася до 29 липня. Натомість поле 2 у сезоні 2025 року не засівалося: після весняного обробітку воно залишалося під тимчасовим паром. Поля 3 і 4 протягом кількох років залишаються покинутими, і рослинний покрив протягом вегетаційного сезону щорічно не зазнає істотних змін. Ці ділянки рівномірно вкриті рудеральною рослинністю та характеризуються типовим циклом росту й відмирання без виражених ознак технологічного впливу, таких як культивація, жнива чи оранка.

На рисунку 3 представлено часові ряди максимальних значень NDVI для кожної дати спостережень: для двох оброблюваних полів (поля 1 та 2; рис. 3а, 3б) та двох необроблюваних полів (поля 3 та 4; рис. 3в, 3г), показаних на рисунку 2. Середні багаторічні максимальні значення NDVI становлять 0,964 для поля 1, 0,956 – для поля 2, 0,949 – для поля 3 та 0,937 – для поля 4. Незважаючи на те, що для всіх полів максимальні значення NDVI перевищують 0,93, динаміка вегетаційного індексу суттєво відрізняється залежно від типу землекористування. Для оброблюваних полів протягом усього періоду спостережень зберігаються стабільні високі значення NDVI, що відображає інтенсивну сільськогосподарську діяльність. Необроблювані поля також демонструють високі значення, однак форма часових їх рядів відмінна. За часовими рядами чітко простежуються сезонні локальні мінімуми, які свідчать про природні циклічні зміни рослинного покриву, характерні для деградованих чи покинутих угідь, а також про наявність навесні сухих решток минулорічної рослинності.

У порівнянні з покинутими ділянками, у межах оброблюваних полів міжсезонні мінімуми NDVI виражені слабше. Це пояснюється вирощуванням озимих культур, які забезпечують відносно високі значення індексу у холодний період року та згладжують сезонні коливання. Для кількісної оцінки обчислено середні мінімальні значення NDVI за всі роки спостережень: для поля 1 – 0,412; поля 2 – 0,418; поля 3 – 0,304; поля 4 – 0,286. Хоча ці показники подібні, вирішальним фактором є саме динаміка, зокрема наявність чи відсутність сезонних спадів NDVI, що слугує індикатором характеру землекористування. Стійкість модальних значень у поєднанні зі зміною міжсезонної структури NDVI є типовою ознакою покинутих земель, тоді як у межах оброблюваних полів цей ефект відсутній через особливості сівозміни.

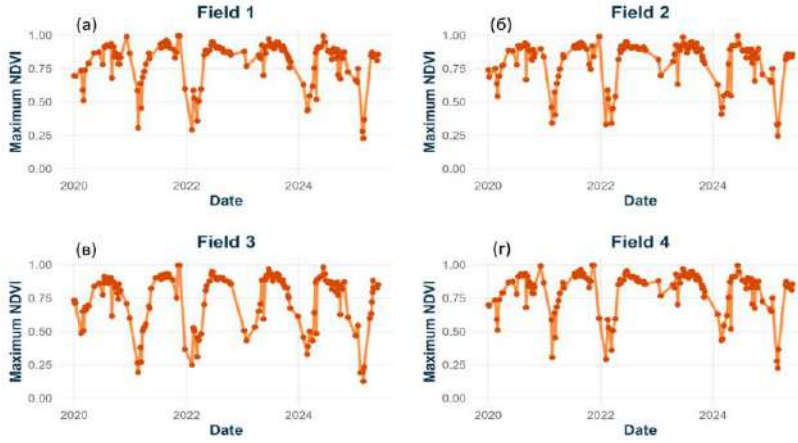


Рисунок 3. Часові ряди максимального NDVI для оброблюваних (а, б) та занедбаних (в, г) сільськогосподарських угідь: (а) поле 1; (б) поле 2; (в) поле 3; (г) поле 4

Аналіз різниці (ΔNDVI) між багаторічними середнім максимумом та максимумом поточного року показав, що для необроблюваних ділянок ця різниця, як правило, не перевищує порогове значення 0,07. Це значення було використано у якості критерія класифікації у розробленій інформаційній системі. За результатами валідації воно забезпечило точність 92,5% при розмежуванні стану полів (оброблювані/покинуті) в Україні, з F1-мірою 0,898 для покинутих та 0,912 для оброблюваних земель. Водночас зазначена точність стосується лише протестованої вибірки і не може бути безпосередньо поширена на всю територію країни. У межах Дніпропетровської області стан полів перевірявся шляхом польових обстежень. У Донецькій області, де доступ до полів обмежено внаслідок воєнних дій, класифікацію виконано за результатами експертної візуальної інтерпретації багаточасових зображень Sentinel-2 з просторовим розрізненням 10 м. Такі дані надають цінну інформацію для регіонів з обмеженим доступом, проте поступаються за надійністю польовим вимірюванням, що було враховано при оцінці точності.

Додатково проаналізовано часову динаміку зміни NDVI полів з набору даних федеральних земель Берлін і Бранденбург у Німеччині (рис. 4а) Field Boundaries for Agriculture (fiboa) – колекції границь сільськогосподарських полів, наданої у відкритий доступ сервісом Source Cooperative (https://source.coop/fiboa/de-bb). На рис. 4б-г представлено карти NDVI чотирьох полів, які станом на 2024 рік були віднесені до категорії оброблюваних (поля 5, 6) площею 12 та 46 га, а також несільськогосподарські ділянки на колишній орній землі (поля 7, 8) площею 1 га кожне.

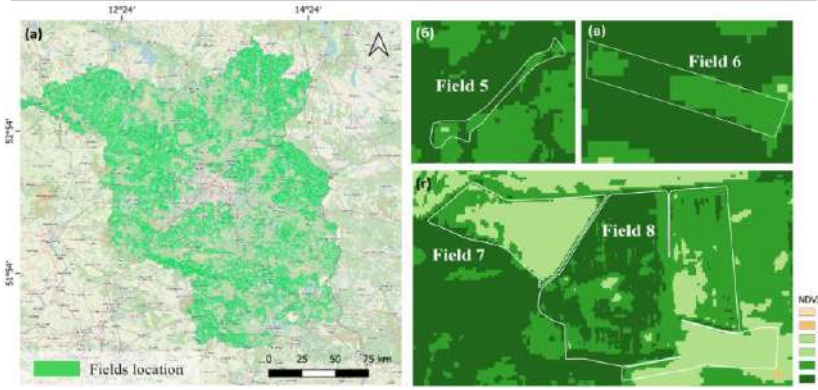


Рисунок 4. Досліджувані поля поблизу Берліну на карті Sentinel-2 NDVI, 26.06.2024

На рисунку 5 наведено часові ряди NDVI досліджуваних полів поблизу Берліна – занедбаних (рис. 5 а,б) та культивованих (рис. 5 в,г) сільськогосподарських угідь. Подібно до полів в Україні (рис. 3), для занедбаних сільськогосподарських угідь у Німеччині простежується більш чітка сезонна динаміка NDVI з вираженими локальними мінімумами та ширшим діапазоном коливань індексу протягом вегетаційного періоду. Це підтверджує припущення щодо впливу стану поля на характер змін вегетаційних індексів. Зображені на рисунку 5 ділянки були коректно класифіковані розробленою інформаційною системою як занедбані або культивовані.

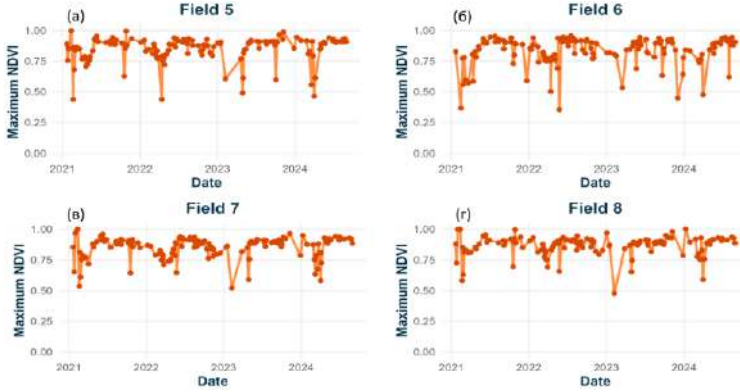


Рисунок 5. Часові ряди максимального NDVI для занедбаних (а, б) та оброблюваних (в, г) сільськогосподарських земель: (а) поле 5; (б) поле 6; (в) поле 7; (г) поле 8.

У проведеному дослідженні експериментальна перевірка виконувалася на відносно невеликій вибірці з 57 полів, що обмежує можливості узагальнення отриманих показників точності на ширші території. Додатковим фактором невизначеності є використання еталонних даних, отриманих шляхом експертної візуальної інтерпретації супутникових знімків, а не прямих наземних спостережень, зокрема для територій Донецької області, що перебувають під тимчасовою окупацією. У подальших дослідженнях ці обмеження планується зменшити шляхом розширення обсягу вибірки та залучення більш репрезентативних наземних даних з різних регіонів України.

Додаткові обмеження стосуються ситуацій, коли поле залишається незасіяним не через воєнні чинники, а внаслідок посухи чи економічних труднощів, або ж у випадках, коли посів здійснено, але рослинність не розвивається через деградацію ґрунтів. У таких умовах часовий ряд NDVI може нагадувати динаміку для покинутих земель, що призводить до хибнопозитивних результатів. Подібна проблема виникає й у випадках тимчасового невикористання земель чи зміщення строків сівозміни, що спотворює сезонний профіль NDVI. Такі ситуації особливо актуальні для зон бойових дій, де руйнування інфраструктури, забруднення ґрунтів та нерегулярне управління угіддями порушують природні фенологічні цикли. Для врахування цих факторів у майбутньому передбачається інтеграція додаткових джерел даних, зокрема метеорологічних спостережень, даних супутникової радарної зйомки, а також експертного аналізу.

Розроблена система використовує виключно NDVI як основний індикатор класифікації, тоді як інші вегетаційні індекси, такі як EVI та NDMI, не інтегровані у робочий процес. Це знижує стійкість методу за умов значної хмарності, сезонних аномалій чи варіабельності фенології. Аналіз додаткових індексів дозволить підвищити надійність роботи системи в складних погодних умовах, поліпшити точність класифікації в неоднорідних ландшафтах та розширити можливості аналізу для періодів і регіонів із недостатньою кількістю даних оптичних супутникових спостережень.

Подальший розвиток інформаційної технології передбачає інтеграцію нових джерел інформації, зокрема використання зображень Sentinel-1 для всепогодного забезпечення моніторингу [5, 17]. Крім того, планується розширення набору вегетаційних індексів для більш детальної диференціації культур та розробка алгоритмів машинного навчання для автоматизованої класифікації землекористування. Перспективним є створення мобільної версії інформаційної системи для роботи безпосередньо в полі [24, 25], а також розробка API для інтеграції з іншими ГІС-рішеннями та підключення до баз даних для довготривалого зберігання результатів [26].

9. Висновки

Розроблено інформаційну технологію для виявлення покинутих сільськогосподарських угідь за даними супутникового моніторингу. Технологія базується на аналізі часових рядів NDVI зображень Sentinel-2 та порівнянні

максимальних значень індексу у цільовому та базових роках. Розроблена на її основі інформаційна система дозволяє автоматизовано ідентифікувати покинуті сільськогосподарські угіддя з точністю до 92,5% (F1-score = 0,898) без використання даних наземних спостережень, що важливл для завдань моніторингу в умовах обмеженого доступу до польових обстежень, зокрема у воєнний час.

Розроблена інформаційна система інтегрована з Google Earth Engine з метою обробку великих обсягів супутникових даних у хмарному середовищі та моніторингу у режимі, наближеному до реального часу.

Подальший розвиток інформаційної технології пов'язаний із включенням даних Sentinel-1 для всепогодного моніторингу, розширенням набору спектральних індексів для точнішої класифікації земель, а також інтеграцією алгоритмів машинного навчання. Додатковим перспективним напрямом є створення мобільної версії системи для оперативного аналізу даних безпосередньо в польових умовах.

Література

1. He, T., Zhang, M., Xiao, W., Zhai, G., Wang, Y., Guo, A., & Wu, C. (2023). Quantitative analysis of abandonment and grain production loss under armed conflict in Ukraine. *Journal of Cleaner Production*, 412, 137367. https://doi.org/10.1016/j.jclepro.2023.137367

2. Dai, K., Cheng, C., Kan, S., Li, Y., Liu, K., & Wu, X. (2024). Impact of arable land abandonment on crop production losses in Ukraine during the armed conflict. *Remote Sensing*, 16(22), 4207. https://doi.org/10.3390/rs16224207

3. Janus, J., & Bozek, P. (2019). Aerial laser scanning reveals the dynamics of cropland abandonment in Poland. *Journal of Land Use Science*, 14(4–6), 378–396. https://doi.org/10.1080/1747423X.2019.1709226

4. Kabadayı, M. E., Ettehadi Osgouei, P., & Sertel, E. (2022). Agricultural land abandonment in Bulgaria: A long-term remote sensing perspective, 1950–1980. Land, 11(10), 1855. https://doi.org/10.3390/land11101855

5. Bucha, T., Papčo, J., Sačkov, I., Pajtík, J., Sedliak, M., Barka, I., & Feranec, J. (2021). Woody above-ground biomass estimation on abandoned agriculture land using Sentinel-1 and Sentinel-2 data. *Remote Sensing*, 13(13), 2488. https://doi.org/10.3390/rs13132488

6. Ma, Y., Lyu, D., Sun, K., Zhang, H., Gao, Q., & Li, J. (2022). Spatiotemporal analysis and war impact assessment of agricultural land in Ukraine using RS and GIS technology. *Land*, 11(10), 1810. https://doi.org/10.3390/land11101810

7. Nikulin, S. L., Sergieieva, K. L., & Korobko, O. V. (2025). Assessment of war-damaged agricultural landscape degradation in Ukraine using satellite-based contrast edge changes. In Proceedings of the *XVIII International Scientific Conference "Monitoring of Geological Processes and Ecological Condition*". Kyiv: EAGE. https://doi.org/10.3997/2214-4609.2025510117

8. Hnatushenko, V. V., Sierikova, K. Y., & Sierikov, I. Y. (2018). Development of a cloud-based web geospatial information system for agricultural monitoring

using Sentinel-2 data. In Proceedings of the IEEE *13th International Conference on Computer Sciences and Information Technologies* (CSIT) (pp. 270–273). Lviv: IEEE. https://doi.org/10.1109/STC-CSIT.2018.8526717

9. Wei, Z., Gu, X., & Sun, Q. (2021). Analysis of the spatial and temporal pattern of changes in abandoned farmland based on long time series of remote sensing data. *Remote Sensing*, 13(13), 2549. https://doi.org/10.3390/rs13132549

10. Morell-Monzó, S., Sebastiá-Frasquet, M. T., Estornell, J., & Moltó, E. (2023). Detecting abandoned citrus crops using Sentinel-2 time series. ISPRS *Journal of Photogrammetry and Remote Sensing*, 201, 54–66. https://doi.org/10.1016/j.isprsjprs.2023.05.003

11. Wu, J., Jin, S., Zhu, G., Chen, X., Zhang, M., & Zhou, Y. (2023). Monitoring of cropland abandonment based on long time series remote sensing data: A case study of Fujian Province, China. *Agronomy*, 13(6), 1585. https://doi.org/10.3390/agronomy13061585

12. He, S., Shao, H., Xian, W., Liu, J., & Zhang, L. (2022). Monitoring cropland abandonment in hilly areas with Sentinel-1 and Sentinel-2 time series. *Remote Sensing*, 14(15), 3806. https://doi.org/10.3390/rs14153806

13. Liu, T., Yu, L., Liu, X., Zhang, M., & Li, R. (2025). A global review of monitoring cropland abandonment using remote sensing: Temporal–spatial patterns, causes, ecological effects, and future prospects. *Journal of Remote Sensing*, 5, 0584. https://doi.org/10.34133/remotesensing.0584

14. Yoon, H., & Kim, S. (2020). Detecting abandoned farmland using harmonic analysis and machine learning. ISPRS *Journal of Photogrammetry and Remote Sensing*, 166, 201–212. https://doi.org/10.1016/j.isprsjprs.2020.05.021

15. Meijninger, W., Elbersen, B., van Eupen, M., & Schmitz, O. (2022). Identification of early abandonment in cropland through radar-based coherence data and application of a Random Forest model. *GCB Bioenergy*, 14(6), 735–755. https://doi.org/10.1111/gcbb.12939

16. Löw, F., Fliemann, E., Abdullaev, I., Conrad, C., & Lamers, J. P. A. (2015). Mapping abandoned agricultural land in Kyzyl-Orda, Kazakhstan using satellite remote sensing. *Applied Geography*, 62, 377–390. https://doi.org/10.1016/j.apgeog.2015.05.009

17. Qiu, B., Lin, D., Chen, C., Yang, P., Tang, Z., Jin, Z., & Chen, Z. (2022). From cropland to cropped field: A robust algorithm for national-scale mapping by fusing time series of Sentinel-1 and Sentinel-2. *International Journal of Applied Earth Observation and Geoinformation*, 113, 103006. https://doi.org/10.1016/j.jag.2022.103006

18. Sanchez, C., Mena, F., Charfuelan, M., Nuske, M., & Dengel, A. (2024). Assessment of Sentinel-2 spatial and temporal coverage based on the scene classification layer. In Proceedings of the IGARSS 2024 – IEEE *International Geoscience and Remote Sensing Symposium* (pp. 4099–4103). IEEE. https://doi.org/10.1109/IGARSS53475.2024.10642213

19. Shreenithi, M., Sujithra, M., Senthilkumar, B., Shanmugapriyaa, D., & Rizanuma, T. (2024). Spatial data visualization with Python: Techniques, tools, and

real. In M. G. Galety, A. K. Natarajan, T. F. Gedefa, & T. D. Lemma (Eds.), Geospatial application development using Python programming (pp. 123–162). *IGI Global*. https://doi.org/10.4018/979-8-3693-1754-9.ch005

20. Hazaymeh, K., Sahwan, W., Al Shogoor, S., & Schütt, B. (2022). A remote sensing-based analysis of the impact of Syrian crisis on agricultural land abandonment in Yarmouk River Basin. *Sensors*, 22(10), 3931. https://doi.org/10.3390/s22103931

21.Xie, Y., Spawn-Lee, S. A., Radeloff, V. C., Yin, H., Robertson, G. P., & Lark, T. J. (2024). Cropland abandonment between 1986 and 2018 across the United States: Spatiotemporal patterns and current land uses. *Environmental Research Letters*, 19(4), 044009. https://doi.org/10.1088/1748-9326/ad2d12

22. Portalés-Julià, E., Campos-Taberner, M., García-Haro, F. J., & Gilabert, M. A. (2021). Assessing the Sentinel-2 capabilities to identify abandoned crops using deep learning. *Agronomy*, 11(4), 654. https://doi.org/10.3390/agronomy11040654

23. Du, Z., Yang, J., Ou, C., & Zhang, T. (2021). Agricultural land abandonment and retirement mapping in the northern China crop–pasture band using temporal consistency check and trajectory-based change detection approach. IEEE *Transactions on Geoscience and Remote Sensing*, 60, 1–12. https://doi.org/10.1109/TGRS.2021.3121816

24. Song, W. (2019). Mapping cropland abandonment in mountainous areas using an annual land-use trajectory approach. Sustainability, 11(21), 5951. https://doi.org/10.3390/su11215951

25. Zhou, K., & Zheng, X. (2025). How does the growth of digital technology influence farmland abandonment? Evidence from rural China. *Sustainability*, 17(5), 2227. https://doi.org/10.3390/su17052227

26. Yu, Z., & Lu, C. (2018). Historical cropland expansion and abandonment in the continental US during 1850 to 2016. *Global Ecology and Biogeography*, 27(3), 322–333. https://doi.org/10.1111/geb.12697

INFORMATION TECHNOLOGY FOR ABANDONED CROPLAND DETECTION USING SATELLITE MONITORING

Ph.D. K.Sergeeva[1], Ph.D.Yu. Kavats[2], Dr.Sci. O. Kovrov[1], D. Chumichov[1]
[1]Dnipro Polytechnic National Technical University, Ukraine,
[2]Ukrainian State University of Science and Technology, Ukraine
EMAIL: [1]sergieieva.k.l@nmu.one, [2]yukavats@gmail.com, [1]kovrov.o.s@nmu.one, [1]chumychov.d.d@nmu.one

Abstract. *Information technology has been developed for the automated detection of abandoned cropland based on time series of Sentinel-2 satellite monitoring data. A classification method is proposed that relies on comparing the maximum values of the Normalized Difference Vegetation Index (NDVI) between reference and target years. A decrease in this indicator is interpreted as a sign of*

degradation of the anthropogenic agricultural landscape and the restoration of natural vegetation cover. The information system, implemented on the basis of the developed technology, is integrated with Google Earth Engine (GEE) and enables classification of fields as cultivated or abandoned in near real time, with the results displayed on an interactive map. The technology uses satellite imagery and does not require a training dataset, which makes it possible to monitor agricultural areas even in war zones where ground surveys are difficult or impossible. Validation on a sample of croplands in Ukraine demonstrated an accuracy of up to 92% (F1-score = 0.896).

Keywords: *information technology, information system, classification, NDVI, Sentinel-2, monitoring, abandoned cropland*

UDC 004.05
DOI https://doi.org/10.36059/978-966-397-538-2-12

ANALYSIS OF PERFORMANCE METRICS FOR LOAD TESTING TOOLS

Ph.D. S. Khlamov[1[0000-0001-9434-1081]], **M. Mendielieva**[1[0009-0002-4282-3147]], **Ph.D. O. Vovk**[3[0000-0001-9072-1634]], **Ph.D. Zh. Deineko**[4[0000-0001-6747-9130]], **S. Lytvynenko**[5[0009-0003-2632-9082]]

Kharkiv National University of Radio Electronics, Ukraine
EMAIL: [1]sergii.khlamov@gmail.com, [2]mariia.mendielieva@nure.ua,
[3]oleksandr.vovk@nure.ua, [4]zhanna.deineko@nure.ua,
[5]serhii.lytvynenko@nure.ua

Abstract. *This study presents a detailed comparative analysis of two widely used tools, JMeter and Postman, for performance testing of application programming interfaces (APIs). As APIs form the backbone of modern distributed applications and cloud-based services, ensuring their reliability and efficiency under different traffic conditions is of significant importance. To address this, performance metrics, including average, minimum, and maximum response times, as well as error rates, were systematically collected and evaluated from five publicly available APIs representing diverse functional domains. The experimental results demonstrate apparent differences in the behavior of the two tools depending on the load intensity. Postman exhibits better stability and efficiency under low and moderate load conditions, which makes it suitable for steady-load testing and routine validation of API endpoints during development. In contrast, JMeter demonstrates superior performance in high-load and peak-load scenarios, highlighting its capability to simulate concurrent user actions and stress-test applications at scale. Furthermore, the performance deltas observed during the experiments indicate that JMeter provides a more accurate model of applications in cases where user interactions introduce delays, making it particularly useful for event-driven or session-based systems. These findings emphasize that no single tool is universally*

optimal, and the selection should depend on the specific testing goals and operational context. By providing empirical evidence of the strengths and limitations of both tools, this study offers practical guidance for developers and QA engineers in choosing not only the appropriate tool but also designing effective testing strategies to ensure system robustness and scalability.

Keywords: *Performance testing, API testing, load testing, system stability, Postman, JMeter, response time analysis, percentile metrics, performance evaluation, decision-making, software testing*

1 Introduction

Performance testing of a web Application Programming Interface (API) involves assessing its response time, reliability, scalability, and resource utilization to evaluate its overall performance. API performance tests evaluate how effectively an API handles high traffic and transactions while maintaining performance standards, helping identify bottlenecks and potential issues in its design and execution [1].

Representational State Transfer (REST) APIs are widely used to build modern microservices and cloud applications [2]. They are available in every programming language, and their interface enables a more efficient and effective development process. Numerous performance tools are available for API performance testing, each offering unique features and capabilities [3].

Effective microservices performance testing requires a combination of appropriate tooling and methodological approaches that account for the distributed nature of request processing in microservices architectures [4, 5].

Several levels of performance testing maturity have been identified [6]. Depending on a company's size, project capabilities (resources allocated for testing, budget, etc.), and the maturity of the performance testing process, different outcomes can result. For example, some organizations might discover only a small number of performance defects after deployment (5%). In comparison, others may find a significant number of performance defects in production (30%) due to the limited time allocated for performance testing, which is often performed late in the development cycle. In contrast, some performance defects may only be addressed in the live environment, exposing the application to serious risk.

Software applications that utilize APIs vary significantly, ranging from very simple to large and complex systems, with different numbers of end users located in various regions, and supporting varying levels of concurrent usage. Therefore, depending on these factors and the available resources for testing, different performance testing tools may be employed. Each tool has its own strengths and weaknesses, which must be considered when integrating it into the performance testing process. When selecting a suitable performance testing tool, consider the project's performance testing goals and the tool's capabilities.

Traditional performance testing processes often rely on synthetic scenarios and lengthy testing procedures, making them challenging to adopt when testing needs to be realistic and efficient [7]. Moreover, during the organization of performance testing, some issues related to tools can be observed, such as tool installation, the

tool's flexibility in the application, and the response time generated by the tool [8]. Thus, there is a need to understand the real effectiveness of performance test tools.

Furthermore, with the widespread adoption of agile methodologies, development processes can now run faster and iteratively [9]. Continuous Integration and Continuous Delivery (CI/CD) are methods used in Agile development to automate and speed up the software development and testing process. Hence, the speed or performance of a test tool for API testing can significantly impact its effectiveness.

Studies [3, 10] indicate that Postman and JMeter are highly popular tools for API and performance testing, with usage rates of 45% and 20%, respectively. Additionally, these test tools have different learning curves, with JMeter's being particularly steep [11], whereas Postman is designed to simplify the testing process of APIs [12].

Both test tools have different adoption rates in development teams, with Postman having a higher rate (78%) and JMeter a slightly lower one (59%) [10].

Postman has a very friendly interface, and it is very popular for API testing. Postman provides built-in performance testing features, including load testing, to evaluate API performance under simulated traffic. This helps identify bottlenecks and ensures APIs can handle loads using load profiles. [13]. The core functionality of Postman is built around the Hypertext Transfer Protocol (HTTP), HTTPS, and WebSocket protocols. Postman is highly effective for functional API testing, seamlessly integrating with CI/CD pipelines via the Newman CLI tool, and providing excellent opportunities for team collaboration. In the context of performance testing, it provides a basic test setup and is suitable for small to medium-sized projects.

Apache JMeter is a performance testing tool that supports various APIs, including Simple Object Access Protocol (SOAP) and GraphQL [14]. It is well-known for its realistic load testing features and integration with CI/CD tools. It's suitable for quickly identifying performance issues in high-load systems [15]. JMeter offers flexible configuration, and it is effective for more complex projects that require precise performance metrics. With the help of JMeter, distributed load testing with more than 10,000 VUs can be conducted, making it a successful tool for large projects with high requirements for accuracy and detailed reporting.

Additionally, as shown in the work [16], Postman outperformed JMeter and Robot Framework in various data environments. The relevance of this study is undeniable, as it provides valuable insights into the performance testing capabilities of modern tools in varied environments.

Studies [17, 18, 19] show that Postman and JMeter are commonly used in conjunction with each other. Still, for different types of tests, Postman is typically used for API functional testing, while JMeter is primarily used for performance testing of APIs. However, researchers rarely provide direct comparisons between these performance testing tools, which could assist in selecting the suitable tool for various projects and testing scenarios.

This work aims to compare Postman and JMeter as tools for API performance testing, using performance testing metrics such as average response time, minimum and maximum values, and error rate (%). By evaluating their strengths and limitations across four common load types (load, stress, spike, and soak tests) for create, update, delete (CRUD) public API calls, this work aims to determine which tool is more effective in different testing scenarios. This comparison will help Information Technology practitioners make informed decisions when selecting the most suitable tool for their API performance testing needs in real software development projects.

The chapter is structured to provide a comprehensive examination of methodologies, tools, and experimental findings. The introduction highlights the importance of load testing in modern software engineering, followed by a literature review that discusses existing studies and industry practices.

The methodology section presents the selected tools, test environment, and experimental design. Next, the chapter outlines the performance metrics under investigation, including response time, throughput, and error rate, along with their relevance. Finally, the results, discussion, and conclusions are provided to summarize the findings and offer practical recommendations.

2 Literature review

Performance testing is crucial for assessing whether web applications deliver a satisfactory user experience under various workloads [20]. A workload reproduces the interactions of multiple concurrent users with the system to observe its actual behavior under stress. Defining meaningful workloads is a key challenge in performance testing [21].

Robust application programming interfaces (APIs) and web applications are critical for the seamless operation of enterprise systems [3]. Robustness in web applications has become increasingly important due to the complexity of software systems and the heightened demand for security, performance, and user satisfaction. Various studies emphasize different facets of robustness, focusing on testing methodologies, performance metrics, and security measures.

Performance testing ensures that the system under test (SUT) behaves as expected under different workloads. For web applications, a workload is equivalent to multiple concurrent users making requests to a web server [22, 23]. By simulating these workloads, testers can analyze system performance and identify potential issues, such as slow response times, resource bottlenecks, or failures under high-stress conditions.

There are several popular methodologies or types of performance tests [24]. Load testing involves testing a software system under various levels of user load to evaluate its response times and overall performance across different usage scenarios. Stress testing involves applying extreme loads to an application under test to identify potential failure points and assess system recovery mechanisms.

Scalability testing aims to determine how well a software system can handle increased loads by adding more resources, such as servers or virtual machines, to

meet the demands of larger user bases. Endurance testing, also known as soak testing, involves subjecting an application to a sustained load for an extended period to identify performance degradation issues.

Spike testing checks the response of an application to sudden and extreme increases in user load (e.g., user traffic spikes unexpectedly). Volume testing focuses on evaluating the system's performance when handling large volumes of data, identifying problems with data handling and database performance. Concurrency testing evaluates a system's ability to handle multiple simultaneous users or transactions efficiently.

Although the types of performance tests described above aim to achieve different goals, their design, execution, and analysis are fundamentally similar [21, 23, 25]. Firstly, during the test design phase, a workload is created. This can be done in two approaches [21, 23]. The first one focuses on making requests according to a specific target rate (e.g., simulating a certain number of immediate requests in a given time interval). The second approach simulates user interactions more realistically through a sequence of requests that reflects typical user behavior, including periods of inactivity.

Secondly, test execution is conducted. It can be automated through a load generation in a test tool. JMeter enables manual workload configuration through a graphical user interface. Its load generation is performed using a component called Thread Group, which represents one or more concurrent users that perform the same actions. Thread Group properties include the number of concurrent users and the frequency or duration of each user's actions. Alternatively, Postman allows users to configure and run performance (load) tests on APIs using the Load Profiles or Collection Runner.

The workload configuration involves settings as load profiles (fixed, ramp-up, spike, peak), virtual users or VUs (specifies the number of concurrent users to simulate during the test, with each virtual user executing requests from the selected collection in a repeating loop, where system resources and collection complexity determine the maximum number of virtual users), test duration, and using data file to supply custom values for each virtual user, allowing for more realistic simulations with varied input data.

Finally, during the test analysis phase, performance test metrics obtained after test execution are analyzed to detect threshold violations or anomalies by comparing observed behavior against expected norms.

To conduct effective API performance testing, it is crucial to utilize tools that not only simulate load but also provide detailed metrics that help identify bottlenecks within the system [26, 27]. Service speed requirements are primarily determined using throughput and response time measurements [28].

Meanwhile, web service workload performance is more easily demonstrated through CPU load and memory usage [29]. Performance metrics can be used as BPIs (Basic Performance Indicators) and are vital for defining SLA (Service Level Agreement). While customers often choose BPI metrics to increase system or

service adoption rates, SLA metrics were developed to govern contractual agreements between service providers and end-users [30].

Performance testing teams measure various metrics during test execution, including average response time, minimum response time, maximum response time, and percentiles (p90, p95, p99) [31]. Response time is the total time taken to process a client request to a web service, including both the request transmission and the server's response time [32]. The average response time is the mean time for all requests made by virtual users during a performance test, measured in milliseconds.

The Error % metric indicates the percentage of failed requests during server connection attempts. Throughput measures the number of requests served per unit of time, reflecting application availability, and it has a linear relationship with average response time.

Performance testing tools verify the system or application before delivering it to customers, and their efficiency builds confidence in users, ensuring they always view accurate statistical results [33]. Existing studies primarily focus on analyzing the key features of Postman, JMeter, and other performance tools based on their technical documentation and specifications, as well as their user-friendliness level, especially in terms of setting up and running tests [3, 10, 34-39].

These tools offer distinct capabilities: JMeter is designed primarily for load and performance testing, and supports a wide variety of protocols. At the same time, Postman focuses on simplifying API performance and functional testing with an intuitive user interface.

A variety of research works compare performance testing tools using quantitative metrics such as average response time, system load, resource consumption, and the complexity involved in test setup and operation [26, 29, 40, 41, 42].

According to authors [43, 44, 45, 46], Apache JMeter is a performance testing tool that enables load testing on various protocols and technologies. It is one of the most widely used open-source tools for performance testing, particularly in the domains of API testing and web applications. JMeter runs on any Java-compatible OS (including Linux, Microsoft Windows, macOS, etc.).

The ability to create and execute complex testing scenarios is one of JMeter's most essential features. The JMeter tool is multithreaded and can simulate a large number of VUs, enabling the simulation of a heavy load by distributing tests across multiple machines.

This tool may be used to test the performance of both static and dynamic resources. Moreover, it is capable of managing both Load and Performance Testing techniques for static and dynamic resources. The results [16] show that JMeter efficiently performs performance testing for quality enhancement compared to other open-source software.

Despite its many advantages, JMeter also has certain limitations, as noted by authors [48-50]. One of the main disadvantages is its relatively high memory consumption, especially when running large-scale tests or simulating a large number of users.

Additionally, JMeter also lacks advanced features, such as real-time monitoring. It has a high learning curve for setting up and configuring distributed tests. Additionally, JMeter lacks a scalable GUI and can be slow when managing complex test plans with a high volume of data.

The effectiveness of using Postman was also established by the authors [50-53] in studies on API performance testing. Postman is a platform for API development and testing that has emerged as a leading tool for API development, offering a very user-friendly interface. It can be used in two forms: as a downloadable client and as a web application. Postman is not an open-source tool. It provides both a free version and a paid version with additional features.

Postman tests can be executed manually using the GUI. Additionally, tests can be run automatically on a schedule using the Collection Runner, or they can be run using the command-line tool companion Newman, which enables the automated execution of Postman Collections directly or as part of a CI/CD pipeline.

Postman supports a wide array of testing functionalities, including request chaining, parameterization, and environmental variable management, which allow for the creation of complex test scenarios. Additionally, Postman enables you to collaborate with teammates by organizing, sharing, and communicating work with APIs.

According to authors [51-54], Postman can be used for API performance testing with a desktop application. Performance tests can be run for a collection of API requests using 1 of 4 load profiles:

1. Fixed, where a constant number of VUs are running tests in parallel;
2. Ramp up, when the number of VUs slowly increases from the initial load to the maximum;
3. Spike, where the number of virtual users increases from the base load to the maximum, then decreases back to the base load;
4. Peak, during which the number of virtual users increases from the base load to the maximum, holds steady, then decreases back to the base load.

Postman provides the option to reuse existing API collections for performance testing with minimal scripting effort. The data file feature enables testers to use the dataset file required to load-test the API with different datasets in each iteration.

Additionally, the number of VUs and test duration should be configured before running a performance test. It is essential to highlight that during performance test execution in Postman, each virtual user runs the requests in the specified order in a repeating loop. All of the virtual users operate in parallel to simulate real-world load on the API in a collection.

Performance test execution can be monitored in real-time through the Postman Summary tab. A summary of performance metrics is available in both tabular and graphical forms, including test duration, VUs count, total request count, requests/second, average response time, and API errors. Furthermore, percentile performance metrics are provided in the report (P90, P95, P99). Test reports, including results and details of each performance metric, are available after test execution. They can be downloaded in PDF or HTML formats and shared via a link.

Thus, analyzing the research results in the reviewed authors' works [51, 52], it is worth noting that there are some limitations to running performance tests in Postman. Firstly, a limited number of performance runs are available each month at no additional cost. If the performance run number is reached, the month's limit is then met, and the paid Postman plan or purchased add-ons can be used.

Secondly, the number of VUs in a performance test depends on available system resources and the collection used for the test. According to the Postman 2024 guidelines, a host with 8 CPU cores and 16 GB of RAM can simulate up to 250 VUs, and a host with 16 CPU cores and 32 GB of RAM can simulate up to 500 VUs.

Attempting to simulate a higher number of virtual users may cause inaccurate metrics and reduced throughput. Additionally, one area for improvement in Postman is that timer features for managing the frequency of requests and a sleep time option to introduce a delay between requests, emulating real-world scenarios, are unavailable, unlike in other load-testing tools. Also, a performance test scenario can have only one data file, which is an unlikely scenario in load testing.

Although both JMeter and Postman have their respective advantages and drawbacks, the selection of the appropriate tool depends on the specific performance goals of the project. However, studies comparing these tools across various scenarios remain limited, and many aspects, such as the performance of the tools, require further investigation.

3 Methodology

This work focuses on a comparative approach to evaluate the performance of JMeter and Postman under controlled conditions. To perform performance testing, a set of public APIs was selected for testing, as this approach does not violate their ethical use.

The methodology focuses on testing create, update, and delete (CRUD) endpoints of public APIs, such as GET, POST, PUT, and DELETE, because this helps to simulate real-world interactions with them. These HTTP methods represent the typical operations that users or systems perform when interacting with an API. Each request type was used under different types of load. When using different types of requests with public APIs, it is essential to note that public APIs often return simulated responses (mocked data instead of real data).

However, those public APIs often have different performance characteristics depending on the type of request. Although the responses may be mocked, they offer insight into how the API processes requests and handles various loads. This can help assess whether the API's response times are efficient and stable under load.

The endpoints selected for testing include APIs from five public resources, featuring both simulated and real data, allowing for quick testing without the need to develop a custom backend, such as ReqRes.in, DummyJSON, SampleAPIs, JsonPlaceholder, FakeStoreAPI.

CRUD operations were tested in the form of sequential requests, with a request payload similar to the one shown in Figure 1. It should be noted that the *{postId}*

variable shown in Figure 1 is a generated ID of a new user by the service in the POST request.

This *{postId}* variable was saved after executing the POST request using the *pm.environment.set()* function in Postman collections, as well as by using the Regular Expression Extractor in JMeter tests. After that, *{postId}* was passed to the PUT and DELETE requests for subsequent API calls. Such an approach was implemented in tests for ReqRes.in, SampleAPIs, FakeStoreAPI services. For the DummyJSON and JsonPlaceholder services, existing resource values were used for testing due to limitations of those services.

Different resources were used for public APIs:

- 'users' resource https://reqres.in/api/users (Figure 1);
- 'products' resource of DummyJSON service https://dummyjson.com/products;
- 'codingResources' resource https://api.sampleapis.com/codingresources/codingResources;
- 'posts' resource https://jsonplaceholder.typicode.com/posts;
- 'products' resource of FakeStoreAPI service https://fakestoreapi.com/products.

```
curl --location 'https://reqres.in/api/users?page=1' --header 'x-api-key: reqres-free-v1'

curl --location 'https://reqres.in/api/users' \
--header 'x-api-key: reqres-free-v1' \
--header 'Content-Type: application/json' \
--data-raw '{
            "email": "pikachu@reqres.in",
            "first_name": "Pika",
            "last_name": "Chu",
            "avatar": "https://reqres.in/img/faces/12-image.jpg"
        }'

curl --location --globoff --request PUT 'https://reqres.in/api/users/{{postId}}' \
--header 'x-api-key: reqres-free-v1' \
--header 'Content-Type: application/json' \
--data '{
    "name": "Anders",
    "job": "Anderson"
}'

curl --location --globoff --request DELETE 'https://reqres.in/api/users/{{postId}}' \
--header 'x-api-key: reqres-free-v1'
```

Figure 1. Requests to ReqRes.in 'users' resource

For both test tools, JMeter and Postman, the test scenarios involved sequential execution of GET, POST, PUT, and DELETE requests to the public APIs. All requests were identical between the tools and were directed to the same public APIs test servers.

It's important to get objective comparison results of Postman and JMeter test tools. For this reason, it is critical to conduct performance testing under identical

conditions, considering the following factors: number of VU, Think Time or Delay, and test duration. VUs simulate the behavior of real users interacting with the system. The number of VUs directly impacts the load on the system being tested.

If the number of VUs is too low, the results may not accurately reflect how the system performs under high traffic. Conversely, if the number of VUs is too high, the system might experience excessive strain, potentially leading to bottlenecks that don't align with standard usage patterns.

Think Time, also known as Delay, refers to the pause between user actions (or requests). In real-world scenarios, users don't send requests continuously without any pause; they typically take a brief moment to think or interact with the system.

The Test Duration specifies how long the performance test will run, and it can impact the stability and consistency of the results. Short tests may not provide sufficient data to accurately measure the system's performance under sustained load.

In contrast, longer tests can identify performance degradation, memory leaks, or other issues that emerge over time. A combination of different values for the factors mentioned above can be used to design test cases that closely resemble realistic user behavior. Performance test cases should be performed for all load profiles (Ramp Up, Spike, Peak, and Fixed) in all five public APIs using both tools, JMeter and Postman.

Think Time can be implemented in JMeter using the Constant Timer element, as shown in Figure 2a. Similarly, a GET Delay request can be used in Postman for this purpose - GET https://postman-echo.com/delay/X, where X is the number of seconds to pause, to emulate realistic user behavior.

In general, the structure of all tests for all five public APIs in JMeter is similar to Figure 2a (using Loop Controller with GET, POST, PUT, DELETE requests and listeners inside it). Still, the Thread Group type should be varied depending on the load type (Thread Group, Ultimate Thread Group, or Concurrent Thread Group).

In Postman, all CRUD requests and Delay requests after each of those requests were organized into a collection for all five public APIs, as shown in Figure 2b.

a) b)

Figure 1. Test cases structure: a) JMeter Loop Controller with requests; b) Postman collection of requests

The load parameters of test cases should be designed to have similar parameters for different load scenarios in both test tools.

For this reason, different Thread Group types can be used in JMeter, including Thread Group, Ultimate Thread Group, and Concurrent Thread Group.

Performance tests were conducted under identical conditions:

- Number of VUs: 10, 20, 30, 40, or 50, depending on the scenario;
- Think Time was ranged: 1000 ms (peak load), 3000-5000 ms (realistic load), 10000 ms (low load);
- Test duration: 5 or 10 minutes, depending on the scenario.

To evaluate Postman and JMeter behavior under different user scenarios, four load profiles were used: Ramp Up, Spike, Fixed Load, and Peak Load.

In each of these profiles, the values of VUs and Think Time intervals were varied depending on the test goal (see test cases in Table 1).

Table 1

Performance Test Cases for JMeter and Postman

TC No	Load Profile	VU, N	Think Time, sec	Duration, min	Comment
1	Ramp Up	0-10	10	10	Increase in load
2	Ramp Up	10-30	3	10	Increase in load
3	Spike	1-10-1	10	10	Users attack simulation
4	Spike	5-50-5	2	5	Users attack simulation
5	Peak	2-10-2	10	10	Check of requests max
6	Peak	8-40-8	1	5	Check of requests max
7	Fixed Load	10	10	10	Stable request flow
8	Fixed Load	20	5	10	Stable request flow

The baseline scenario was 10 VU with a Think Time of 10 seconds, during which all five public APIs functioned without errors. It was done to ensure consistent conditions and simplify the results.

Additionally, stress testing scenarios were applied with increased request frequency (think time ranging from 1 to 5 seconds) and peak load values (up to 80 VUs) to obtain results that closely mimic realistic user behavior. Test cases are described in Table 1, and each test case (TC) was run on five public APIs.

The experiment was conducted in a controlled environment (laptop) with consistent configurations to ensure reliable results, including Windows 11 x64 24H2, an 8-core CPU, and 16GB RAM. During the experiment, the following tools were utilized: Postman Desktop application version 11.50.2 and Apache JMeter version 5.6.3.

The primary metrics used to evaluate the performance of test tools included average (avg) response time, minimum (min) and maximum response time values, and error rate (%). The average response time measures the sum of all response times divided by the total number of requests [31].

It represents the typical response time the user will experience. It can be used to evaluate the overall performance efficiency of a system under normal operating conditions. The minimum response time indicates the shortest amount of time it takes for the system to respond to a user request. It represents the best-case scenario and can be used to assess the system's baseline performance under optimal conditions.

Maximum response time refers to the longest period the system takes to respond to a user request. It represents the worst-case scenario and can be used when identifying potential performance bottlenecks or issues under stress or peak load conditions. Error Rate shows the percentage of requests that failed or didn't receive a response.

This metric identifies the issues and bottlenecks that impact the system's performance. In this study, we focused on analyzing the average response time as an indicator of overall test tool performance.

Data was collected using JMeter's built-in test reports and Postman's performance test report. Results were collected for each scenario and metric.

During the performance tests, the difference in average response time metric values (deltas) was calculated to analyze the impact of varying load conditions on response times. Additionally, the aggregated mean values for response times were calculated across all test cases and for each (API, test tool, HTTP method) combination, which helps to provide a representative performance assessment.

The difference in metric values (deltas) can be calculated to analyze which types of requests were slower in Postman or JMeter. This can be done using the formula:

$$\Delta\text{Avg} = \text{Postman}_{\text{Avg}} \quad \text{JMeter}_{\text{Avg}}, \tag{1}$$

where ΔAvg is the difference in average response time between Postman and JMeter;

$\text{Postman}_{\text{Avg}}$ is the value of the average response time in Postman;

$\text{JMeter}_{\text{Avg}}$ is the value of the average response time in JMeter.

If the case is a positive value, then Postman had a longer average response time than JMeter. If it is a negative value, then JMeter's average response times exceeded those in Postman, and Postman's performance was faster.

Average values of each performance metric are calculated for 8 test cases for each API and HTTP method:

$$\text{Avg\_metric} = \frac{\sum_{k=1}^{n} m_k}{n}, \tag{2}$$

where m_k – value of performance metric for $test\_case_k$;

n – number of test cases

This should be repeated for 5 APIs using a test tool (such as Postman or JMeter) and a selected HTTP method. Then, mean aggregated values should be calculated for average performance metric values for the selected HTTP method:

$$\text{Mean\_metric} = \frac{\sum_{i=1}^{l} \text{Avg\_metric}_i}{l}, \quad (3)$$

where Avg_metric – value of performance metric for selected HTTP method;

l – number of APIs

The final step after analyzing aggregated metrics and their deltas is visualizing them through bar charts to evaluate the differences between the test tools.

It is important to note that although the load parameters were standardized across both test tools, the execution architecture differs between them. In Postman, each VU executes requests sequentially (one after the other). In contrast, in JMeter, each thread is executed in parallel, potentially creating a higher load (higher Requests Per Second (RPS)).

As a result, Loop Controller elements with GET, POST, PUT, and DELETE requests can be added to JMeter test plans within a Thread Group to ensure more accurate and comparable test execution with similar load profiles, similar to Postman.

4 Results

Obtained performance test results for JMeter and Postman tools, based on the metric of average response time. A method of comparing aggregated metrics (deltas) was used, along with their visualization through bar charts, to analyze the differences between the tools. This visualization helps to ensure that the differences between the tools are not random fluctuations, but are consistent across all APIs. Both delta values and aggregated mean values for the average response time are calculated and analyzed to provide a comprehensive comparison of the tools' performance. The response times for JMeter and Postman were compared across different test cases.

The test load in TC1 begins with an initial load of 0 VUs and then steadily ramps up to 10 users over 5 minutes, with a total test duration of 10 minutes. Postman tests used the Ramp Up load profile in Postman with 10 VUs, an initial load of 0 VUs, and a test duration of 10 minutes. To obtain an equivalent load in JMeter, a Thread Group with 10 VUs and a ramp-up period of 300 s and a duration of 600 s was used. In this case, the Loop Count was set to 1 in the Loop Controller of JMeter.

The delta between response times in the TC1 ranged from 13 to 87 ms, with Postman showing a slight advantage (negative values) over JMeter (positive values) for the majority of public APIs' services, as illustrated in the bar chart (see Figure 3).

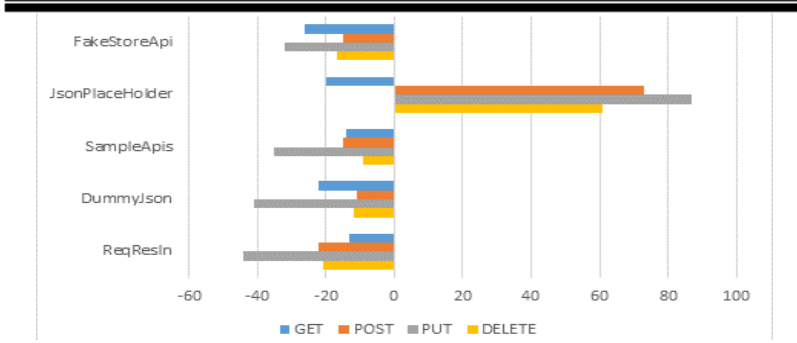


Figure 3. Deltas of avg response time, TC1

The load in TC2 is gradually increased from 10 to 30 VUs over 10 minutes. It is generated with a Ramp-Up profile, starting with 10 VUs that run for 2 minutes and 30 seconds, then ramping up to 30 VUs for another 2 minutes and 30 seconds. After that, the load is maintained at 30 VUs for 5 minutes. Postman tests used the Ramp Up load profile with 30 VUs, an initial load of 10 VUs, and a test duration of 10 minutes. To obtain an equivalent load in JMeter, Ultimate Thread Group with two threads was used, as shown in Figure 3b: 1) start threads count = 10, initial delay = 0 sec, startup time = 0 sec, hold load for = 600 sec, shutdown time = 0; 2) start threads count = 20, initial delay = 150 sec, startup time = 150 sec, hold load for = 300 sec, shutdown time = 0. In both cases, the Loop Count was set to 'Infinite' in the Loop Controller of JMeter. Deltas show that the average response time in Postman was greater than in JMeter for 3 out of 5 public APIs. The difference in average ranges was from 1 to 113 ms, indicating an advantage of JMeter over Postman under those load conditions, with a time of 3000 ms (see Figure 4).

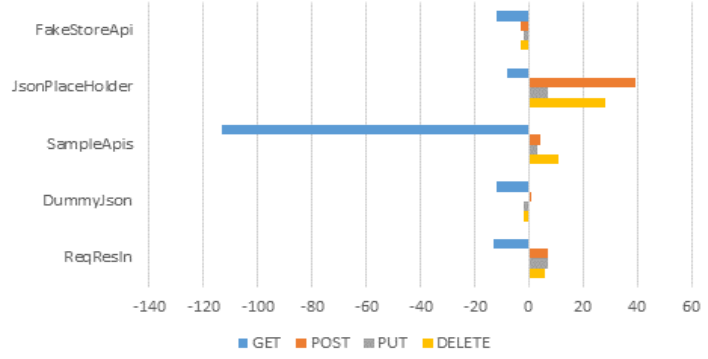


Figure 4. Deltas of avg response time, TC2

TC3 used load generation with a sudden spike to 10 VUs, followed by a decline, with a total test duration of 10 minutes and a think time of 10,000 ms. Postman tests

used a Spike load profile with 10 VUs, a base load of 1 VU, and a test duration of 10 minutes.

To obtain an equivalent load in JMeter, Ultimate Thread Group was used: 1) start threads count = 1, initial delay = 0 sec, startup time = 0 sec, hold load for = 600 sec, shutdown time = 0; 2) start threads count = 9, initial delay = 240 sec, startup time = 60 sec, hold load for = 0 sec, shutdown time = 60.

The delta between response times in TC3 was from 16 to 113 ms, with Postman showing a slight advantage over JMeter in 3 out of 5 APIs (see Figure 5).

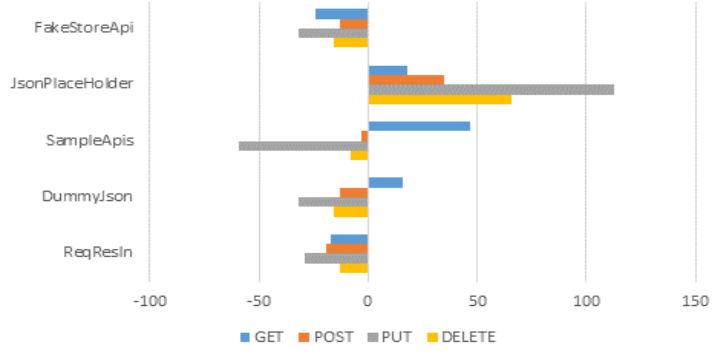


Figure 5. Deltas of avg response time, TC3

In TC4, a sudden spike to 50 VUs was applied, followed by a decline in the number of VUs. The test duration was 5 minutes, and the think time was 5000 ms. Postman tests used a Spike load profile with 50 VUs, a base load of 5 VUs, and a test duration of 5 minutes.

To obtain an equivalent load in JMeter, Ultimate Thread Group was used: 1) start threads count = 5, initial delay = 0 sec, startup time = 0 sec, hold load for = 300 sec, shutdown time = 0; 2) start threads count = 45, initial delay = 120 sec, startup time = 30 sec, hold load for = 0 sec, shutdown time = 30.

It's worth mentioning that for the ReqRes.in service, a vast majority of 429 errors (too many requests) were returned in response when the number of VUs started to increase, for instance, significantly in TC4.

For other APIs, 429 and 502 errors were returned in response, but their number was insignificant. The delta between response times in TC4 was from 1 to 75 ms, with Postman showing a slight advantage over JMeter in 3 out of 5 APIs (see Figure 6).

TC5 used a load with a sharp increase to a peak of 10 VUs, followed by a gradual decrease in voltage. The test duration was 10 minutes, and the think time was 10,000 ms. Postman tests used a Peak load profile in Postman with 10 VUs, a base load of 2 VUs, and a test duration of 10 minutes.

To obtain an equivalent load in JMeter, Ultimate Thread Group was used: 1) start threads count = 2, initial delay = 0 sec, startup time = 0 sec, hold load for = 600 sec, shutdown time = 0; 2) start threads count = 8, initial delay = 120 sec, startup time = 120 sec, hold load for = 120 sec, shutdown time = 120.

For most types of requests, Postman demonstrated a faster average response time than JMeter, with deltas ranging from 4 ms to 123 ms. Notably, the most significant positive delta (123 ms) occurred for the JsonPlaceHolder API, where JMeter outperformed Postman. The chart highlights the consistency of Postman's performance across the majority of APIs (see Figure 7).

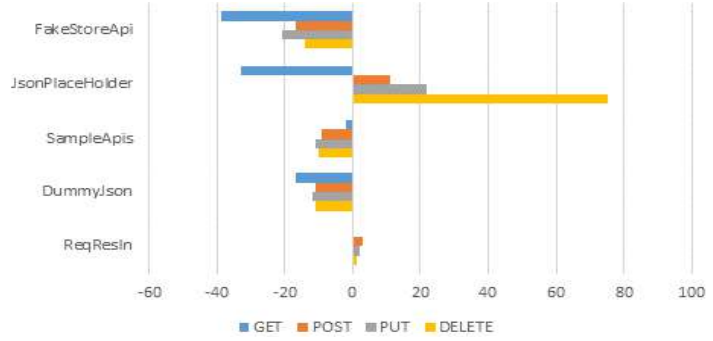


Figure 6. Deltas of avg response time, TC4

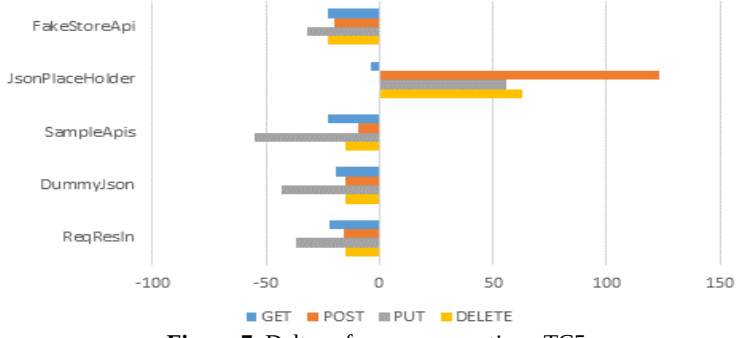


Figure 7. Deltas of avg response time, TC5

In TC6, a sharp increase in load to a peak of 40s VUs was used, followed by a gradual decrease. The test duration was 5 minutes, with a think time of 1000 ms. Postman tests used a Peak load profile in Postman with 40 VUs, a base load of 8 VUs, and a test duration of 5 minutes. To obtain an equivalent load in JMeter, Ultimate Thread Group was used, as shown in Figure 8b: 1) start threads count = 8, initial delay = 0 sec, startup time = 0 sec, hold load for = 300 sec, shutdown time = 0; 2) start threads count = 32, initial delay = 60 sec, startup time = 60 sec, hold load

for = 60 sec, shutdown time = 60. Deltas show that JMeter significantly outperformed Postman for 4 out of 5 public APIs. The difference in the average ranges was from 1 to 59 ms. Those results demonstrate a practical advantage of JMeter under load conditions, with a sharp increase in load to a peak of 40 VUs, followed by a gradual decrease and a realistic short think time (see Figure 8).

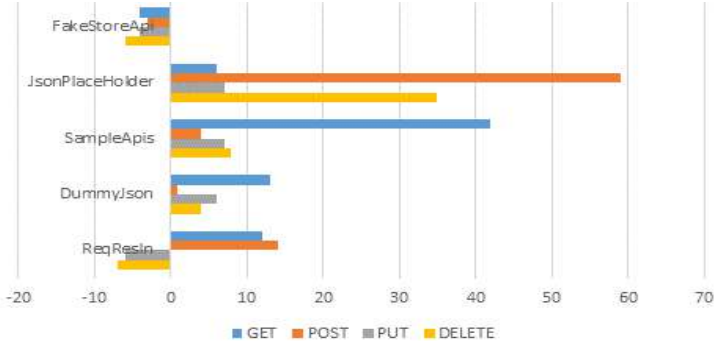


Figure 8. Deltas of avg response time, TC6

TC7 generated a load fixedly, maintaining a stable load for 10 VUs over 10 minutes and a think time equal to 10 minutes. Postman tests used a Fixed load profile in Postman with 10 VUs, test duration = 10 minutes. To obtain an equivalent load in JMeter, the Concurrency Thread Group was used, with a target concurrency of 10 and a hold target rate of 10 minutes. Deltas illustrate that the average response time in Postman was less than in JMeter for 4 out of 5 public APIs. The difference in the average ranges was from 6 to 116 ms. Those results demonstrate a practical advantage of Postman under the stable load conditions with few VUs and an ample think time (see Figure 9). In TC8, the load remained stable for 20 VUs over 10 minutes and had a shorter think time of 5,000 ms. Postman tests used a Fixed load profile in Postman with 20 VUs, test duration = 10 mins. To obtain an equivalent load in JMeter, a Concurrency Thread Group was used with the following settings: target concurrency = 20, hold target rate = 20 minutes. Deltas show that the average response time in Postman was much less than in JMeter for most types of requests in TC7 for 3 out of 5 public APIs. Only PUT and DELETE requests were faster in JMeter for 2 APIs. The difference in the average ranges was from 8 to 35 ms. Those results demonstrate a practical advantage of Postman under stable load conditions with a low number of VUs and medium think time (see Figure 10).

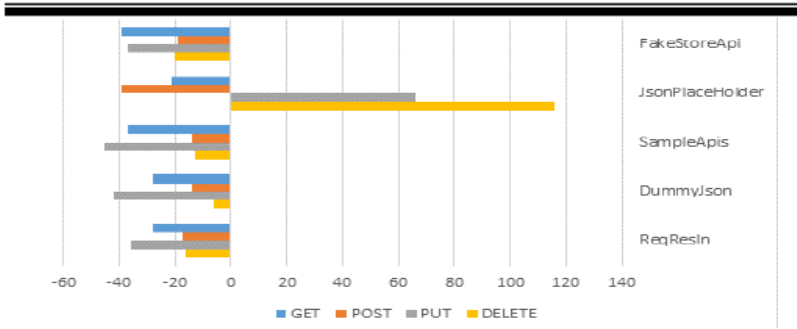


Figure 9. Deltas of avg response time, TC7

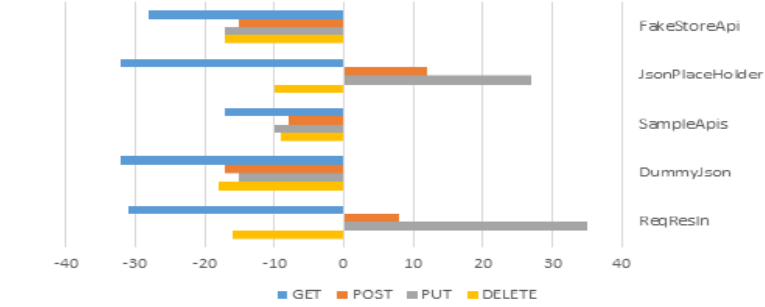


Figure 10. Deltas of avg response time, TC8

These values per API were subsequently aggregated across 5 APIs. The final metrics, including mean response time, minimum, maximum, and error percentage values, accurately reflect the tool's performance per HTTP method, as shown in Table 2. The final aggregated values in Table 2 show that, in general, Postman demonstrates better performance compared to JMeter. Although the test was conducted once, aggregated values of metrics were calculated from hundreds of requests, allowing for an objective assessment of system behavior.

Table 2

Aggregated performance results for Postman and JMeter

Method	Test Tool	Mean Avg, ms	Mean Min, ms	Mean Max, ms	Mean Error, %
GET	Postman	166,175	110,650	415,575	0,935
GET	JMeter	181,650	115,925	499,625	0,739
POST	Postman	148,100	117,250	877,200	7,671
POST	JMeter	147,175	124,700	805,525	7,999
PUT	Postman	144,925	118,100	341,425	7,777
PUT	JMeter	152,075	125,525	377,275	7,936
DELETE	Postman	166,225	136,525	315,475	7,931
DELETE	JMeter	163,325	143,525	374,475	8,515

These data suggest a preliminary conclusion that Postman exhibits higher performance in low to moderate load conditions, particularly during stable loads with a fixed number of VUs, and in other types of loads where think time exceeds 5 seconds. On the other hand, JMeter demonstrates higher performance under conditions of high load and high request intensity, especially in cases of a sharp increase in load to peak, followed by a gradual decrease with a short think time of 1 second. Additionally, a greater number of samples were generated in JMeter than in Postman in all testing cases.

5 Discussions

The analysis is based on a single performance test run for five public APIs, which limits statistical generalizability. However, aggregated values were calculated from hundreds of requests, which allows for a reliable assessment of a tool's behavior at the API level.

Since public open APIs were used, the exact number of active users at the time of the performance test execution remains unknown. However, to minimize this uncertainty, performance tests were conducted at the same time of day during the experiment. Additional experiments with repeated execution of scenarios can be applied to confirm the robustness of the observed effects.

6 Conclusions

In conclusion, based on the analysis of performance testing with JMeter and Postman, it is clear that each tool is more efficient in specific load scenarios. Postman demonstrated higher performance in steady-state conditions, particularly with fixed load profiles and gradual ramp-ups, where it excelled in providing stable response times despite increasing load.

Alternatively, JMeter outperformed Postman in high-load and high-intensity scenarios, showing superior performance in rapid peak load conditions, with minimal delays between user actions.

In the context of performance testing, these differences underscore the importance of selecting the appropriate tool for the specific testing scenario. Postman is well-suited for scenarios that require steady performance under controlled conditions, whereas JMeter is more appropriate for high-stress, high-traffic environments. Understanding the capabilities of these tools can have a positive impact on software development processes.

Additionally, calculated deltas show that decreasing think time for the same type of load, such as ramp-up and peak, outperforms JMeter in those types of requests, which can be helpful for testing applications with delays between user actions.

Postman can be more effective in performance tests, which enable the configuration of a gradual increase in load. This is particularly useful when the load is predictable in advance, such as when testing an API with a limited request intensity. These tools can be a good option for small and medium projects,

especially when there are no strict requirements to simulate a realistic user load. The application is quite simple, and resources for testing are limited.

JMeter is better suited for testing with peak loads and evaluating system behavior under high request intensity (e.g., complex applications with high traffic), as well as for identifying the system's performance limits.

By considering factors such as virtual user load, think time, and the complexity of responses, developers and QA engineers can reduce testing time, mitigate the risk of underestimating system performance, and ensure that the system performs effectively under real-world load conditions.

7 Acknowledgements

The research was supported by the Ukrainian project of fundamental scientific research, “Development of computational methods for detecting objects with near-zero and locally constant motion by optical-electronic devices” (#0124U000259) from 2024 to 2026.

References

1. Godinho, A., Rosado, J., Sá, F. A., & Cardoso, F. (2023). Method for evaluating the performance of web-based APIs. *International Conference on Smart Objects and Technologies for Social Good.* https://doi.org/10.1007/978-3-031-52524-7_3

2. Golmohammadi, A., Zhang, M., & Arcuri, A. (2023). Testing RESTful APIs: A survey. *ACM Transactions on Software Engineering and Methodology.* https://doi.org/10.1145/3617175

3. Joshi, N. Y. (2023). Developing robust APIs and web applications for enterprise applications: Automation frameworks and testing strategies. *Journal of Basic Science and Engineering.*

4. Dhandapani, A. (2025). Automation testing in microservices and cloud-native applications: Strategies and innovations. *Journal of Computer Science and Technology Studies, 7*(3), 826–836. https://doi.org/10.32996/jcsts

5. Niedermaier, S., et al. (2019). On observability and monitoring of distributed systems: An industry interview study. In *International Conference on Service-Oriented Computing* (pp. 36–52). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-33702-5_3

6. Molyneaux, I. (2014). *The art of application performance testing: From strategy to tools.* O’Reilly Media, Inc.

7. Cooper, Q., Krishnamurthy, D., & Amannejad, Y. (2024). Budget aware performance test selection for microservices. In *2024 IEEE 17th International Conference on Cloud Computing (CLOUD)* (pp. 376–385). Shenzhen, China. https://doi.org/10.1109/CLOUD62652.2024.00049

8. Lenka, R. K., Dey, M. R., Bhanse, P., & Barik, R. K. (2018). Performance and load testing: Tools and challenges. In *2018 International Conference on Recent Innovations in Electrical, Electronics & Communication Engineering (ICRIEECE)*

(pp. 2257–2261). Bhubaneswar, India. https://doi.org/10.1109/ICRIEECE44171.2018.9009338

9. Pratama, M. R., & Sulistiyo Kusumo, D. (2021). Implementation of continuous integration and continuous delivery (CI/CD) on automatic performance testing. In *2021 9th International Conference on Information and Communication Technology (ICoICT)* (pp. 230–235). Yogyakarta, Indonesia. https://doi.org/10.1109/ICoICT52021.2021.9527496

10. Nagineni, S. (2025). Advancing software reliability through systematic API testing: A comparative analysis of modern automation frameworks and methodological implications for distributed systems. *JCSTS, 7*(8), 798–805. https://doi.org/10.32996/jcsts.2025.7.8.94

11. Arif, S., Faisal, M., et al. (2024). Software automation testing: Comparing no-code, low-code, and traditional approaches. *Contemporary Journal of Social Science Review*, 2138–2147. https://doi.org/10.63878/cjssr.v2i04.402

12. Sri, S. D., et al. (2024). Automating REST API Postman test cases using LLM. *arXiv preprint.* https://doi.org/10.48550/arXiv.2404.10678

13. Postman. (2024). Configuring performance tests. Retrieved from https://learning.postman.com/docs/collections/performance-testing/performance-test-configuration

14. Apache JMeter. (2021). Supported protocols. Retrieved from https://jmeter.apache.org/usermanual/component_reference.html

15. Bondi, A. B., & Saremi, R. (2021). Experience with teaching performance measurement and testing in a course on functional testing. In *ICPE 21: Companion of the ACM/SPEC International Conference on Performance Engineering* (pp. 115–120). https://doi.org/10.1145/3447545.3451196

16. Hsieh, C.-H., et al. (2021). Evaluation system for software testing tools in complex data environment. In *2021 4th International Conference on Information Communication and Signal Processing (ICICSP)* (pp. 604–609). Shanghai, China. https://doi.org/10.1109/ICICSP54369.2021.9611846

17. Ball, A., & Ochei, L. C. (2024). Remote performance monitoring system for managed service providers. *International Journal of Applied Information Systems (IJAIS).*

18. Koppanati, P. K. (2021). Automation testing for custom insurance quotation engines using microservices architecture. *Journal of Scientific and Engineering Research*, 326–332. https://doi.org/10.5281/zenodo.14005848

19. Noetzold, D., et al. (2024). JVM optimization: An empirical analysis of JVM configurations for enhanced web application performance. *SoftwareX*. https://doi.org/10.1016/j.softx.2024.101933

20. Thakur, S. S. (2025). Technical review: Performance testing methodologies and implementation strategies. *Sarcouncil Journal of Multidisciplinary.*

21. Battista, E., Martino, S. D., Di Meglio, S., Scippacercola, F., & Lucio Starace, L. L. (2023). E2E-Loader: A framework to support performance testing of web applications. In *2023 IEEE Conference on Software Testing, Verification and*

Validation (ICST) (pp. 351–361). Dublin, Ireland. https://doi.org/10.1109/ICST57152.2023.00040

22. Menasce, D. A. (2002). Load testing of web sites. *IEEE Internet Computing, 6*(4), 70–74.

23. Di Meglio, S., et al. (2025). Performance testing in open-source web projects: Adoption, maintenance, and a change taxonomy.

24. Pargaonkar, S. (2023). A comprehensive review of performance testing methodologies and best practices: Software quality engineering. *International Journal of Science and Research (IJSR),* 2008–2014. https://doi.org/10.21275/SR23822111402

25. Jiang, Z. M., & Hassan, A. E. (2015). A survey on load testing of large-scale software systems. *IEEE Transactions on Software Engineering*, 1091–1118. https://doi.org/10.1109/TSE.2015.2445340

26. Hendayun, M., Ginanjar, A., & Ihsan, Y. (2023). Analysis of application performance testing using load testing and stress testing methods in API service. *Jurnal Sisfotek Global***,** 28–34. https://doi.org/10.38101/sisfotek.v13i1.2656

27. Yenugula, M., Kodam, R., & He, D. (2019). Performance and load testing: Tools and challenges. *International Journal of Engineering in Computer Science*, 57–62. https://doi.org/10.33545/26633582.2019.v1.i1a.102

28. Blinowski, G., Ojdowska, A., & Przybyłek, A. (2022). Monolithic vs. microservice architecture: A performance and scalability evaluation. *IEEE Access, 10***,** 20357–20374. https://doi.org/10.1109/ACCESS.2022.3152803

29. Lawi, A., Panggabean, B. L. E., & Yoshida, T. (2021). Evaluating GraphQL and REST API services performance in a massive and intensive accessible information system. *Computers.* https://doi.org/10.3390/computers10110138

30. Duman, I., & Eliiyi, U. (2021). Performance metrics and monitoring tools for sustainable network management. *Bilişim Teknolojileri Dergisi***,** 37–51.

31. BlazeMeter. (2020). Key performance metrics. Retrieved from https://help.blazemeter.com/docs/guide/performance-kpis-key-perf-test-metrics.htm

32. Dhalla, H. K. (2021). A performance comparison of RESTful applications implemented in Spring Boot Java and MS .Net Core. *Journal of Physics: Conference Series***.** IOP Publishing. https://doi.org/10.1088/1742-6596/1933/1/012041

33. Jacob, A., & Karthikevan, A. (2018). Scrutiny on various approaches of software performance testing tools. In *2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA)* (pp. 509–515). Coimbatore, India. https://doi.org/10.1109/ICECA.2018.8474876

34. Srivastava, N., Kumar, U., & Singh, P. (2021). Software and performance testing tools. *Journal of Information, Electronics and Electronic Engineering***,** 1–12.

35. Ali, A., Maghawry, H. A., & Badr, N. (2022). Automation of performance testing: A review. *International Journal of Intelligent Computing & Information Sciences***.** https://doi.org/10.21608/ijicis.2022.161846.1219

36. Samlı, R., & Orman, Z. (2023). A comprehensive overview of web-based automated testing tools. *İleri Mühendislik Çalışmaları ve Teknolojileri Dergisi***,** 13–28.

37. Ushakova, I., Plokha, O., & Skorin, Y. (2022). Approaches to web application performance testing and real-time visualization of results. *Kharkiv National Automobile and Highway University (KHADU), Ukraine***.** https://doi.org/10.30977/BUL.2219-5548.2022.96.0.71

38. Abbas, R., Sultan, Z., & Bhatti, S. N. (2017). Comparative analysis of automated load testing tools: Apache JMeter, Microsoft Visual Studio (TFS), LoadRunner, Siege. In *2017 International Conference on Communication Technologies (ComTech)* (pp. 39–44). Rawalpindi, Pakistan. https://doi.org/10.1109/COMTECH.2017.8065747

39. Theivendran, P. (2023). Investigating usability and user experience of software testing tools. *Authorea Preprints, TechRxiv.* https://doi.org/10.36227/techrxiv.23251076.v1

40. Lenka, R. K., Mamgain, S., Kumar, S., & Barik, R. K. (2018). Performance analysis of automated testing tools: JMeter and TestComplete. In *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)* (pp. 399–407). Greater Noida, India. https://doi.org/10.1109/ICACCCN.2018.8748521

41. Tiwari, V., Upadhyay, S., Goswami, J. K., & Agrawal, S. (2023). Analytical evaluation of web performance testing tools: Apache JMeter and SoapUI. In *2023 IEEE 12th International Conference on Communication Systems and Network Technologies (CSNT)* (pp. 519–523). Bhopal, India. https://doi.org/10.1109/CSNT57126.2023.10134699

42. Neelapu, M. (2023). Enhancement of software reliability using automatic API testing model. *International Journal of Multidisciplinary Research and Growth Evaluation.*

43. Rodrigues, A. G., Demion, B., & Mouawad, P. (2019). *Master Apache JMeter—From load testing to DevOps: Master performance testing with JMeter*. Packt Publishing Ltd.

44. Akiladevi, R., Vidhupriya, P., & Sudha, V. (2018). A study and analysis on software testing tools. *International Journal of Pure and Applied Mathematics*, 1783–1800.

45. Erinle, B. (2017). *Performance testing with JMeter 3.* Packt Publishing Ltd.

46. Indrianto, I. (2023). Performance testing on web information system using Apache JMeter and BlazeMeter. *Jurnal Ilmiah Ilmu Terapan Universitas Jambi*, 138–149. https://doi.org/10.22437/jiituj.v7i2.28440

47. Shyam Mohan, J. S., & Goswami, K. (2025). Performance analysis and comparison of Node.js and Java Spring Boot in implementation of RESTful applications. *Software: Practice and Experience***,** 1209–1233. https://doi.org/10.1002/spe.3418

48. Siddhant, S., & Prapull, S. B. (2020). Comprehensive review of load testing tools**.** *International Research Journal of Engineering and Technology.*

49. AutomateNow. (2023). Advantages and disadvantages of using JMeter. Retrieved from https://automatenow.io/advantages-and-disadvantages-of-using-jmeter

50. TestSigma. (2025). JMeter vs Postman. Retrieved from https://testsigma.com/blog/jmeter-vs-postman

51. Golian, N., & Tisheninova, V. (2025). Integrated graph-based testing pipeline for modern single-page applications. *Bulletin of National Technical University "KhPI". Series: System Analysis, Control and Information Technologies,* 51–59. https://doi.org/10.20998/2079-0023.2025.01.08

52. NashTech. (2024). Performance testing with Postman: Is it worth? Retrieved from https://blog.nashtechglobal.com/performance-testing-with-postman-is-it-worth

53. Susan Rini, V. S. (2024). When Postman goes that extra mile to deliver performance to APIs. *Software Testing Magazine***.** Retrieved from https://www.softwaretestingmagazine.com/tools/when-postman-goes-that-extra-mile-to-deliver-performance-to-apis

54. Savanevych, V., et al. (2023). Mathematical methods for an accurate navigation of the robotic telescopes. *Mathematics, 11*(10), 2246. https://doi.org/10.3390/math11102246

АНАЛІЗ ПОКАЗНИКІВ ЕФЕКТИВНОСТІ ІНСТРУМЕНТІВ НАВАНТАЖУВАЛЬНОГО ТЕСТУВАННЯ

Ph.D. С. Хламов[1][0000-0001-9434-1081]**, М. Мендієлєва**[2][0009-0002-4282-3147], **Ph.D. О. Вовк** [3][0000-0001-9072-1634]**, Ph.D. Ж. Дейнеко**[4][0000-0001-6747-9130]**, С. Литвиненко**[5][0009-0003-2632-9082]

Харківський національний університет радіоелектроніки, Україна
EMAIL: [1]sergii.khlamov@gmail.com, [2]mariia.mendielieva@nure.ua, [3]oleksandr.vovk@nure.ua, [4]zhanna.deineko@nure.ua, [5]serhii.lytvynenko@nure.ua

Анотація. *У цьому дослідженні подано детальний порівняльний аналіз двох широко використовуваних інструментів — JMeter та Postman — для тестування продуктивності інтерфейсів прикладного програмування (API). Оскільки API є основою сучасних розподілених застосунків і хмарних сервісів, забезпечення їхньої надійності та ефективності за різних умов навантаження має надзвичайно важливе значення. З цією метою було систематично зібрано й оцінено показники продуктивності, зокрема середній, мінімальний і максимальний час відгуку, а також рівень помилок, із п'яти відкрито доступних API, що репрезентують різні функціональні домени. Експериментальні результати демонструють очевидні відмінності у поведінці двох інструментів залежно від інтенсивності навантаження. Postman показує кращу стабільність і ефективність за умов низького та середнього навантаження, що робить його придатним для тестування стабільних навантажень і рутинної валідації API-ендпойнтів під час*

розробки. Натомість JMeter демонструє вищу продуктивність у сценаріях високого та пікового навантаження, підкреслюючи його здатність моделювати паралельні дії користувачів і здійснювати стрес-тестування застосунків у масштабі. Крім того, різниця у показниках продуктивності, виявлена під час експериментів, свідчить, що JMeter надає більш точну модель застосунків у випадках, коли взаємодія користувачів спричиняє затримки, що робить його особливо корисним для подієво-орієнтованих або сесійних систем. Отримані результати підкреслюють, що жоден інструмент не є універсально оптимальним, а вибір має залежати від конкретних цілей тестування та операційного контексту. Надаючи емпіричні докази сильних і слабких сторін обох інструментів, це дослідження пропонує практичні рекомендації для розробників і QA-інженерів у виборі не лише відповідного інструмента, але й у проєктуванні ефективних стратегій тестування для забезпечення надійності та масштабованості системи.

Ключові слова: *тестування продуктивності, тестування API, навантажувальне тестування, стабільність системи, Postman, JMeter, аналіз часу відгуку, перцентильні метрики, оцінка продуктивності, прийняття рішень, тестування програмного забезпечення*

UDC 004.05
DOI https://doi.org/10.36059/978-966-397-538-2-13

PERFORMANCE PERCENTILE ANALYSIS FOR API-BASED TESTING

Ph.D. S. Khlamov[1][0000-0001-9434-1081], **M. Mendielieva** [2][0009-0002-4282-3147],
Ph.D. O. Vovk[3][0000-0001-9072-1634], **T. Trunova**[4][0000-0003-2689-2679],
Yu. Teslenko[5][0009-0009-8349-3683]
Kharkiv National University of Radio Electronics, Ukraine
EMAIL: [1]sergii.khlamov@gmail.com, [2]mariia.mendielieva@nure.ua,
[3]oleksandr.vovk@nure.ua, [4]tetiana.trunova@nure.ua, [5]yuliia.teslenko@nure.ua

Abstract. *This paper focuses on the systematic analysis of performance testing metrics, with particular attention to comparing the behavior of two widely used tools, Postman and JMeter, in the context of public application programming interfaces (APIs). API performance testing plays a critical role in evaluating the responsiveness and reliability of modern software systems; however, one persistent challenge lies in relying solely on average response times. Mean values can be easily skewed by anomalous outliers, which often mask significant response delays and lead to an incomplete picture of system performance. To overcome this limitation, the present study emphasizes the use of percentile-based analysis, which provides a more accurate and user-centered indicator of performance for the majority of requests. The methodology involved comparing average response time*

values with percentile distributions, with a primary focus on the 90th, 95th, and 99th percentiles. Additionally, the percentage deviation of percentile values from the average was calculated, serving as a measure of stability for the testing tools. This approach enabled a more reliable evaluation of the tool's behavior under different workloads. The experimental findings revealed distinct advantages for each tool. Postman demonstrated favorable results under medium load conditions for Create and Delete requests. At the same time, JMeter proved more effective for Read (GET) and Update operations, where stability and predictability are critical in complex systems. To support practical application, the study also proposes a decision-making diagram that guides the allocation of test scenarios between the two tools, ultimately improving testing efficiency and ensuring more reliable API performance assessments.

Keywords: *Performance testing, API testing, load testing, system stability, Postman, JMeter, response time analysis, percentile metrics, performance evaluation, decision-making, software testing*

1 Introduction

One of the challenges in analyzing performance test results is that the average response time (ART) represents the typical response time a user will experience while using a software application. On the other hand, the ART performance testing metric does not take into consideration outliers and large deviations in response times [1], which can be caused by issues such as memory leaks, thread contention, Input/Output (I/O) bottlenecks, and long-running SQL queries. A system may have an acceptable ART, but some users may still face significant delays, especially during load spikes or stress conditions [2].

Percentile response time, as a performance testing metric, provides a more reliable representation of system behavior for most users, capturing significant delays that the average response time would miss. A percentile is the value below which a percentage of the response times are completed successfully [3].

The concept of considering slow requests that exceed the threshold is crucial in performance testing, as it helps identify potential performance issues in applications [4]. It often indicates that the system may begin to degrade under high loads, especially in microservice environments and cloud-native applications [5-7].

Furthermore, performance analysts can investigate the internal behavior of various slow and fast web requests in greater detail, and by clustering and comparing their execution patterns, identify the factors that cause specific requests to perform slowly or exhibit unexpected behavior [8]. Additionally, service level agreements (SLAs), as contracts between a service provider and a customer, are often written with a specification of the maximum response time for the majority of users, rather than the average response time of the system [9].

Despite the emergence of modern performance tools and technologies in the performance testing process, a challenge remains in successfully applying testing tools to test microservices and cloud-native application APIs [10, 11]. The distributed nature of these systems creates complex testing scenarios where services

interact across network boundaries with varying degrees of reliability and latency [10].

The interconnected nature of microservices creates numerous dependencies, which increase the testing surface area. This leads to the problem of prolonged testing times in cloud environments, where costs directly correlate with resource allocation. Thus, there is a challenge of resource limitations [11] that is important for organizations with limited budgets or restricted access to resources.

Additionally, in the field of microservices API testing, it's essential to find an approach that can effectively validate complex service interactions while maintaining test stability. Effective API management in microservice architectures demands both the application of general strategic methods and the integration of reliable engineering practices that maintain system stability and reliability [12, 13].

Selecting the appropriate performance tools can be a challenging task, but they should be aligned with the project's requirements and the development team's expertise [11, 14].

In relation to the decision-making process regarding the most suitable performance test tool, it is essential to consider its stability, which is related to its reliability and robustness in consistently performing functions over time, as outlined in the ISO/IEC 25010 standard [15].

Postman and JMeter are popular tools for performance testing. However, they have architectural differences and offer different options for load generation, work collaboration and reporting. These differences raise questions about the efficient allocation of test scenarios between two performance testing tools, which can be aligned with the project's specific requirements. The results will provide IT professionals with objective insights for creating a successful performance test strategy.

The purpose of this work is to study the approach to the distribution of test scenarios between two testing tools, Postman and JMeter, based on performance metrics such as ART and percentile. The results will provide IT professionals with objective insights for developing an effective performance testing strategy and enhancing the flexibility and adaptability of the testing process.

The chapter is organized to emphasize the role of percentile metrics in evaluating system responsiveness and reliability. It begins with an introduction that explains the limitations of traditional average-based analysis and the need for percentile-oriented approaches in API testing. The literature review section examines prior research and current practices in the use of percentiles across performance engineering. The methodology outlines the selected APIs, test scenarios, and statistical models applied. The results section presents percentile distributions under varying loads, while the discussion interprets their significance. The chapter concludes with recommendations for engineers and future research directions.

2 Literature review

API testing is essential for software development teams to ensure the performance, scalability, and security of large-scale systems that handle millions of transactions daily. API testing is mandatory in the Continuous Integration and Continuous Delivery (CI/CD) phases [16]. In modern continuous development, the back and front-end interaction typically happens via methods and functions that the back end offers directly via API. Testing both the front-end and back-end allows for fast application quality feedback and enables the CI/CD pipeline to start automatically and promptly.

Performance testing is a subset of automatic tests and is designed to evaluate an application's behavior under particular load conditions. Effective load testing incorporates diverse scenarios, including peak traffic simulations, sustained operation periods, and irregular usage spikes, to comprehensively assess API resilience across operational conditions that distributed systems commonly encounter [17].

Apache JMeter is a performance testing tool that allows users to perform load tests on various protocols and technologies. It is one of the most widely used open-source tools for performance testing, particularly in the domains of API testing and web applications. The ability to create and execute complex testing scenarios is one of JMeter's most essential features. JMeter tool is multithreaded and can simulate a large number of VUs, enabling the simulation of a heavy load by distributing tests across multiple machines.

JMeter supports integration with external services and tools such as CI/CD pipelines, monitoring systems, and third-party performance analysis platforms. Despite its many advantages, JMeter also has certain limitations, as noted by authors [18, 19]. One of the main disadvantages is its relatively high memory consumption, especially when running large-scale tests or simulating a large number of users.

Additionally, JMeter lacks advanced features, such as real-time monitoring and has a high learning curve for setting up and configuring distributed tests.

Postman is a platform for API development and testing that has emerged as a leading tool for API development, boasting a very user-friendly interface [20]. Postman is not an open-source tool, and it has a paid version. Postman tests can be executed manually using the GUI or run automatically using the Collection Runner or the Newman command-line tool. Additionally, Postman enables collaboration with teammates by organizing, sharing, and communicating work with APIs. According to authors [20, 21], Postman can be used for API performance testing with a desktop application.

Performance tests can be executed using one of four load profiles: Fixed (the maximum number of virtual users is used throughout the test), Ramp-up (VUs gradually increase from initial load to the maximum), Spike (VUs increase from base load to maximum, then drop back to base load), and Peak (VUs increase from base load to maximum, stabilize, and then return to base load).

It is essential to highlight that during performance test execution in Postman, each virtual user runs the requests in the specified order within a repeating loop. Performance test execution can be monitored in real-time through the Postman Summary tab, which provides a summary of performance metrics available in both tabular and graphical forms.

Thus, analyzing the research results in the reviewed author's works [22, 23], it is worth noting that there are some limitations to running performance tests in Postman. Firstly, a limited number of performance runs can be used each month at no additional cost. Secondly, the number of VUs in a performance test depends on available system resources and the collection used for the test.

3 Methodology

Using performance test tools should help improve the chances of achieving performance testing goals (e.g., validating that the system can function under high load, ensuring the reliability and performance of a system, and checking the system's performance under everyday operational situations using the upper bound of performance).

The use of performance test tools and their integration into test automation frameworks can enhance testing performance by increasing test speed and efficiency, improving test accuracy, reducing test maintenance costs, and mitigating risks [24]. However, performance load testing often requires a significant amount of time, running from hours to even days [25].

Performance testing tools, CI/CD instruments and reporting tools can be parts of a test automation framework. They provide developers and QA engineers with feedback on system performance metrics, including response times and resource utilization. Furthermore, there is a growing trend of adopting a shift-left testing approach, combined with CI/CD practices, which implies that testing is done early and frequently throughout the project life cycle [26-29]. Additionally, automated API tests should be simple, fast, and stable [16]. Thus, the speed of performance test tools becomes an essential factor.

In the context of performance testing, response time is closely related to the perceived speed of the system. The lower the response time, the higher the perceived performance. Thus, there is a strong correlation between these characteristics in performance testing tasks. This allows for using response time as one of the leading performance indicators.

The ART is a metric that provides an overview of the general user experience, and it is also important as the ideal baseline response time, such that any lags that should be investigated or considered critical can be identified [30]. However, percentiles provide a more accurate representation of user experience by accounting for the distribution of response times.

This is crucial for identifying performance bottlenecks and ensuring that even the slowest responses are within acceptable limits. Percentiles should be used when focusing on responsiveness for the majority of users, identifying outliers, and ensuring consistent performance across all users. [31].

Percentiles are essential reliability metrics, capturing the threshold below which a given percentage of observations fall [32]. Performance test tools use percentile values, such as the 90th, 95th, and 99th percentiles (P90, P95, and P99, respectively).

SLAs in performance testing specify target metrics, such as response times or error rates [33, 34], typically using percentiles to represent acceptable performance levels. P99 helps identify performance bottlenecks; its high values indicate severe slowdowns affecting a small percentage of users. P95 and P90 values are commonly used in performance teams instead of ART when defining acceptable response times in SLA [35].

When comparing Postman and JMeter test tools, it is helpful to calculate the difference between the percentiles and the ART. If the ART and percentiles values are closer, then response times show little deviation, and there is confidence in the performance of your system.

It may also mean that the tool is more predictable in its performance [36], which is essential for high-load or response-critical applications [37 - 39]. On the other hand, if the ART is better for a tool, it means that, overall, this tool processes requests more quickly. This could be a result of more efficient request processing.

However, it is essential to consider that ART can be affected by extreme values (e.g., very slow requests). Considering factors such as the stability of a test tool, a larger difference between percentile and ART values may indicate that the tool is prone to performance fluctuations.

This may be especially important for testing real-time applications (e.g., gaming, streaming, and mobile applications, or high-traffic web applications as marketplaces) where minor delays can lead to false positive results (the application appears to be more productive than it actually is) and missed issues.

To understand the degree of stability in a test tool's performance, percentage deviation (D) from ART can be calculated using Formula 1. D value reflects the stability of a test tool, where a deviation below 20% indicates better stability:

$$P_{\text{failure}} = f(x_1, x_2, \ldots, x_n), \quad (1)$$

where P_p is the percentile value (e.g., P_{90} is the value of the 90th percentile).

A positive deviation indicates that the percentile value is significantly higher than the ART. This may indicate that some slow queries are contributing to the system's increased load.

A deviation of zero or close to zero indicates that the percentile value is close to the ART, meaning that the responses are relatively stable and slow queries do not significantly impact the system.

The following scale can be used for D values evaluation:

- low percentage deviation (less than 20%) that indicates stable performance of a system;

- moderate percentage deviation (20% - 50%), which suggests that the system has a noticeable number of slow queries that affect performance;
- high percentage deviation (more than 50%), which demonstrates significant performance degradation that impacts overall stability;
- critical deviation (more than 100%) that indicates there are non-optimal queries or unpredictable latencies, which have a significant impact on stability.

Thus, a test tool with a minor deviation between percentile and ART can be selected for performance testing as a more stable option, which is essential for highly loaded systems where predictability needs to be maintained. In contrast, if the testing goal is to reduce overall testing time, then a tool with a lower ART may be more suitable.

To evaluate deviation values (D) for Postman and JMeter test tools, performance testing of Hypertext Transfer Protocol (HTTP) requests was conducted on five public APIs. For each of the five public APIs, four types of HTTP requests were sent: GET, POST, PUT, and DELETE. Additionally, eight different test cases (TCs) were executed for each type of request in both test tools.

The TCs used different types of loads: Ramp Up, Spike, Peak, and Fixed in Postman and corresponding Thread Groups in JMeter, along with the number of virtual users (VU), think time or delay, and test duration.

The following types of loads were used in Postman:

- “Ramp Up” is used for gradual increase and scalability test;
- “Spike” is used for sudden surge and resilience test;
- “Peak” is used for sustained high load and endurance test;
- “Fixed” is used for stable load and baseline test.

“Ramp Up” load type in Postman simulates a gradual increase in traffic over time:

- *Example*: starting with 10 virtual users (VUs) and adding more users every few seconds until reaching 1,000 VUs;
- *Purpose*: tests how the system handles progressive load growth and whether it scales smoothly without errors or degradation;
- *Use case*: mimicking real-world traffic growth during product launches or normal adoption curves.

“Spike” load type in Postman introduces a sudden, extreme surge in traffic within a very short period:

- *Example*: jumping from 50 users to 5,000 users almost instantly;
- *Purpose*: evaluates stability and resilience when the system experiences an unexpected traffic spike;
- *Use case*: Black Friday sales, ticket bookings, flash sales, or viral traffic events.

“Peak” load type in Postman simulates sustained high traffic after a ramp-up period:

- *Example*: traffic ramps up to 2,000 users and then maintains that level for a long duration;

- *Purpose*: tests how the system performs under continuous heavy load without downtime or performance decline;
- *Use case*: streaming platforms during a popular live event, social media surges, or sustained API demand in production.

"Fixed" load type in Postman maintains a constant, steady number of users throughout the test:

- *Example*: running 500 users continuously for 30 minutes;
- *Purpose*: establishes a baseline performance benchmark for latency, error rates, and throughput under normal conditions;
- *Use case*: routine performance validation, regression testing, or comparing infrastructure changes.

ART values and values of the 90th, 95th, and 99th percentiles (P90, P95, and P99, respectively) were collected for each test case (TC) and type of request from performance test reports in both test tools.

Since percentile values were obtained for each test run, P90, P95, and P99 values were collected per test iteration to assess the stability and typical behavior of performance percentiles. To obtain an aggregate representation, we calculated the respective percentiles (e.g., 90th percentile of all P90 values) using *numpy.percentile()* function in Python.

This approach leverages the empirical distribution of performance metrics to identify representative or worst-case scenarios while mitigating the influence of outliers.

Calculation of percentage deviation D values for Postman and JMeter test tools was performed as the next step in evaluating the performance of a test tool's stability.

After that, mean values [40] of ART, P90, P95, P99, and D were calculated for each type of HTTP request, using Formula 2:

$$\text{Mean\_metric} = \frac{\sum_{k=1}^{n} m_k}{n}, \tag{2}$$

where m_k is the value of the performance metric for TC_k;
n is the number of test cases.

In the final phase, the aggregated performance data of ART and corresponding D values were analyzed for each type of HTTP request to assess the degree of deviation and draw conclusions about the stability of the testing tool.

4 Results

Research shows that techniques such as performance and regression testing frequently co-occur, combining enhancements to the application's performance with the absence of regression issues that appear during continuous deployment [41].

The inconsistency of the interaction chain, the independence of services, dynamic and frequent deployments, and the specific challenges of cloud-native

environments can influence the testing of microservices. Effective API testing should aim to strike a balance between thoroughness and resource efficiency, maximizing coverage while minimizing redundancy.

Additionally, Continuous Integration (CI) requires efficient regression testing to ensure software quality without significantly delaying its CI builds. This warrants the need for techniques to reduce regression testing time, such as Test Case Prioritization (TCP) techniques that prioritize the execution of test cases to detect faults as early as possible [42].

Thus, the approach of prioritizing test cases by test tools can be applied in the context of performance testing. To optimize resource utilization, minimize total test execution time, and use tool-specific strengths, a structured decision-making diagram is required.

The development of a decision-making diagram that supports the distribution of test cases between two performance testing tools must consider various factors. For Postman and JMeter test tools, important factors may include the number of VUs, think time, and the type of load planned for the test (e.g., high, medium, low), based on their differences in application during a planned test.

Additionally, statistical data on the performance of test tools need to be analyzed to identify other factors that can influence the decision on how to distribute tests among test tools. Percentile and D values for Postman and JMeter were calculated for each HTTP request method for 8 TCs and then averaged across all 5 APIs, as shown in Tables 1, 2, and 3. Percentile values exceeded ART for both tools, indicating the presence of performance degradation. The most significant values are highlighted in bold. Postman has a better ART and processes requests faster than JMeter.

Table 1

Aggregated mean values of ART for different requests in Postman and JMeter

Method	Tool	ART, sec
GET	Postman	**166,175**
GET	JMeter	**181,65**
POST	Postman	148,100
POST	JMeter	147,175
PUT	Postman	**144,925**
PUT	JMeter	**152,075**
DELETE	Postman	166,225
DELETE	JMeter	163,325

Table 2

Aggregated mean values of percentiles for different requests in Postman and JMeter

Method	Tool	P90, sec	P95, sec	P99, sec
GET	Postman	298,480	392,650	521,656
GET	JMeter	306,320	326,500	496,108
POST	Postman	188,820	219,400	328,572
POST	JMeter	194,040	207,600	385,248
PUT	Postman	193,460	224,800	643,546
PUT	JMeter	234,980	286,100	364,792
DELETE	Postman	200,040	210,690	434,812
DELETE	JMeter	212,300	232,660	581,036

The Postman shows minimal differences in P90 and P95 D for POST (6.855% and 0.112%) and DELETE requests (11.364% and 17.611%), indicating good stability. Moderate and high differences in P90 D and P95 D for GET (26.45% and 68.236%) requests suggest stability issues and performance fluctuations, which impact the user experience. Additionally, JMeter demonstrates better performance in P99 for PUT requests and in all scenarios for GET requests.

Table 3

Aggregated mean values of percentage deviation for different requests in Postman and JMeter

Method	Tool	P90 D, %	P95 D, %	P99 D, %
GET	Postman	**101,240**	158,613	311,005
GET	JMeter	**74,790**	90,376	267,864
POST	Postman	20,233	36,872	106,621
POST	JMeter	27,088	36,760	161,460
PUT	Postman	25,320	43,121	**399,459**
PUT	JMeter	53,631	82,779	**146,312**
DELETE	Postman	16,309	22,629	160,728
DELETE	JMeter	27,673	40,240	229,010

The calculated difference in average response time between Postman and JMeter (deltas) in each of the 8 TCs shows that Postman has higher performance in low to moderate load conditions. On the other hand, JMeter illustrates higher performance under the conditions of high load and request intensity (sharp increase in load to peak, followed by a gradual decrease with a short think time).

Thus, Postman with a lower ART improves response time and efficiency under average loads, but does not ensure stability in the presence of significant outliers. JMeter, with more minor deviations for GET and PUT requests, is more stable and

predictable, which is crucial for testing complex systems under fluctuating load conditions.

Moreover, applying a successful API performance testing strategy should also focus on the longevity and sustainability of the software's organizational structure. To improve such testing, it is critical to optimize developers' collaboration to ensure the proper allocation of responsibility and reduce ineffective communication [43]. As a result, collaboration becomes a crucial factor in performance testing.

n the context of performance testing tools, collaboration can be done in a built-in way, when the test tool itself has an option of sharing test cases, results and collections for API requests through UI and workspaces, or in an external way (using Git, version control or even manual sharing of test cases and test plans). For instance, Postman has a built-in collaboration option, and the external collaboration approach is mainly used in JMeter.

A critical difference between JMeter and Postman is their ability to be integrated into CI/CD pipelines, such as those built with Jenkins. The implementation of CI/CD in performance testing enables integrated, automated, and periodic execution of test processes [44].

It can also quickly respond to changes in parameter values. JMeter offers direct support for such integrations, making it well-suited for automated and continuous performance testing. Currently, Postman’s support for such automation workflows is comparatively limited, particularly in terms of its performance load profiles.

Thus, to address the challenge of efficiently allocating test scenarios across Postman and JMeter test tools, we propose a decision-making diagram based on a rule-based classification of test scenarios, as shown in Figure 1. The diagram incorporates both qualitative and quantitative criteria, serving as a practical guide for QA engineers and developers during the planning phase of performance test execution. Quantitative factors, such as the type of test load (number of VUs and request intensity), and the need to prioritize specific HTTP requests, provide objective benchmarks for selecting the right tool.

In contrast, qualitative aspects such as collaboration features, think-time flexibility, and CI/CD integration capabilities reflect practical considerations that influence QA or developer productivity and tool suitability. By combining these criteria, the diagram provides a balanced framework for allocating performance test scenarios.

The proposed decision-making diagram consists of four levels of decision logic. The diagram is designed to guide the allocation of performance test scenarios between two tools, JMeter and Postman, based on scenario attributes and tool capabilities.

At the first level, the diagram begins with a quantitative parameter that characterizes the test scenario, specifically the type of test load (required number of VUs and the expected request intensity). This parameter determines the initial classification of the scenario based on its load intensity.

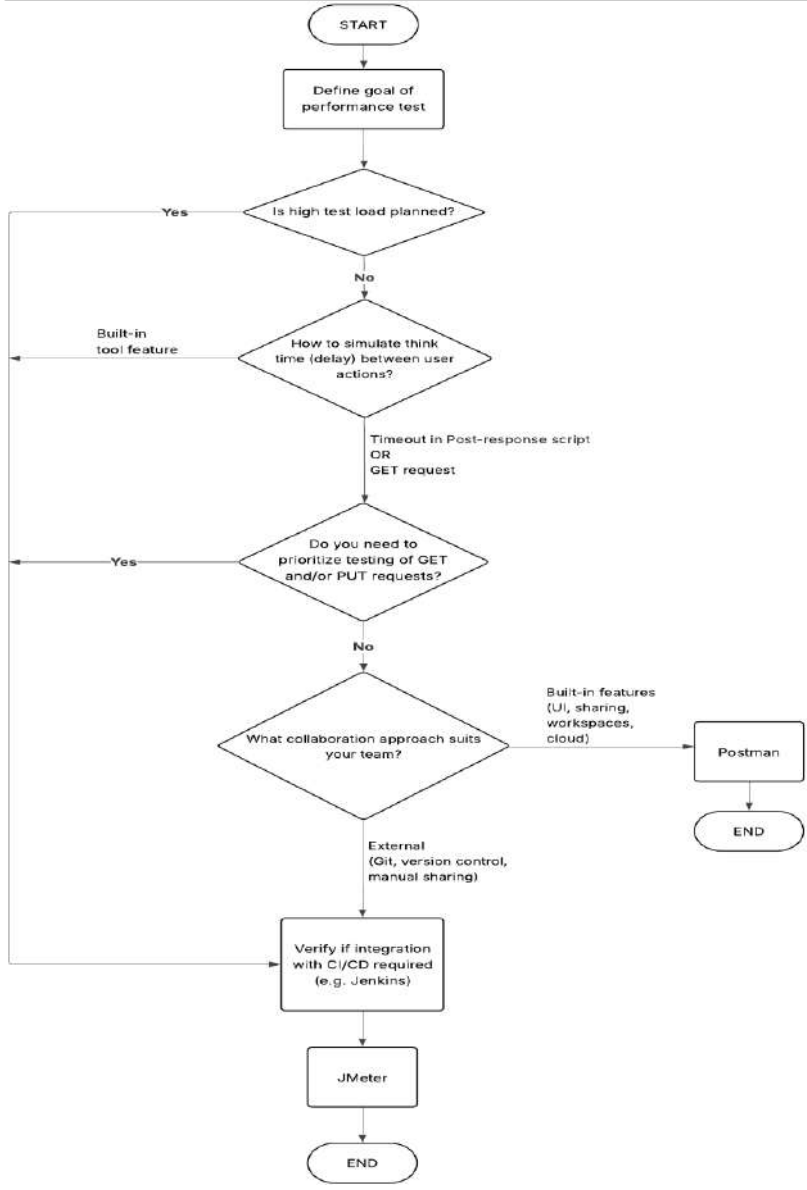


Figure 1. Decision-making diagram of the distribution of tests between test tools

The second level introduces qualitative criteria related to the flexibility of simulating think time. This includes factors such as the ease of implementing delays between user actions, either through built-in, low-overhead mechanisms (e.g., configurable timers) or using manual workarounds, such as post-response timeouts or additional GET requests intended to emulate pauses. This decision branch helps refine tool selection by assessing the functional (e.g., the need for realistic user behavior simulation) and operational requirements (e.g., efficiency, script simplicity, resource consumption (memory and CPU load)) of the test scenario.

At the third level, tool-specific technical considerations are evaluated, including the need for prioritizing specific HTTP requests (e.g., GET and PUT requests). This criterion serves to validate or adjust the preliminary tool assignment by comparing fine-grained compatibility between the scenario and the tool features.

The fourth level shows qualitative criteria related to the execution environment. This includes factors such as the presence of a collaboration feature that refines the tool selection by evaluating the functional requirements of the test scenario.

Each decision node represents a binary or categorical condition (e.g., high load, built-in think time needed, focus is needed for GET and/or PUT requests, collaboration using UI and shared workspaces required), which directs the flow to the following relevant decision node, leading to two terminal outcomes: assignment to JMeter or Postman test tool. This means that during the process of planning and executing performance tests, test cases can be split between two tools, each of which will be used for different types of tests, workloads or tasks, based on their strengths and weaknesses.

The diagram is constructed to be both extendable and interpretable, allowing new criteria to be incorporated without altering the core logic, and ensuring consistency in decision-making during the planning phase of performance testing.

For instance, in a scenario involving a complex API with high concurrency (over 500 simultaneous VUs), the need for realistic think time, and integration with Jenkins, the decision diagram leads to selecting JMeter due to its robust load-handling capabilities and native support for CI/CD pipelines. In contrast, for performance testing a lightweight API involving fewer than 50 users and simple GET/POST requests, with the ability to share results using a UI and cloud, the diagram suggests Postman as the more efficient choice.

The proposed decision-making diagram provides a structured approach for distributing performance test scenarios between two tools, Postman and JMeter, based on defined scenario parameters and tool capabilities. This contributes to reducing subjectivity in tool selection by replacing ad hoc decisions with transparent, criteria-driven logic.

In practical terms, the diagram enables efficient use of both tools: lightweight or low-load scenarios can be quickly executed using Postman. In contrast, more complex or high-load scenarios are directed to JMeter. This selective allocation helps reduce test execution time and unnecessary overhead, improves overall testing efficiency, especially in projects with limited time or resources.

5 Discussions

The proposed decision-making diagram addresses the lack of formal criteria for selecting performance testing tools in practical settings. By providing a structured approach based on scenario characteristics (load, think time requirements, collaboration needs, etc.), the diagram helps reduce subjectivity and improve the efficiency and consistency of test planning.

The developed approach can be helpful in development teams with mixed tool stacks (JMeter and Postman) where decisions on tool usage are often ad hoc. It can be embedded into test management workflows as a decision-support mechanism.

Unlike existing tool comparison matrices or benchmark studies, the proposed diagram integrates context-sensitive decision logic, allowing assessment based on test scenario attributes rather than static tool capabilities.

While the proposed decision-making approach provides a structured and repeatable approach for allocating performance test scenarios between different tools, several limitations should be acknowledged, such as limited tool scope (the approach currently considers only Postman and JMeter tools), the decision logic does not dynamically adjust the weight or priority of individual criteria based on project-specific needs or context, and the diagram does not incorporate empirical performance data (e.g., response time, error rate, CPU/memory usage of test execution) that could further inform the selection process.

6 Conclusions

The developed decision-making approach offers practical value by providing a structured and transparent framework for assigning performance test scenarios to the most suitable tools based on objective criteria. In practice, the selection of performance testing tools is often driven by subjective preferences, prior experience, or ad hoc judgments, which can lead to inconsistent tool usage, suboptimal test configurations, or even inaccurate test results.

The developed approach supports simple standardization of tool selection across teams and projects. This is especially beneficial in collaborative environments, where multiple stakeholders (e.g., QA engineers, DevOps specialists, project managers) participate in test planning and execution. The approach can serve as a reference, reducing ambiguity and supporting decisions aligned with scenario-specific requirements, thereby increasing both the transparency and reproducibility of the test process. Furthermore, the approach enables decision-making at an early stage during test planning, allowing better resource allocation and minimizing the risk of rework caused by choosing inappropriate performance test tools. The decision-making diagram facilitates the distribution of performance test scenarios between Postman and JMeter, enabling the rapid execution of simple tests using lightweight tools and delegating complex or high-load scenarios to more robust solutions, thereby improving overall testing speed and resource utilization.

7 Acknowledgements

The research was supported by the Ukrainian project of fundamental scientific research, "Development of computational methods for detecting objects with near-zero and locally constant motion by optical-electronic devices" (#0124U000259) from 2024 to 2026.

References

1. OctoPerf. (2020). *Statistical analysis in performance testing*. Retrieved from https://blog.octoperf.com/statistical-analysis-in-performance-testing
2. TestRail. (2025). *Performance testing metrics*. Retrieved from https://www.testrail.com/blog/performance-testing-metrics
3. BlazeMeter. (2020). *Key performance metrics*. Retrieved from https://help.blazemeter.com/docs/guide/performance-kpis-key-perf-test-metrics.htm
4. Soldani, J., & Brogi, A. (2022). Anomaly detection and failure root cause analysis in (micro) service-based cloud applications: A survey. *ACM Computing Surveys (CSUR), 55*(3), 1–39. https://doi.org/10.1145/3501297
5. Panahande, M., & Miller, J. (2023). A systematic review on microservice testing. *Research Square*. https://doi.org/10.21203/rs.3.rs-3158138/v1
6. Khader Basha, S., Purimetla, N. R., Roja, D., Vullam, N., Dalavai, L., & Vellela, S. S. (2023). A cloud-based auto-scaling system for virtual resources to back ubiquitous, mobile, real-time healthcare applications. *2023 3rd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, 1223–1230. https://doi.org/10.1109/ICIMIA60377.2023.10426107
7. Batubara, J., Sinta, R., & Panjaitan, F. (2024). Performance analysis of web-based e-commerce information systems using load testing method. *Idea, 2*(1), 18–27.
8. Fournier, Q., Ezzati-jivan, N., Aloise, D., & Dagenais, M. R. (2019). Automatic cause detection of performance problems in web applications. *2019 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)*, 398–405. https://doi.org/10.1109/ISSREW.2019.00102
9. Chadwick, D., et al. (2008). *Using rational performance tester version 7*. IBM.
10. Dhandapani, A. (2025). Automation testing in microservices and cloud-native applications: Strategies and innovations. *Journal of Computer Science and Technology Studies, 7*(3), 826–836. https://doi.org/10.32996/jcsts
11. Pargaonkar, S. (2023). A comprehensive review of performance testing methodologies and best practices: Software quality engineering. *International Journal of Science and Research (IJSR), 12*(8), 2008–2014. https://doi.org/10.21275/SR23822111402
12. Ajiga, D., et al. (2024). Methodologies for developing scalable software frameworks that support growing business needs. *International Journal of Management and Entrepreneurship Research, 6*(8), 2661–2683. https://doi.org/10.51594/ijmer.v6i8.1413

13. Yaroshynskyi, M., Puchko, I., Prymushko, A., Kravtsov, H., & Artemchuk, A. (2025). Investigating the evolution of resilient microservice architectures: A compatibility-driven version orchestration approach. *Digital*. https://doi.org/10.3390/digital5030027

14. Molyneaux, I. (2014). *The art of application performance testing: From strategy to tools*. O'Reilly Media.

15. ISO Standards. (2020). Retrieved from https://iso25000.com/index.php/en/iso-25000-standards/iso-25010

16. Manukonda, K. R. R. (2024). *Optimizing performance: Designing API test automation frameworks*. JEC Publication.

17. Nagineni, S. (2025). Advancing software reliability through systematic API testing: A comparative analysis of modern automation frameworks and methodological implications for distributed systems. *JCSTS, 7*(8), 798–805. https://doi.org/10.32996/jcsts.2025.7.8.94

18. AutomateNow. (2023). *Advantages and disadvantages of using JMeter*. Retrieved from https://automatenow.io/advantages-and-disadvantages-of-using-jmeter

19. TestSigma. (2025). *JMeter vs Postman*. Retrieved from https://testsigma.com/blog/jmeter-vs-postman

20. Westerveld, D. (2024). *API testing and development with Postman: API creation, testing, debugging, and management made easy*. Packt Publishing.

21. Postman. (2024). *Performance test configuration*. Retrieved from https://learning.postman.com/docs/collections/performance-testing/performance-test-configuration

22. NashTech. (2024). *Performance testing with Postman: Is it worth?* Retrieved from https://blog.nashtechglobal.com/performance-testing-with-postman-is-it-worth

23. Susan Rini, V. S. (2024). When Postman goes that extra mile to deliver performance to APIs. *Software Testing Magazine*. Retrieved from https://www.softwaretestingmagazine.com/tools/when-postman-goes-that-extra-mile-to-deliver-performance-to-apis

24. Umar, M. A., & Chen, Z. (2019). A study of automated software testing: Automation tools and frameworks. https://doi.org/10.5281/zenodo.3924795

25. AlGhamdi, H. M., Bezemer, C.-P., Shang, W., Hassan, A. E., & Flora, P. (2023). Towards reducing the time needed for load testing. *Journal of Software: Evolution and Process*. https://doi.org/10.1002/smr.2276

26. Vaddadi, S. A., Thatikonda, R., Padthe, A., et al. (2023). Shift left testing paradigm process implementation for quality of software based on fuzzy. *Soft Computing*. https://doi.org/10.1007/s00500-023-08741-5

27. Mukherjee, R., & Patnaik, K. S. (2021). A survey on different approaches for software test case prioritization. *Journal of King Saud University - Computer and Information Sciences*, 1041–1054. https://doi.org/10.1016/j.jksuci.2018.09.005

28. Parmar, T. (2025). Implementing CI/CD in data engineering: Streamlining data pipelines for reliable and scalable solutions. https://doi.org/10.5281/zenodo.14762683

29. Rani, V. S., Babu, D. A. R., Deepthi, K., & Reddy, V. R. (2023). Shift-left testing in DevOps: A study of benefits, challenges, and best practices. *2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS)*, 1675–1680. https://doi.org/10.1109/ICACRS58579.2023.10404436

30. Lawi, A., Panggabean, B. L. E., & Yoshida, T. (2021). Evaluating GraphQL and REST API services performance in a massive and intensive accessible information system. *Computers*. https://doi.org/10.3390/computers10110138

31. Godinho, A., Rosado, J., Sá, F. A., & Cardoso, F. (2023). Method for evaluating the performance of web-based APIs. *International Conference on Smart Objects and Technologies for Social Good*. https://doi.org/10.1007/978-3-031-52524-7_3

32. Smith, W. (2025). *Nobl9: Service level objectives in practice: The complete guide for developers and engineers*. HiTeX Press.

33. Amazon. (2020). *What is SLA*. Retrieved from https://aws.amazon.com/what-is/service-level-agreement

34. Seifert, M. (2021). Analysis of public cloud service level agreements: An evaluation of leading software as a service providers. *CIISR@Wirtschaftsinformatik*, 22–35.

35. Qian, W., et al. (2025). Learning unified system representations for microservice tail latency prediction.

36. Anderstedt, H., & Wifvesson, M. (2025). Benchmarking and load testing a dynamic CRM architecture (Bachelor's thesis). Lund University, Helsingborg, Sweden.

37. Hendayun, M., Ginanjar, A., & Ihsan, Y. (2023). Analysis of application performance testing using load testing and stress testing methods in API service. *Jurnal Sisfotek Global, 13*(1), 28–34. https://doi.org/10.38101/sisfotek.v13i1.2656

38. Ramu, V. B. (2023). Performance testing and optimization strategies for mobile applications. *International Journal of Performance Testing and Optimization*. https://doi.org/10.14445/22492615/IJPTT-V13I2P401

39. Gorantla, V. K. C. (2021). A hybrid WebSocket-REST approach for scalable real-time API design. *IJETCSIT*, 60–69. https://doi.org/10.63282/3050-9246.IJETCSIT-V2I3P107

40. Savanevych, V., et al. (2023). Mathematical methods for an accurate navigation of the robotic telescopes. *Mathematics, 11*(10), 2246. https://doi.org/10.3390/math11102246

41. Miao, T., Shaafi, A. I., & Song, E. (2025). Systematic mapping study of test generation for microservices: Approaches, challenges, and impact on system quality. *Electronics*. https://doi.org/10.3390/electronics14071397

42. Yaraghi, A. S., Bagherzadeh, M., Kahani, N., & Briand, L. C. (2023). Scalable and accurate test case prioritization in continuous integration contexts.

IEEE Transactions on Software Engineering, 1615–1639. https://doi.org/10.1109/TSE.2022.3184842

43. Li, X., Calefato, F., Lenarduzzi, V., & Taibi, D. (2024). Toward collaboration optimization in microservice projects based on developer personalities. *2024 IEEE 21st International Conference on Software Architecture Companion (ICSA-C)*, 95–99. https://doi.org/10.1109/ICSA-C63560.2024.00024

44. Pratama, M. R., & Kusumo, D. S. (2021). Implementation of continuous integration and continuous delivery (CI/CD) on automatic performance testing. *2021 9th International Conference on Information and Communication Technology (ICoICT)*, 230–235. https://doi.org/10.1109/ICoICT52021.2021.9527496

АНАЛІЗ ПЕРЦЕНТИЛІВ ПРОДУКТИВНОСТІ ДЛЯ ТЕСТУВАННЯ НА ОСНОВІ API

Ph.D. С. Хламов[1][0000-0001-9434-1081], М. Мендієлєва[2][0009-0002-4282-3147],
Ph.D. О. Вовк[3][0000-0001-9072-1634], Т. Трунова[4][0000-0003-2689-2679],
Ю. Тесленко[5][0009-0009-8349-3683]

Харківський національний університет радіоелектроніки, Україна
EMAIL: [1]sergii.khlamov@gmail.com, [2]mariia.mendielieva@nure.ua, [3]oleksandr.vovk@nure.ua, [4]tetiana.trunova@nure.ua, [5]yuliia.teslenko@nure.ua

Анотація. *У статті зосереджено увагу на системному аналізі метрик тестування продуктивності з особливим акцентом на порівняння поведінки двох широко застосовуваних інструментів — Postman та JMeter — у контексті публічних прикладних програмних інтерфейсів (API). Тестування продуктивності API відіграє критично важливу роль в оцінюванні відгукуваності та надійності сучасних програмних систем; однак однією з постійних проблем залишається опора виключно на середній час відгуку. Середні значення легко спотворюються аномальними викидами, які часто маскують суттєві затримки відповіді та призводять до неповної картини продуктивності системи. Щоб подолати це обмеження, у даному дослідженні наголошується на використанні аналізу на основі перцентилів, що забезпечує точніший і орієнтований на користувача показник продуктивності для більшості запитів. Методологія передбачала порівняння середніх значень часу відгуку з розподілами перцентилів, з основною увагою до 90-го, 95-го та 99-го перцентилів. Додатково було обчислено відсоткове відхилення значень перцентилів від середнього, що слугувало мірою стабільності інструментів тестування. Такий підхід дозволив більш надійно оцінити поведінку інструментів за різних навантажень. Експериментальні результати виявили виразні переваги кожного інструмента. Postman продемонстрував кращі результати за середнього рівня навантаження для запитів Create та Delete. Водночас JMeter виявився більш ефективним для операцій Read (GET) та Update, де стабільність і передбачуваність є критичними у складних системах. Для підтримки практичного застосування*

у роботі також запропоновано діаграму прийняття рішень, що допомагає розподіляти тестові сценарії між двома інструментами, зрештою підвищуючи ефективність тестування та забезпечуючи більш надійну оцінку продуктивності API.

Ключові слова: *тестування продуктивності, тестування API, навантажувальне тестування, стабільність системи, Postman, JMeter, аналіз часу відгуку, перцентильні метрики, оцінювання продуктивності, прийняття рішень, тестування програмного забезпечення*

УДК 69:628.4:681.5:004
DOI https://doi.org/10.36059/978-966-397-538-2-14

ІНТЕЛЕКТУАЛЬНА ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ ІНВЕНТАРИЗАЦІЇ ТА ВИКОРИСТАННЯ БУДІВЕЛЬНИХ ВІДХОДІВ ПРИ РЕКОНСТРУКЦІЇ ПОШКОДЖЕНИХ ІНФРАСТРУКТУРНИХ ОБ'ЄКТІВ

Dr.Sci. О. Арсірій[1[0000-0001-8130-9613]], **Ph.D. О. Іванов**[2[0000-0002-8620-974X]], **Ph.D. N. Cudecka-Purina**[3[0000-0002-5736-7730]], **К. Бєляєв**[4[0009-0001-7135-3562]]

[1,2,4]Національний університет «Одеська політехніка», Україна,
[3]BA School of Business and Finance, Riga, Latvia,
[3]EKA University of Applied Sciences, Riga, Latvia
EMAIL: [1]e.arsiriy@gmail.com, [2]lesha.ivanoff@gmail.com,
[3]natalija.cudecka-purina@ba.lv, [4]kirillbelyaev2921@gmail.com

Анотація. *Ця робота розглядає потенціал підвищення ефективності інвентаризації та утилізації відходів від будівництва та демонтажу (ВБД) для створення та застосування як вторинних ресурсів під час відбудови пошкодженої інфраструктури, зокрема об'єктів, зруйнованих війною. Це досягається шляхом розробки інтелектуальної інформаційної технології (ІІТ). Актуальність вирішення цього питання підкреслюється тим, що великомасштабна відбудова пошкодженої інфраструктури вимагає значних витрат вітчизняних та інвестиційних первинних ресурсів. З огляду на неминучий дефіцит цих ресурсів, розвиток ринку вторинної сировини та можливість вилучення цінних матеріалів з потоків відходів для їх ефективного повторного використання стають надзвичайно важливими. Аналіз досвіду європейських колег щодо ефективного впровадження бізнес-моделі циркулярної економіки для подолання обмежень у використанні первинних ресурсів підкреслює необхідність розробки ІІТ для інвентаризації та утилізації ВБД. Попередній досвід розробників вказує на те, що ІІТ може бути реалізована як геоінформаційна система (ГІС), що дозволяє ідентифікувати об'єкти інфраструктури через базу даних. Розроблена база*

даних ГІС містить просторові та атрибутивні дані, які відповідають вимогам бізнес-моделі циркулярної економіки, формуючи основу для концептуальної системи інвентаризації об'єктів. Інтелектуальний компонент ІІТ базується на використанні великих мовних моделей для аналізу зображень місць утворення відходів (Sources). Цей аналіз включає інтерпретацію візуальних ознак, класифікацію типів будівельних матеріалів та надання приблизної оцінки обсягу. Дослідження також представляє рішення для попередньої класифікації центрів прийому відходів (Sinks) на основі атрибутивних та геопросторових даних, а також розробку інтерактивної карти «Sinks-Sources» та визначення ефективних маршрутів для транспортування ВБД, що сприяє створенню вторинних ресурсів.

Ключові слова. *Відходи від будівництва та демонтажу, великі мовні моделі (ВММ), інтелектуальна інформаційна технологія, геоінформаційна система, штучний інтелект, реконструкція.*

1. Постановка проблеми

У сучасній геополітичній ситуації в Україні життєво важливою є масштабна реконструкція пошкоджених інфраструктурних об'єктів, у тому числі зруйнованих війною, що безперечно потребує високих витрат внутрішніх та інвестиційних первинних ресурсів. Враховуючи їх неминучий дефіцит та тенденцію зростання цін на їх споживання, набуває великого значення розвиток ринку вторинної сировини, а також можливість вилучення цінних матеріалів із потоку відходів з метою їх ефективного повторного використання.

Під впливом цих факторів актуальною є розробка інтелектуальної інформаційної технології (ІІТ) інвентаризації та використання відходів від будівництва та демонтажу (ВБД) для підвищення ефективності реконструкції пошкоджених інфраструктурних об'єктів на основі використання вторинних ресурсів. При розробці технології враховано досвід колег із країн Європи щодо розвитку циркулярної економіки при вирішенні проблем обмеженого використання первинних ресурсів [1]; зменшення інвестиційних та транспортних витрат на переробку первинних ресурсів і виробництво будівельних матеріалів; збільшення зайнятості на місцях, в тому числі «зелених» робочих місцях; потенційне зниження викидів CO_2 в процесі реконструкції тощо.

Для ефективного впровадження бізнес-моделі циркулярної економіки вторинних ресурсів пропонується створення геоінформаційної системи (ГІС) інфраструктурних об'єктів з визначенням тих, що потребують реконструкції, у тому числі зруйнованих війною. ГІС міститиме просторові атрибутивні дані, які відповідають запиту бізнес-моделі циркулярної економіки та складають основу для концептуальної системи інвентаризації об'єктів. Інтелектуальна складова технології базується на використанні великих мовних моделей (ВММ) для аналізу зображень об'єктів утворення ВБД та подальшої їх класифікації та центрів прийому.

2. Аналіз попередніх досліджень

У статті [2] обговорюються загальні можливості та виклики використання ВММ, як-от моделі GPT, у будівельній галузі. Хоча вона не присвячена безпосередньо аналізу відходів, стаття створює основу для застосування цих моделей, зокрема для вибору та оптимізації матеріалів, що має відношення до повторного використання відходів. Стаття [3] надає ширший огляд генеративного штучного інтелекту (ШІ), включно з ВММ, у будівництві. Вона підкреслює їхній потенціал для різноманітних завдань, деякі з яких можуть бути адаптовані для управління відходами.

Наступні статті заглиблюються у критичні аспекти аналізу зображень для оцінки пошкоджень та класифікації відходів. У [4] оцінюються моделі генеративного ШІ (які можуть включати принципи ВММ для токенізації зображень) для класифікації структурних пошкоджень на зображеннях після землетрусу. Вони обговорюють використання ШІ для ідентифікації рівнів пошкодження та типів матеріалів. Дослідження [5] зосереджене на використанні нейронних мереж (як-от YOLO та ResNet) для класифікації ВБД за зображеннями. Хоча тут ВММ не використовуються безпосередньо в тому ж сенсі, як для тексту, стаття розглядає аспект аналізу зображень для класифікації відходів, що є ключовою частиною запропонованої технології.

Декілька джерел показують, як ШІ використовується для управління ВБД та впровадження циркулярної економіки. Високорелевантний проєкт [6] присвячений безпосередньо відбудові України. Він демонструє, як ШІ аналізує кадри з дронів, зняті над розбомбленими будівлями, для ідентифікації матеріалів, придатних для повторного використання. Джерело [7] обговорює різні способи, якими ШІ може сприяти скороченню відходів та їх переробці у будівництві, включно з плануванням демонтажу за допомогою ШІ для ідентифікації матеріалів, які можна зберегти. Стаття [8] надає ширше уявлення про те, як ШІ, машинне навчання та обробка природної мови (ОПМ) можуть оптимізувати процеси поводження з відходами, прогнозувати їхню генерацію та ідентифікувати можливості для повторного використання та переробки матеріалів у контексті циркулярної економіки.

Нещодавні досягнення підкреслюють, як ОПМ може використовуватися у будівництві (для текстових даних, що може доповнювати аналіз зображень). У [9] автори використовують ВММ для аналізу текстових даних (наприклад, звітів про нещасні випадки) у сфері будівельної безпеки. Хоча це не стосується безпосередньо відходів, це демонструє силу ВММ в обробці неструктурованого тексту, що потенційно може бути використано для аналізу звітів, специфікацій або інших текстових даних, пов'язаних з будівельними матеріалами та відходами. Огляд [10] надає вичерпний огляд застосувань ОПМ у будівництві, підкреслюючи потенціал для обробки та аналізу текстових даних для досягнення «будівельної інтелектуальності».

Подальші наукові роботи зосереджені на досягненнях у сфері розумних технологій та підходів, що ґрунтуються на даних, для управління ВБД.

Дослідження [11] представляє стратегічну основу, розроблену для сприяння ефективній інтеграції розумних технологій у сферу управління ВБД. В окремому дослідженні [12] було ретельно вивчено класифікацію ВБД, що розглядається як ключовий інструмент для сприяння їх ефективній реінтеграції в матеріальний цикл. Дослідники всебічно вивчили технологічну доцільність процесів переробки та надали рекомендації щодо впровадження новаторських методологій утилізації. Стаття [13] є критичним оглядом різних методів машинного навчання, що використовуються для оцінки, класифікації та прогнозування обсягів ВБД з метою сприяння більш сталому управлінню відходами.

Впровадження ГІС відкриває значні перспективи для покращеного управління міськими відходами, зокрема у дослідженні [14] продемонстровано корисність інформаційного моделювання будівель (Building Information Modeling (BIM-моделей) у цифровому симулюванні фізичних атрибутів будівельних проєктів для цілей планування та управління. У синергії з ГІС цей підхід надає надійні інструменти для вищого рівня просторового та екологічного аналізу. Систематичний огляд літератури [15] досліджує застосування геопросторових технологій та дистанційного зондування в управлінні ВБД. Автори аналізують існуючі дослідження, щоб ідентифікувати ключові сценарії застосування – включно з ідентифікацією відходів, вибором майданчиків, кількісною оцінкою та підтримкою прийняття рішень – та визначити прогалини у дослідженнях для спрямування майбутніх робіт. Крім того, автори [16] розробили автоматизовану систему, здатну точно ідентифікувати географічне розташування ВБД та оптимізувати розподіл ресурсів, необхідних для операцій з демонтажу та подальшого транспортування відходів.

Недавні дослідження показують, що цифрові технології, як-от оцінка супутникових знімків та фотограмметрія, можуть суттєво сприяти точності обліку ВБД, зокрема на територіях, що постраждали від постійних руйнувань. Ці методи дозволяють ідентифікувати не лише обсяги відходів, але й їх геопросторове розташування, що значно полегшує планування логістики для управління вторинними ресурсами [17]. Крім того, інтеграція технологій блокчейну в управління даними дозволяє досягти більш прозорого та надійного відстеження матеріалів, що є однією з передумов для впровадження ефективної циркулярної економіки [18].

Одночасно стає все більш актуальною тема роботизації та автоматичного сортування ВБЗ. Низка досліджень підкреслює, що поєднання комп’ютерного зору з роботизованими маніпуляціями суттєво сприяє економії часу в процесі розділення матеріалів, одночасно знижуючи ризик негативного впливу на людину під час роботи з небезпечними матеріалами або у небезпечних місцях [19]. Алгоритми машинного навчання застосовуються для прогнозування обсягів генерації ВБД у конкретних проєктах, ґрунтуючись на історичних даних та параметрах, специфічних для будівельного процесу [20].

У контексті політики сталого розвитку Європейського Союзу значна роль відводиться синергії технологій BIM та цифрових двійників із ГІС-рішеннями. Вони дозволяють проводити моніторинг у режимі реального часу не лише самого будівельного процесу, а й динаміки генерації та управління відходами [21]. Такий тип інтеграції дозволяє моделювати різні сценарії повторного використання ресурсів та їхній вплив на зменшення викидів CO_2 [22]. Крім того, все частіше досліджується соціальний вимір бізнес-моделей циркулярної економіки – оцінка залучення місцевих громад та розвиток «зелених» робочих місць, що в сукупності веде до підтримки суспільством різних ініціатив з повторного використання та підвищення обізнаності громадськості щодо питань сталого розвитку в ширшому масштабі [23, 24].

3. Постановка мети дослідження

Метою дослідження є підвищення ефективності реконструкції пошкоджених інфраструктурних об'єктів, у тому числі зруйнованих війною, за рахунок розробки інтелектуальної інформаційної технології інвентаризації та утилізації відходів від будівництва та демонтажу для створення та використання вторинних ресурсів. Для створення ІІТ у роботі вирішуються наступні задачі:

- розробка методу автоматизованої інвентаризації ВБД на основі порівняльного аналізу зображень об'єкту до і після руйнування, із використанням ВММ для інтерпретації візуальних ознак, класифікації типів будівельних матеріалів та орієнтовної оцінки їх об'єму;
- розробка концептуальної схеми ІІТ інвентаризації та утилізації ВБД для створення та використання вторинних ресурсів;
- реалізація версії ГІС інвентаризації пошкоджених об'єктів, що використовує базу геопросторових та атрибутивних даних щодо центрів прийому і утилізації ВБД (*Sinks*) та пошкоджених інфраструктурних об'єктів (*Sources*), а також їх взаємозв'язків у вигляді інтерактивної карти для переміщення відходів з метою подальшого отримання вторинних ресурсів.

4. Метод автоматизованої інвентаризації ВБД на основі порівняльного аналізу зображень інфраструктурного об'єкту до і після руйнування

Авторами запропонований метод автоматизованої інвентаризації ВБД на основі порівняльного аналізу зображень інфраструктурного об'єкту до і після руйнування, що складається із наступних етапів.

Етап 1. Отримання зображень інфраструктурного об'єкту до та після пошкодження/руйнування. В якості джерел використовують зображення з дронів, супутників, архівні та приватні зображення, а також результати BIM-моделей, що надають цифрове 3D-представлення будівлі. Для подальшого аналізу рекомендовано використання візуальних пар (до/після), що дозволяє локалізувати зони пошкоджень та ідентифікувати нові об'єкти (відходи) на сцені. Приклад таких зображень для об'єкту «Школа» в форматі .jpg до та після руйнування показано на рис. 1.

Етап 2. Аналіз зображень з метою класифікації відходів (бетон, цегла, скло тощо). Для виконання сегментації зображень (тобто виділення та окреслення конкретних об'єктів або зон на зображенні, таких як руїни, цілі частини будівлі, різні типи сміття), використовують спеціалізовані інструменти машинного зору або інтеграцію з ними. До основних підетапів відносять попередню обробку зображень, сегментацію (кластеризацію) зображення на об'єкти/фрагменти та класифікацію сегментованих об'єктів. Проведення попередньої обробки дозволяє виконати вирівнювання та нормалізацію яскравості/контрасту, визначити області інтересу (Regions of Interest ROI) – фрагменти з відходами, де видно уламки будівлі (руїни, завали), області, де присутні матеріали (бетон, цегла, метал тощо), зони з контрастними змінами після руйнування (відсутні стіни, зсуви даху тощо). Зазначимо, що ROI – це області, які змінилися на зображеннях «після» у порівнянні з зображенням «до» (зникли частини фасаду, з'явився дим або уламки) (рис. 2).

Рисунок 1. Зображення об'єкту «Школа» до та після руйнування

```
{
  "roi_1": {
    "location": [150, 300, 450, 600],
    "material": "brick",
    "confidence": 0.87
  },
  "roi_2": {
    "location": [500, 700, 900, 1100],
    "material": "concrete",
    "confidence": 0.92
  }
}
```


Рисунок 2. Приклад ROI в форматі JSON за зображеннями об'єкту «Школа»

Проведення сегментації об'єктів/фрагментів потребує використання моделей семантичної сегментації (наприклад, U-Net, DeepLabv3+, Mask R-CNN) для виділення окремих шматків будівельних матеріалів за зображеннями. При цьому можливе навчання на власному датасеті з фотографій реальних руйнувань. При класифікації сегментованих об'єктів кожному фрагменту призначається клас, наприклад бетон, цегла, скло, метал, дерево або змішані відходи. Можливо використання моделей типу ResNet, EfficientNet, або візуальні енкодерів з CLIP/BLIP2 для мультимодальних підходів в випадку змішаних відходів. В якості анотованих датасетів використовують існуючі open-source: TrashNet, TACO або власні сегментовані та марковані зображення руйнувань.

Етап 3. Оцінка об'єму відходів. Це ключовий етап для практичного застосування. Після класифікації матеріалів у кожній ROI зображення, враховуючи його масштаб, виконується перетворення піксельних площ у реальні площі. Наприклад, якщо 1 піксель зображення складає 10 см, тоді площа сегмента ROI $S_{roi} = X \times Y \times (0,1 \times 0,1)$ м^2. Для двовимірних зображень ROI робляться припущення щодо товщини шару уламків, наприклад товщина бетонних плит складатиме ~0,2–0,3 м і тоді S_{roi} перераховується зважаючи на товщину. Для отримання більш точних значень оцінок об'єму відходів створюються 3D-реконструкції з кількох зображень, використовуються геометричні методи для точного розрахунку кубатури уламків та порівняння об'ємів об'єкту до і після руйнування. Приклад орієнтовної оцінки об'ємів різних типів будівельних відходів на основі зображень об'єкту «Школа» до та після руйнування показаний в табл. 1.

Таблиця 1

Фрагмент приблизної оцінки обсягів різних видів будівельних відходів на основі зображень об'єкту «Школа»

Тип матеріалу	Площа, м^2	Товщина шару, м	Об'єм, м^3	Приблизна вага, т
Бетон	35	0,25	8,75	~20
Цегла	22	0,3	6,6	~10
Скло	8	0,02	0,16	~0,4

Етап 4. Застосування ВММ як мультиінструмента для генерації пояснень. Необхідно зазначити, що сценарії виконання цього інноваційного етапу залежатимуть від початкової ініціалізації ВММ. Тобто вказати режим використання ВММ – автономний або в комплексі з CV-моделлю, а також яка ВММ використовується – текстова (GPT3.5, Gemini) або мультимодальна (GPT-4o Gemini Pro Vision). Перевагою використання мультимодальної ВММ є можливість одночасного аналізу зображення та тексту, що є корисним при вирішенні задачі автоматизованої інвентаризації ВБД за зображеннями.

Успішне виконання етапів щодо автоматизованої інвентаризації ВБД на основі порівняльного аналізу зображень інфраструктурного об'єкту до і після руйнування залежить від ефективної генерації промпта для ВММ. Необхідно, щоб згенерований промпт мав чітку інструкцію, опис вхідних та вихідних даних, одиниці вимірювання даних, вказівку на рівень деталізації результату та персону (роль) ВММ. Далі наводимо приклад уточненого промпта для Gemini (Multimodal):

«*Призначення промпта: автоматизована інвентаризація будівельних відходів на основі зображень, класифікація матеріалів, оцінка обсягів і можливості повторного використання.*

Промпт:

Нижче наведено чотири зображення:

- перше (Image 1) показує інфраструктурний об'єкт до руйнування;
- три наступні (Image 2, Image 3, Image 4) – з різних ракурсів після руйнування.

Проаналізуй ці зображення разом і виконай наступне.

Визнач зони пошкоджень шляхом візуального порівняння між зображенням «до» та трьома зображеннями «після». Опиши локалізацію і характер пошкоджень (наприклад, обвалення фасаду, руйнування даху тощо).

Ідентифікуй типи матеріалів, які, ймовірно, стали будівельними відходами (наприклад: бетон, цегла, скло, дерево, метал). Зазнач, в яких зонах вони спостерігаються.

Оціни об'єм відходів для кожного матеріалу – у кубометрах або приблизних відсотках загального об'єму (якщо точно неможливо).

Оціни потенціал вторинного використання кожного виду відходів: придатний до переробки/повторного використання; частково придатний; непридатний.

Сформуй структурований інвентаризаційний звіт у табличному або JSON-форматі з такими полями: zone: опис або координати пошкодженої ділянки; material_type: тип матеріалу; estimated_volume: обсяг у $м^3$ або %; reuse_potential: короткий висновок щодо вторинного використання; confidence: рівень впевненості (високий/середній/низький).»

Фрагмент результату відповіді ВММ Gemini у форматі JSON показаний на рис. 3. Повний JSON-файл доступний за посиланням [17].

```
[
  {
    "zone": "Central Facade and Entrance Group (from foundation to roof)",
    "description_of_damage": "Complete collapse of the upper floors above the entr
    "material_type": "Concrete/Reinforced Concrete",
    "estimated_volume": "60–70% of total volume (approx. 2400–2800 m³)",
    "reuse_potential": "Suitable for recycling (secondary crushed stone, aggregate
    "confidence": "High"
  },
  {
    "zone": "Central Facade and Entrance Group (from foundation to roof)",
    "description_of_damage": "Complete collapse of the upper floors above the entr
    "material_type": "Brick/Stone/Plaster",
    "estimated_volume": "15–20% of total volume (approx. 600–800 m³)",
    "reuse_potential": "Partially suitable (crushing for backfill, limited reuse o
    "confidence": "High"
  },
```


Рисунок 3. Фрагмент результату відповіді ВММ в форматі JSON за зображеннями об'єкту «Школа»

Загальний орієнтовний об'єм будівельних відходів для такої чотириповерхової будівлі становить приблизно 3700–4300 м^3. Детальний звіт у табличному форматі через його громіздкість доступний за посиланням [17].

5. Концептуальна схема ІІТ інвентаризації та утилізації ВБД

Отже, можна виділити наступні завдання, які можуть бути вирішені за допомогою ІІТ для аналізу та управління ВБД:

- попередня класифікація центрів прийому відходів на основі атрибутивних даних, а також отримання їхніх геопросторових даних та створення відповідної бази знань *Sinks*;
- автоматизована інвентаризація ВБД на основі порівняльного аналізу зображень інфраструктурного об'єкту до і після руйнування, класифікація типів будівельних матеріалів та орієнтовна оцінка їх об'єму (розділ 4);
- створення відповідної бази знань *Sources* («Джерела») на основі атрибутивних та геопросторових даних про пошкоджені інфраструктурні об'єкти;
- розробка інтерактивної карти *Sinks-Sources* та визначення ефективного маршруту переміщення ВБД з метою створення вторинних ресурсів.

Загальна концептуальна схема ІІТ для інвентаризації та використання ВБД показана на рис. 4.

Рисунок 4. Концептуальна схема ІІТ для інвентаризації та утилізації ВБД

Стає очевидною нагальна потреба у зборі інформації про потоки відходів для їх подальшої категоризації та переробки. Для цього існують 2 основні методології збору даних: традиційний підхід, що передбачає використання структурованих форм для всебічного переліку типів відходів у контексті конкретного проєкту відбудови; та інноваційний підхід, що використовує ШІ для визначення типів та обсягів відходів залежно від характеристик об'єкту відбудови.

Порівняльний аналіз підходів до збору даних. Кожен підхід має свої переваги та обмеження. Опитування користувачів через форми забезпечує високу точність збору даних, але часто обмежене значними витратами часу на ручне введення. Натомість, парадигма ШІ дозволяє пришвидшити збір даних, хоча її точність безпосередньо залежить від якості навчання моделі. Наразі рівень точності ШІ, як правило, не перевищує той, що досягається при ретельному заповненні форм.

У цьому дослідженні ми переважно використовували методологію, орієнтовану на форми, де користувачі вручну вводять обсяги відходів через інтерфейс опитування. Хоча підхід, керований ШІ, демонструє значний потенціал у цій предметній області та потребує подальшого поглибленого вивчення та вдосконалення, його поточні обмеження не дозволяють використовувати його автономно для високоточної інвентаризації.

Виклики у кількісній оцінці відходів на основі нейронних мереж. Використання нейронних мереж для кількісної оцінки ВБД, отриманих з даних, специфічних для об'єктів будівництва, реконструкції чи знесення, є перспективним напрямом для прогнозування обсягів відходів за різними категоріями матеріалів. Однак цей напрям має кілька суттєвих обмежень.

По-перше, потрібна більш деталізована специфікація критеріїв для об'єктів утворення відходів. Ці критерії, хоча й вимагають мінімального введення даних користувачем, матимуть вирішальне значення для точності

прогнозів обсягів відходів. По-друге, значна нестача даних створює серйозну перешкоду. Нейронні мережі потребують великих наборів даних для навчання, збір та підготовка яких вимагають значних підготовчих досліджень. По-третє, поточна точність прогнозування моделей нейронних мереж у цьому контексті залишається порівняно скромною, що потенційно робить цей метод недостатнім для надання надійних, практично застосовних рекомендацій.

Підсумовуючи, хоча цей підхід має свої недоліки, його синергічна інтеграція зі згаданими вище методами класифікації відходів має потенціал для спрощення та покращення результатів аналізу.

Геопросторові дані та оптимізація маршрутів. Результатом роботи ІІТ є геопросторові дані, що стосуються центрів прийому відходів (*Sinks*), представлені у вигляді інтерактивної карти. Надаються рекомендації щодо належних протоколів поводження з усіма класами відходів, визначеними користувачем через веб-форму, з вказівкою оптимального найближчого центру збору для кожного типу відходів.

Майбутні ітерації цього ІТ-рішення включатимуть можливості для пропонування транспортної логістики та визначення оптимальних маршрутів для перевезення відходів до центрів збору, використовуючи ГІС.

Завдання ідентифікації оптимального транспортного маршруту в межах цієї проблеми є нетривіальним. По-перше, може існувати не один маршрут через великі обсяги відходів, що потребують транспортування, та різноманітність типів відходів, які необхідно доставляти до різних центрів збору. По-друге, як зазначалося раніше, потрібно вирішити проблему розрахунку навантаження для кожного транспортного засобу. Додаткові параметри оптимізації можуть включати сценарії, де початкова точка маршруту – не сам об’єкт будівництва, реконструкції чи зносу, а початкові місця базування транспортних засобів.

6. Впровадження версії ГІС «*Sinks-Sources*» для інвентаризації пошкоджених об’єктів інфраструктури

Була спроєктована та розроблена ГІС-версія для комплексного аналізу та інвентаризації ВБД. Ця система складається з 4 основних модулів: 3 з них були ретельно розроблені нашою командою: фронтенд, сервер та база даних, – а четвертий модуль інтегрує запити до OpenStreetMap для візуалізації карти в інтерфейсі користувача (рис. 5).

Архітектура системи та технології. Серверна частина була спроєктована з використанням мови програмування Java на базі Java Development Kit 21.0.8 з використанням надійного фреймворку Spring Framework, включно зі Spring Boot 3.3.0 та Spring Data JPA. Для системи управління базами даних було обрано PostgreSQL 17.0 зі стратегічним включенням PostGIS для розширення геопросторових можливостей у майбутніх розробках. Фронтенд-інтерфейс був створений за допомогою Thymeleaf 3.1.3, доповненого стандартними вебтехнологіями, такими як HTML5, CSS3 та JavaScript ES14.

Сервер забезпечує зв'язок з веб-застосунком користувача через протокол HTTP, де вхідні запити обробляються контролерами, а потім керуються службами. Взаємодія між сервером та базою даних організована через Hibernate – бібліотеку, призначену для з'єднання Java-програм з базами даних.

Клас BuildingController відіграє ключову роль у збереженні даних для об'єктів та в аналізі зібраних даних для надання практичних рекомендацій. Одночасно клас CenterService відповідає за ідентифікацію всіх центрів збору відходів, які відповідають критеріям, визначеним користувачем [26].

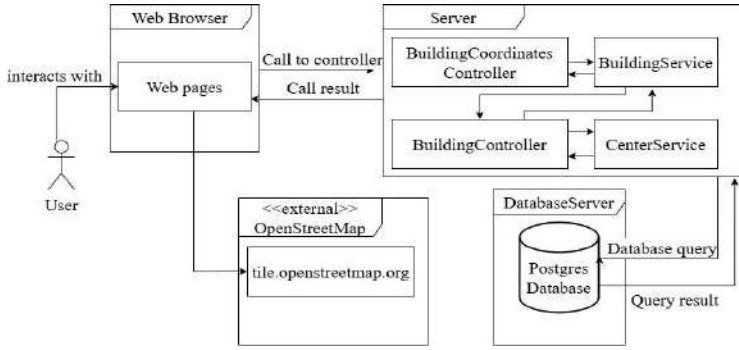


Рисунок 5. Логічна схема ГІС інвентаризації ВБД [26]

Функції підтримки користувача та управління відходами. У поточній версії ця прикладна ІІТ має на меті допомогти користувачам в управлінні ВБД, надаючи відібраний перелік відповідних центрів збору відходів. Ці центри, класифіковані за типом, ідентифікуються на основі їхньої близькості до будівельного майданчика та їхньої здатності приймати хоча б один з типів відходів, зазначених користувачем у вхідній формі. При першому вході в систему користувачам пропонується вказати географічне розташування свого будівельного майданчика на інтерактивній карті (рис. 6).

Після завершення першого кроку, користувач переходить до наступного інтерфейсу, натискаючи кнопку «Далі», розташовану внизу екрана. Друга сторінка вимагає введення детальної інформації, що стосується будівельного майданчика. Перше завдання користувача – вказати тип об'єкту, де відбуваються будівельні роботи (рис. 7). За цим слідує вибір характеру робіт, які призведуть до утворення відходів. Обидва параметри: тип об'єкту та тип робіт – вибираються зі спадних меню, що дозволяє користувачеві обрати один відповідний варіант із заздалегідь визначеного списку. Для класифікації проєктів будівництва, реконструкції або знесення користувачам пропонується вибрати із заздалегідь визначених категорій, що включають квартири, приватні будинки, промислові будівлі та об'єкти інфраструктури. Аналогічно,

для характеру робіт, що призводять до утворення відходів, доступні варіанти: будівництво, знесення, ремонт або «інше» для різноманітних робіт. Обидва запити вимагають вибору лише одного варіанту з відповідних спадних списків, що забезпечує стандартизацію введення даних.

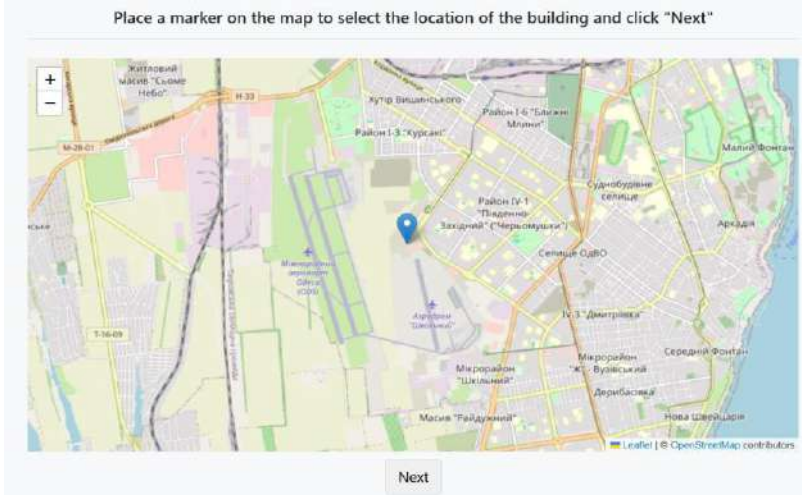


Рисунок 6. Екранна форма для вибору місця будівництва, реконструкції або зносу [26]

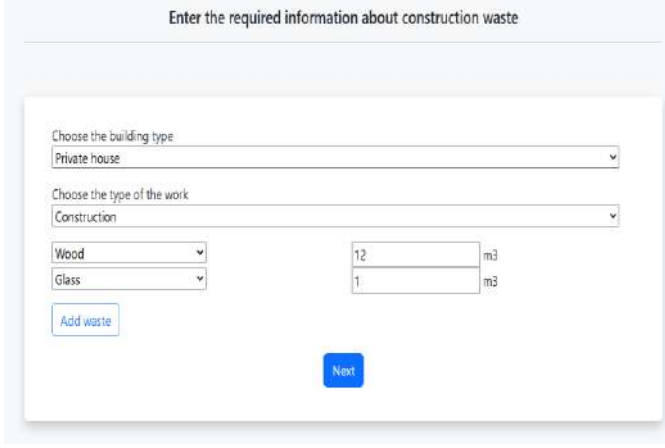


Рисунок 7. Екранна форма для введення даних про місце утворення відходів та даних про утворені ВБД [26]

Після цього користувачі повинні вказати типи відходів та їх відповідні обсяги в кубічних метрах. Вибір типу відходів здійснюється за допомогою спадного списку з одним варіантом, що сприяє уніфікації даних. Натомість, обсяг відходів є обов'язковим числовим полем для введення. Для випадків, що включають кілька потоків відходів, система надає кнопку «Додати відходи», яка динамічно генерує додаткові рядки для введення. Кожен новий доданий рядок вимагає незалежного зазначення типу та обсягу відходів, що забезпечує повну документацію всіх утворених відходів [26].

Заздалегідь визначені категорії відходів, доступні для вибору, включають: бетон і кладка; дерево; метал; гіпсокартон; асфальт і покрівельні матеріали; скло; пластик; двері, вікна та сантехнічне обладнання; небезпечні матеріали; інше.

Після завершення введення даних про відходи користувачі переходять на третю сторінку, натиснувши кнопку «Далі». Ця сторінка представляє табличний огляд найближчих центрів збору ВБД (рис. 8). Таблиця динамічно показує, які типи відходів приймає кожен центр, на основі раніше введених користувачем даних, відмічаючи відповідні клітинки.

Рисунок 8. Екранна форма таблиці пунктів збору відходів, найближчих до місця утворення відходів [26]

Для оптимізації логістики утилізації відходів система застосовує максимально допустимі порогові значення відстані для різних типів об'єктів збору відходів відносно будівельного майданчика: центри збору відходів: включаються, якщо знаходяться в радіусі до 5 км; центри переробки: включаються, якщо знаходяться в радіусі до 10 км; полігони для захоронення: включаються, якщо знаходяться в радіусі до 20 км.

Для розрахунків відстані була використана метрика відстані Чебишова між джерелами відходів A_i та об'єктами їх збору B_j з відповідними координатами (x_i, y_i):

$$d(A, B) = max|x_i - y_i|$$

Це було зроблено для підвищення обчислювальної ефективності, оскільки точність, яку пропонують альтернативні метрики, була визнана непотрібною для цього конкретного завдання.

Після перегляду табличних даних користувачі можуть вибрати «Показати карту», що переводить їх на наступну сторінку, де відображається інтерактивна карта з маркерами різного кольору для різних об'єктів збору відходів (рис. 9).

Символіка маркерів наступна: зелений маркер: позначає місце розташування об'єкту будівництва, реконструкції або знесення; синій маркер: позначає центр переробки відходів; червоний маркер: вказує на полігон для захоронення, призначений для утилізації відходів; помаранчевий маркер: позначає загальний центр збору відходів.

Умовні позначення, що пояснюють ці кольори, доступні внизу екрану. Для зручності, при наведенні курсора на маркер центру збору відходів з'являється підказка, що відображає відповідні деталі об'єкту, включно з назвою, адресою, контактним номером та списком прийнятих матеріалів.

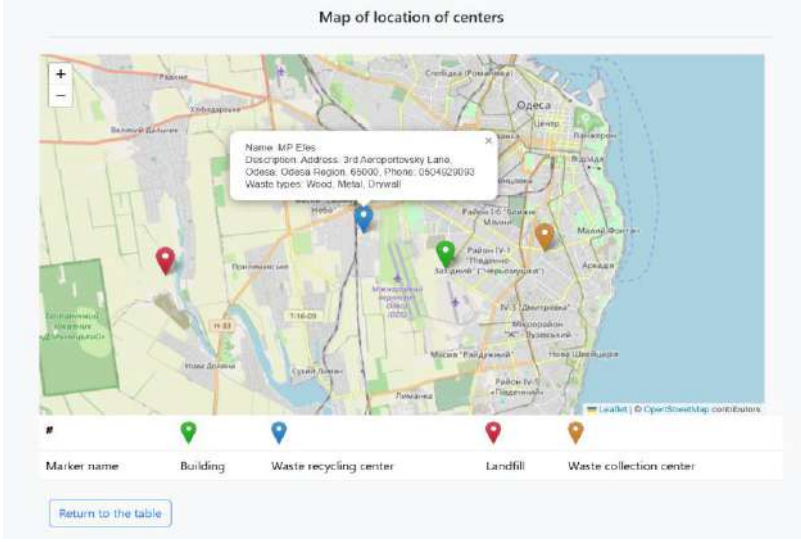


Рисунок 9. Екранна форма для геопросторових даних про пункти збору відходів, найближчі до місця утворення відходів [26]

Важливо, що система фільтрує відображувані центри, гарантуючи, що на карті з'являться лише ті, які здатні приймати хоча б один з типів відходів, зазначених користувачем, навіть якщо інші центри знаходяться географічно ближче. Користувачі мають можливість повернутися до табличного вигляду, натиснувши кнопку «Повернутися до таблиці». Слід зазначити, що дані центрів

збору відходів, представлені на рис. 8 і 9, є ілюстративними тестовими даними [26].

7. Висновки та перспективи

Дослідження спрямоване на підвищення ефективності інвентаризації та утилізації ВБД для створення та застосування вторинних ресурсів під час відбудови пошкодженої інфраструктури, включно з об'єктами, зруйнованими війною, шляхом розробки відповідної ІІТ. Необхідність створення ІІТ, заснованої на моделі циркулярної економіки, підкреслюється тим, що великомасштабна відбудова пошкодженої інфраструктури вимагає значних витрат вітчизняних та інвестиційних первинних ресурсів. Отже, можливість ідентифікувати та вилучати цінні матеріали з потоків ВБД для їх ефективного повторного використання стає надзвичайно важливою.

Інтелектуальний компонент ІІТ ґрунтується на розробці автоматизованого методу інвентаризації ВБД, заснованого на порівняльному аналізі зображень об'єкту до та після руйнування. Після отримання зображень об'єкту інфраструктури з дронів, супутників, архівних джерел, приватних фотографій та результатів BIM, вони сегментуються на області інтересу ROI. Це включає ідентифікацію та окреслення руїн, уцілілих частин будівлі, різних типів відходів тощо, з використанням спеціалізованих інструментів машинного зору. Після класифікації матеріалів у кожній ROI зображення, з урахуванням масштабу, площі в пікселях перетворюються на реальні площі. Попередньо навчена ВММ потім використовується як мультиінструмент для генерації пояснень щодо автоматизованої інвентаризації ВБД.

ГІС-версія для інвентаризації пошкоджених об'єктів була реалізована як веб-застосунок, що дозволяє користувачам визначати джерела утворення відходів та оцінювати їх приблизні обсяги. Ця ГІС використовує базу геопросторових та атрибутивних даних щодо центрів прийому та утилізації ВБД, а також пошкоджених об'єктів інфраструктури та їх взаємозв'язків. Результати представлені у вигляді інтерактивної карти для навігації рухом відходів, що має на меті сприяти відновленню вторинних ресурсів. Маршрутизація відходів враховує тип ВБД та тип центру прийому відходів (відображаються на карті як маркери різних кольорів), надаючи запропоновані маршрути доставки відходів.

Переваги та практичні наслідки дослідження. Серед переваг цього дослідження – використання передових технологій, які дозволяють попередньо, хоча й приблизно, оцінити типи та обсяги відходів зі зруйнованого об'єкту інфраструктури. Також система надає потенційні варіанти поводження з відходами та центри прийому, інтегровані з інтерактивною картою. Запропонований метод, який ґрунтується на порівняльному аналізі зображень «до» і «після» з використанням ВММ для візуальної інтерпретації, демонструє значний потенціал для швидкої та ефективної класифікації матеріалів та оцінки обсягів відходів.

Для посилення розроблених висновків було б корисно ідентифікувати та виділити практичні наслідки дослідження, як на регіональному, так і на міжнародному рівнях. Розроблена ІІТ не лише сприяє більш ефективному застосуванню великих даних у післявоєнному відновленні України, але й має реальний потенціал стати універсальним інструментом для подолання криз подібного характеру або тих, що виникають внаслідок різних природних катастроф у різних частинах світу. Цей підхід має сприяти зменшенню залежності від первинних ресурсів, просуванню принципів циркулярної економіки та скороченню викидів CO_2, які в іншому випадку були б утворені при виробництві нових будівельних матеріалів. Таким чином, це дослідження робить суттєвий внесок у розвиток сталих будівельних і відновлювальних практик та пом'якшення наслідків зміни клімату.

Важливо також підкреслити відтворюваність розробленої методології. Система базується на технологіях, доступних для широкого кола користувачів – супутникових і дронних знімках, моделях комп'ютерного зору з відкритим кодом, ВММ та ГІС-інструментах. Це означає, що з відносно незначними адаптаціями це рішення може бути застосоване до різних географічних областей та будівельних контекстів. Крім того, відтворюваність дає можливість накопичувати порівняльні дані протягом тривалого часу, що є важливим для розробки міжнародних стандартів обліку та повторного використання будівельних відходів від руйнувань. Така перспектива значно розширює практичне застосування дослідження та посилює його актуальність у науковому та політичному дискурсі.

Обмеження та майбутні напрямки досліджень. Однак, обмеження цього дослідження включають його попередній характер, що представляє перше наближення реалізації ІІТ. Воно не враховує повної різноманітності та складності класифікації ВБД або всіх можливих сценаріїв утилізації. Крім того, аналіз пошкоджених об'єктів за допомогою ВММ надає приблизну оцінку з певною похибкою. Поточна робота також не враховує витрати на транспортування до відповідних центрів, що може зробити певні види утилізації відходів економічно недоцільними.

Хоча повна валідація з використанням масштабних реальних даних не була проведена в рамках поточного дослідження, приблизні розрахунки, отримані за допомогою методу, показали 87 % відповідність у порівнянні з узагальненими експертними оцінками, наданими Одеською державною академією будівництва та архітектури. Цей результат підтверджує, що розроблена технологія може слугувати надійним інструментом для попередньої інвентаризації, оптимізації логістики та сприяння переходу до циркулярної економіки.

Майбутні напрямки досліджень включають більш детальне вивчення ВБД із залученням відповідних експертів з будівельної галузі та екологів для доопрацювання розробленої ІІТ та покращення її якості. Також буде зроблено акцент на розширенні бази даних як для навчання ВММ, так і для доповнення даних про існуючі та нові пункти збору ВБД. Крім того, надзвичайно важливо

досягти повноцінного складання ІІТ згідно з концептуальною схемою, що охоплює всі завдання, які ІІТ покликана вирішувати.

Література

1. Cudecka-Purina, N., Kuzmina, J., Butkevics, J., Olena, A., Ivanov, O., & Atstaja, D. (2024). A comprehensive review on construction and demolition waste management practices and assessment of this waste flow for future valorization via energy recovery and industrial symbiosis. *Energies, 17*(21), 5506. https://doi.org/10.3390/en17215506.

2. Saka, A., Taiwo, R., Saka, N., Salami, B. A., Ajayi, S., Akande, K., & Kazemi, H. (2024). GPT models in construction industry: Opportunities, limitations, and a use case validation. *Developments in the Built Environment, 17*, 100300. https://doi.org/10.1016/j.dibe.2023.100300.

3. Ghimire, P., Kim, K., & Acharya, M. (2024). Opportunities and challenges of generative ai in construction industry: Focusing on adoption of text-based models. *Buildings, 14*(1), 220. https://doi.org/10.3390/buildings14010220.

4. Estêvão, J. M. C. (2024). Effectiveness of generative ai for post-earthquake damage assessment. *Buildings, 14*(10), 3255. https://doi.org/10.3390/buildings14103255.

5. Molchanova, M., Didur, V., Mazurets, O., Sobko, O., & Zakharkevich, O. (2025). Method for construction and demolition waste classification using two-factor neural network image analysis. In N. Shakhovska, A. T. Augousti, S. Liaskovska, & O. Duran (Eds.), *Proceedings of the 2nd International Conference on Smart Automation & Robotics for Future Industry (SMARTINDUSTRY 2025)* (Vol. 3970, pp. 168–182). CEUR. https://ceur-ws.org/Vol-3970/PAPER14.pdf.

6. Aouf, R. S. (2025, July 7). *Circularity on the Edge AI finds reusable materials in rubble of Ukrainian buildings*. Dezeen. https://www.dezeen.com/2025/07/07/circularity-on-the-edge-ai-ukraine/.

7. StruxHub. (2024, January 30). *AI in Construction: Reducing Waste & Promoting Recycling*. https://struxhub.com/blog/top-15-ways-ai-and-digital-construction-management-tools-improve-waste-reduction-and-recycling-in-commercial-construction/.

8. Alimi, K., Jin, R., Nguyen, B. N., Nguyen, Q., Chen, W., & Hosking, L. (2025) Exploring artificial intelligence applications in construction and demolition waste management: a review of existing literature, *Journal of Science and Transport Technology*, *5*, 104–136. https://doi.org/10.58845/jstt.utt.2025.en.5.1.104-136.

9. Ahmadi, E., Muley, S., & Wang, C. (2025). Automatic construction accident report analysis using large language models (LLMs). *Journal of Intelligent Construction*, *3*(1), 1–10. https://doi.org/10.26599/JIC.2024.9180039.

10. Ding, Y., Ma, J., & Luo, X. (2022). Applications of natural language processing in construction. *Automation in Construction, 136*, 104169. https://doi.org/10.1016/j.autcon.2022.104169.

11. Wu, Z., Pei, T., Bao, Z., Ng, S. T., Lu, G., & Chen, K. (2025). Utilizing intelligent technologies in construction and demolition waste management: From a

systematic review to an implementation framework. *Frontiers of Engineering Management, 12*(1), 1–23. https://doi.org/10.1007/s42524-024-0144-4.

12. Shuvaev, A. A. (2023). Tools for the involvement of construction and demolition waste in the repeated production cycle. *Science and Transport Progress, 4*(104), 48–55. https://doi.org/10.15802/stp2023/297521.

13. Samal, C. G., Biswal, D. R., Udgata, G., & Pradhan, S. K. (2025). Estimation, classification, and prediction of construction and demolition waste using machine learning for sustainable waste management: A critical review. *Construction Materials, 5*(1), 10. https://doi.org/10.3390/constrmater5010010.

14. Zawada, K., Mikołaj Donderewicz, Agnieszka Gertner, & Kinga Rybak-Niedziółka. (2024). The impact of BIM and GIS on the efficiency of implementing construction projects. *Acta Scientiarum Polonorum. Architectura, 23*, 358–368. https://doi.org/10.22630/ASPA.2024.23.28.

15. Bao, Z., Li, S., Chen, Y., Xie, H., Long, W., & Chen, W. (2025). Applications of geospatial technologies for construction and demolition waste management: A systematic literature review. *Journal of Industrial Ecology, 29*(1), 279–297. https://doi.org/10.1111/jiec.13606.

16. Marzouk, M., Othman, E., & Metawie, M. (2024). Managing demolition wastes using GIS and optimization techniques. *Cleaner Engineering and Technology, 23*, 100852. https://doi.org/10.1016/j.clet.2024.100852.

17. Saka, A., Taiwo, R., Saka, N., Oluleye, B., Dauda, J., & Akanbi, L. (2024). *Integrated bim and machine learning system for circularity prediction of construction demolition waste* (No. arXiv:2407.14847). arXiv. https://doi.org/10.48550/arXiv.2407.14847.

18. Wu, L., Lu, W., Peng, Z., & Webster, C. (2023). A blockchain non-fungible token-enabled 'passport' for construction waste material cross-jurisdictional trading. *Automation in Construction, 149*, 104783. https://doi.org/10.1016/j.autcon.2023.104783.

19. Li, X., Lu, W., Xue, F., Wu, L., Zhao, R., Lou, J., & Xu, J. (2022). Blockchain-enabled iot-bim platform for supply chain management in modular construction. *Journal of Construction Engineering and Management, 148*(2), 04021195. https://doi.org/10.1061/(ASCE)CO.1943-7862.0002229.

20. Xu, Y., Chi, M., Chong, H.-Y., Lee, C.-Y., & Chen, K. (2024). When BIM meets blockchain: A mixed-methods literature review. *Journal of Civil Engineering and Management, 30*(7), 646–669. https://doi.org/10.3846/jcem.2024.21638.

21. Shojaei, A., Ketabi, R., Razkenari, M., Hakim, H., & Wang, J. (2021). Enabling a circular economy in the built environment sector through blockchain technology. *Journal of Cleaner Production, 294*, 126352. https://doi.org/10.1016/j.jclepro.2021.126352.

22. Cha, G.-W., Moon, H. J., Kim, Y.-M., Hong, W.-H., Hwang, J.-H., Park, W.-J., & Kim, Y.-C. (2020). Development of a prediction model for demolition waste generation using a random forest algorithm based on small datasets. *International Journal of Environmental Research and Public Health, 17*(19), 6997. https://doi.org/10.3390/ijerph17196997.

23. Yu, L., & Han, K. (2024). *Using construction waste hauling trucks' GPS data to classify earthwork-related locations: A Chengdu case study* (No. arXiv:2402.14698). arXiv. https://doi.org/10.48550/arXiv.2402.14698.

24. Bradley, P., Whittard, D., Green, E., Brooks, I., & Hanna, R. (2025). Empirical research on green jobs: A review and reflection with practitioners. *Sustainable Futures, 9*, 100527. https://doi.org/10.1016/j.sftr.2025.100527.

25. Arsirii, O., Ivanov, O., Bieliaiev, K., & Cudecka-Purina, N. (2025). *Construction & demolition waste AI analysis report*. https://drive.google.com/drive/folders/1jlurYWZrWomnhfOtOkNdQUXw-iwKyJ68?usp=sharing.

26. Arsirii, O. O., Cudecka-Purina, N., Ivanov, O. V., & Bieliaiev, K. O. (2025). The intelligent information technology for construction waste analysis and management. *Herald of Advanced Information Technology, 8*(1), 87–99. https://doi.org/10.15276/hait.08.2025.6.

INTELLIGENT INFORMATION TECHNOLOGY FOR INVENTORY AND UTILIZATION OF CONSTRUCTION AND DEMOLITION WASTE FROM DAMAGED INFRASTRUCTURE FACILITIES

Dr.Sci. O. Arsiriy[1][0000-0001-8130-9613], **Ph.D. O. Ivanov**[2][0000-0002-8620-974X], **Ph.D. N. Cudecka-Purina**[3][0000-0002-5736-7730], **K. Belyaev**[4][0009-0001-7135-3562]

[124]Odesa Polytechnic National University, Ukraine
[3]BA School of Business and Finance, Riga, Latvia
[3]EKA University of Applied Sciences, Riga, Latvia
EMAIL: [1]e.arsiriy@gmail.com, [2]lesha.ivanoff@gmail.com, [3]natalija.cudecka-purina@ba.lv, [4]kirillbelyaev2921@gmail.com

Abstract. *This work examines the potential for enhancing the efficiency of inventory management and utilization of construction and demolition (C&D) waste to create and apply secondary resources during the reconstruction of damaged infrastructure, including objects destroyed by war. This is achieved through the development of an intelligent information technology (IIT). The urgency of addressing this issue is underscored by the fact that large-scale reconstruction of damaged infrastructure demands significant expenditures of domestic and investment-based primary resources. Given the inevitable scarcity of these resources, the development of a secondary raw material market and the ability to extract valuable materials from waste streams for effective reuse become critically important. An analysis of European colleagues' experiences in effectively implementing a circular economy business model to address the limitations of primary resource utilization highlights the necessity of developing IIT for C&D waste inventory and utilization. Prior developer experience indicates that IIT can be implemented as a geoinformation system (GIS), enabling the identification of*

infrastructure objects through a knowledge base. The developed GIS database contains spatial and attribute data that align with the requirements of the circular economy business model, forming the foundation for a conceptual object inventory system. The intelligent component of the IIT is based on the use of large language models to analyze images of waste generation sites (Sources). This analysis involves interpreting visual cues, classifying types of construction materials, and providing an approximate volume assessment. The work also presents solutions for the preliminary classification of waste reception centers (Sinks) based on attribute and geospatial data, along with the development of an interactive "Sinks-Sources" map and the determination of efficient routes for transporting C&D waste to facilitate the creation of secondary resources.

Keywords. *Construction and demolition waste, LLMs, intelligent information technology, geoinformation system, artificial intelligence, reconstruction.*

УДК 004.8:681.5:697
DOI https://doi.org/10.36059/978-966-397-538-2-15

ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ КЕРУВАННЯ МІКРОКЛІМАТОМ У РОЗУМНИХ СЕРЕДОВИЩАХ
INTELLIGENT MICROCLIMATE CONTROL SYSTEMS IN SMART ENVIRONMENTS

Dr.Sci. Н. Аксак[1][0000-0001-8372-8432], **Ph.D. М.Кушнарьов**[2] [0000-0002-3772-3195], **Ю. Шеліхов**[3][0009-0009-8970-6571]
Харківський національний університет радіоелектроніки, Україна
EMAIL: [1]nataliia.axak@nure.ua, [2]maksym.kushnarov@nure.ua, [3]yurii.shelikhov@nure.ua

Анотація. *Робота присвячена розробленню та впровадженню інтелектуальних систем керування мікрокліматом у «розумних» середовищах – від житлових і громадських будівель до замкнених виробничих приміщень та сіті-ферм. Запропоновано узагальнену трирівневу архітектуру Edge–Transport–Cloud з подієвою шиною, сховищами часових рядів і HMI, а також мультиагентну модель, що формалізує ролі агентів (клімат, пристрої, безпека, сповіщення) і інваріанти безпеки. Розглянуто методи ШІ для прогнозування та оптимізації: від supervised ML/DL (прогноз станів, оцінка VPD/PMV) до MPC і навчання з підкріпленням (RL), зокрема Q-learning для вироблення політик керування у змінних умовах. Окрема увага приділена інтелектуальному керуванню вентиляцією (DCV), енергомоделям і критеріям якості (точність регулювання, частка часу в «зелених» коридорах, енерговитрати, надійність). Наведено кейси для smart-home середовищ, замкнених приміщень і сіті-ферм, показано інтеграцію моделей у*

мікросервісну IoT-платформу та практичні аспекти впровадження: калібрування сенсорів, «safety wrappers», захист каналу (TLS) і роль оператора. Це дослідження формує цілісний місток між теорією і практикою, демонструючи, як поєднання IoT-інфраструктури та методів ШІ забезпечує стійкий комфорт і ощадність ресурсів.

Ключові слова. *Інтелектуальні системи; IoT; хмарні обчислення; мультиагентні системи; агенти; машине навчання, розподілені обчислення; Q-learning; контрольоване середовище; мікроклімат.*

Вступ

Мета роботи - сформувати методологію інтелектуальних систем керування мікрокліматом у «розумних» середовищах, що поєднує моделі об'єкта, архітектуру Edge–Transport–Cloud, мультиагентну координацію та методи ШІ (MPC, Q-learning). Об'єкт дослідження — процеси регулювання мікроклімату в будівлях і агросередовищах; предмет — методи, моделі й інженерні практики, що забезпечують підтримання цільових профілів за умов безпеки й мінімізації витрат ресурсів.

Наукова новизна полягає у: архітектурі Edge–Transport–Cloud із подієвою шиною та розділенням «policies ↔ actuation»; мультиагентній моделі з LTL-інваріантами й «safety wrapper»; інтеграції прогнозних моделей (ANN/LSTM) з MPC та RL, включно з безпечними режимами RL та гібридом RL→MPC; уніфікованому наборі KPI й експериментальному дизайні для порівняння стратегій.

Методологічну основу складають теорія керування, ML для прогнозу часових рядів, RL для синтезу політик і статистичні методи оцінювання. Інженерний апарат охоплює MQTT/HTTP/LoRaWAN, TSDB-сховища, HMI та засоби кіберзахисту.

Практична значущість підтверджується кейсами: smart-home системи, замкнуті приміщення, сіті-ферми з автоматизованим зрошенням і DCV-вентиляцією. Структурно робота переходить від теорії та мультиагентної моделі (розд. 1–2) до архітектур (розд. 3), застосувань у CEA (розд. 4), інтеграції ШІ (розд. 5), вентиляції (розд. 6), Q-learning (розд. 7) та дорожньої карти впровадження (розд. 8).

1.Теоретичні основи інтелектуального керування мікрокліматом

Розумні се редовища це керовані простори (житлові/офісні будівлі, теплиці, сіті-ферми), в яких параметри середовища (температура *T*, відносна вологість *RH*, концентрація CO_2, освітленість *L*, швидкість повітря *v*) вимірюються сенсорами і регулюються виконавчими пристроями (HVAC, зволоження/осушення, вентиляція, освітлення, зрошення).

У таблиці 1 наведено всі змінні, параметри, оператори та одиниці вимірювання, які використовуються далі в розділі.

CEUR фіксує тренд на повну телеметрію мікроклімату в теплицях (*T, RH,* CO_2, освітленість, тощо) з віддаленим моніторингом та керуванням: від

типових Wi-Fi/ESP8266 рішень до інтегрованих платформ (UI, мобільні застосунки) – це безпосередньо підсилює значуість компонентів «x_t, u_t, d_t» та конвеєр даних для ознак/прогнозу/рішень [1].

Датчики «на тросі»/cablebot збирають просторово-часові карти мікроклімату в теплиці (мікрозони, локальні стреси) – важливо для просторового розширення вектора стану x_t і коридорів $x_{\min}\ldots x_{\max}$ [2].

Метою керування є забезпечення комфорту/якості (для людини чи рослин), енергоефективності та стабільності процесів при дотриманні обмежень безпеки.

Таблиця 1

Позначення, змінні, параметри та одиниці вимірювання

Позначення	Опис	Одиниці
T	Температура повітря	°C
RH	Відносна вологість	%
CO_2	Концентрація вуглекислого газу	ppm
$CO_{2,out}$	Зовнішня (етелонна) концентрація CO_2	ppm
G_t	Інтенсивність утворення CO_2 (люди/процеси)	ppm·м^3/с
L	Освітленість/рівень світла	lx
V	Об'єм контрольованої зони	м^3
v	Швидкість руху повітря	м/с
x_t	Вектор стану системи у момент t	–
u_t	Вектор керувальних впливів у момент t	–
d_t	Зовнішні збурення	–
ε_t	Стохастичний шум/похибка моделі	–
x^*_t	Референтний (цільовий) вектор стану	–
x_{min}, x_{max}	Допустимі межі станів	–
u_{min}, u_{max}	Допустимі межі дій	–
Δu_{max}	Максимально допустима зміна дії за крок	–
$\|\cdot\|_2$, $\|\cdot\|_\infty$	Евклідова/максимальна норми	–
Q, R	Вагові матриці у критеріях якості	–
$\Phi(x,u)$	Штрафи/бар'єрні функції	–
H	Горизонт оптимізації/прогнозування	кроки
J	Функція якості/втрат	–
$e_s(T)$	Насичений парціальний тиск пари при T	кПа
VPD	Дефіцит тиску пари	кПа
PMV/PPD	Індекси теплового комфорту людини	– / %
THI	Індекс теплового навантаження (тварини)	–
s_t	Стан у MDP (RL)	–
a_t	Дія у MDP (RL)	–
$P(s'\|s,a)$	Ймовірність переходу у MDP	–
r_t	Миттєва винагорода у RL	–
γ	Коефіцієнт дисконтування у RL	–
λ, κ	Коефіцієнти ваг у винагороді/штрафі	–

E	Сумарне енергоспоживання/вартість	кВт·год / валюта
c^{el}_t, c^{th}_t	Тарифи електро-/теплоенергії	валюта/кВт·год
P^{fan}_t	Потужність вентиляторів	кВт
$Q^{heat/cool}_t$	Теплова/холодильна потужність	кВт
Q_t	Витрата припливного (змішуваного) повітря	м³/с
$\dot{m}$	Масова/об'ємна витрата повітря	кг/с або м³/с
ΔT	Різниця температур (внутрішня–зовнішня)	°C
C_{max}	Гранично допустимий CO_2	ppm
HMI	Інтерфейс «людина–машина» / Human–Machine Interface	–
MQTT	протокол обміну повідомленнями	–
LPWAN	технологія LPWAN/LoRaWAN (LNS – LoRaWAN Network Server	–
DCV	Demand-Controlled Ventilation	

Система описується вектором стану $x_t = [T_t, RH_t, CO_{2,t}, L_t, v_t, \ldots]^T$, керувальним вектором $u_t = [u_T, u_{RH}, u_{CO_2}, u_L, u_v, \ldots]^T$ (позиції заслінок, швидкості вентиляторів, подача тепла/холоду, інтенсивність освітлення, полив тощо) та збуреннями d_t (зовнішня погода, теплові надходження від людей/обладнання, добовий цикл). Динаміка має вигляд:

$$x_{t+1} = f(x_t, u_t, d_t) + \varepsilon_t, \quad (1)$$

де $f(\cdot)$ – (квазі)безперервна, невизначена та повільно змінна через сезон/культуру/навантаження.

Для рослин ключовим є дефіцит тиску пари (*VPD*), що інтегрує *T* та *RH*:

$$VPD = e_s(T) \cdot (1 - RH/100)\ [\text{kPa}],\ e_s(T) = 0.6108 \cdot \exp(17.27 \cdot T/(T+237.3))\ [\text{kPa}]. \quad (2)$$

Для людини використовують індекси комфорту PMV/PPD, для тварин – THI. Для кожного сценарію задаються референтні траєкторії x^*_t (годинні/добові профілі температури, вологості, CO_2, фотоперіод) та жорсткі обмеження на стани та дії:

$$x_{min} \le x\_t \le x_{max},\ u_{min} \le u\_t \le u_{max}. \quad (3)$$

Додатково обмежують швидкість зміни керувань, щоб уникати зносу обладнання:

$$||u_t - u_{t-1}||_\infty \le \Delta u_{max}. \quad (4)$$

Класичні підходи керуваня (гістерезисні правила, ПІД-регулятори) прості, але слабко враховують багатовимірні зв'язки (наприклад, компроміс CO_2↔вентиляція↔втрати тепла). Оптимізаційні підходи (MPC) використовують явну модель $f(\cdot)$ і мінімізують крос-цілі (помилка + енерговитрати). Дані-керовані підходи (ML/DL/RL) моделюють $f(\cdot)$ та/або безпосередньо політику керування з даних. Узагальнена функція якості (енергія/точність/штрафи за порушення) має тикий вигляд:

$$J = \sum_{t=0}^{H} \left(\|x_t - x_t^*\|_Q^2 + \|u_t\|_R^2 + \Phi(x_t, u_t) \right) \quad (5)$$

де Q,R≥0, Ф – бар'єрні/штрафні функції за вихід за межі, часті пуски, тощо; $\|x_t - x_t^*\|_Q^2$ – якість, урожай, комфорт; $\|u_t\|_R^2$ – енергія/ресурси; $\Phi(x_t,u_t)$ – штрафи за порушення/знос.

Інтелектуальне керування з навчанням з підкріпленням формулюється як Марковський процес прийняття рішень (MDP)

$$\mathcal{M} = \langle S, A, P, r, \gamma \rangle, \quad (6)$$

де стан $s_t=\psi$ (x_t, історія, зовнішні прогнози); дія $a_t \in A$ (дискретні режими/неперервні уставки); перехід $P(s_{t+1}|s_t,a_t)$ – невідомий, але оцінюваний з даних; винагорода $r_t = -(\|x_t - x_t^*\|_Q^2 + \lambda\|u_t\|_R^2) - 1_{\text{порушення}}\cdot\kappa$ та коефіцієнт дисконту γ.

Мета – максимізувати очікувану знижку сумарної винагороди:

$$\max_\pi \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right]. \quad (7)$$

Для Q-learning (дискретні дії) оцінюється $Q(s,a)$, для складніших систем – DDPG/SAC (неперервні дії), з обов'язковими обмеженнями безпеки (safe RL, штрафи/проєкції дій).

Типова багатошарова архітектура включає: пристрої/сенсори й приводи (DHT/SHT, MH-Z19, BH1750, PIR), приводи (вентилятори, заслінки, клапани); рівень Edge/Fog попередня обробка (фільтрація, калібрування, виявлення аномалій), локальні правила для швидких реакцій, кешування при втраті зв'язку; подієвий транспорт (MQTT для телеметрії, Kafka для потокової аналітики); хмарні/мікросервісні сервіси (сервіси збору й нормалізації, сховища (TSDB/PostgreSQL/кеш), аналітика (ML/RL), правила та оптимізація, API-шлюз, моніторинг/алертинг); HMI (веб-дашборди, мобільні застосунки, інтеграції (REST/WebSocket)).

Обробка даних здійснюється від фільтрації до злиття й ознак. Подібні інформаційні моделі системи (сенсори → обробка → керування вентиляцією/зрошенням/світлом) описані у [3]. Загальна схема конвеєру обробки показана на рисунку 1.

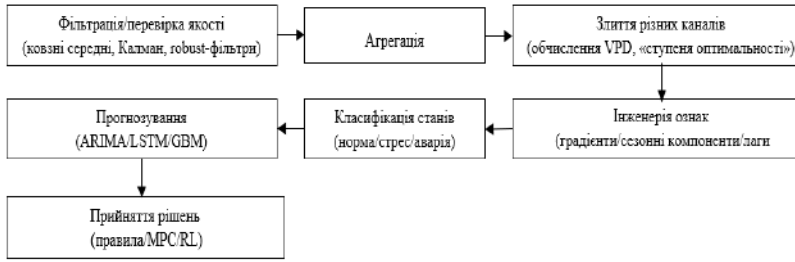


Рисунок 1. Узагальнена схема конвеєру обробки даних

Ключовими вимогами обробки даних є відсутність дрейфу калібрування, обробка пропусків та онлайнова переоцінка моделей.

Для багатомодульних об'єктів використовується мультиагентний підхід: агенти сенсингу, керування пристроями, безпеки, комфорту, медичного/агрономічного нагляду. Взаємодія через події та узгодження рішень (contract-net, чорні дошки, консенсус). Формальне описання поведінки можливе засобами темпоральної логіки (LTL) та моделей Крипке, що дозволяє верифікувати інваріанти безпеки (наприклад, «завжди, якщо $CO_2 > C_{max}$, тоді зрештою увімкнеться вентиляція»).

Для масштабування можна використовувати хмарні/мікросервісні ACS-архітектури з IoT-платформою (агенти, шини повідомлень) та сховищами часових рядів; це підсилює вашу вимогу до взаємодії HMI/аналітики/шлюзів [4].

Платформа IEEE IoT Journal для тепличного виробництва (хмара, REST/«Greenhouse Models as a Service») також приділяє увагу GMaaS/ML-сервісам [5, 6, 7].

Вентиляція балансує якість повітря та втрати енергії. "On-demand" стратегії (за CO_2, VOC, заповненістю) мінімізують надмірний повітрообмін. Вартість у функції якості J розширюється енергомоделлю:

$$E = \sum_t \left(c_t^{el} P_t^{fan} + c_t^{th} Q_t^{heat/cool} \right). \quad (8)$$

де $P_t^{fan} \propto m_t^3$ (кубічний закон), $Q_t^{heat/cool}$ залежить від від ΔT та інфільтрації. ML/RL-політики вчаться мінімізувати E при утриманні $CO_2 \leq C_{max}$ та обмежень шуму/комфорту.

ASHRAE 62.1 (оновлення 2022–2023) прямо визнає CO_2-керовану DCV (Demand-Controlled Ventilation) як інструмент енергоощадності при змінній заповненості – це формалізує поріг C_{max} та політику u_t для вентиляції [8].

Нові CO_2-орієнтовані стратегії DCV (пост-COVID) пов'язують вентиляцію з ризиком інфекцій і показують практичні схеми керування – корисні як додаткові Ф-штрафи/обмеження в J [9].

Сучасні дослідження демонструють застосування ANN/LSTM для короткострокового прогнозу параметрів теплиці (T, RH, CO_2, L). Це дозволяє перейти від реактивного до проактивного керування: прогнозований вектор стану x_{t+1} підставляється у ваш критерій якості J і зменшує член «похибка щодо профілю» $\|x_t - x_t^*\|_Q^2$ [10].

В роботі [11] показано LSTM-прогноз навантаження HVAC, який напряму з'єднує модуль прогнозу з економічним керуванням: розклад/сетпойнти обираються так, щоб мінімізувати енергетичну складову (8). На практиці це означає: заздалегідь «підхоплювати» або «приглушувати» установки у вигідні тарифні вікна; згладжувати пікові навантаження; утримувати клімат у «зелених коридорах» з меншими витратами. Рекомендація до архітектури: винести прогноз (15–60 хв горизонт) у окремий сервіс IoT–Edge–Cloud; використовувати MAE/MAPE/CRPS як метрики якості, а невизначеність прогнозу – як ваги/обмеження під час вибору дій.

Критеріми оцінювання ефективності у таких ситемах є:

- якість регулювання: MAE/RMSE/IAE по *T, RH,* CO_2, *L*; частка часу в «зелених» коридорах; динаміка VPD/PMV.
- енергія/ресурси: кВт·год, м³ води, CO_2-футпринт.
- надійність: відсоток часу без порушень, середній час відновлення, втрати пакетів.

ML/RL: точність/MAE прогнозів, стабільність політики, безпечність (кількість/тривалість порушень), узагальнюваність на нові сезони/культури/зони.

У роботі [12] описане керування теплицями та енергоменеджмент із використанням MPC: це безпосередньо відповідає формулюванню *J* з вагами *Q/R*, жорсткими обмеженнями (3) і обмеженням швидкості (4). MPC добре працює з прогнозами і дає керовані гарантії за обмежень.

У [13] активно розвивається безпечне Batch-RL: політики навчаються офлайн на історичних даних, що зменшує ризики «небезпечних» експериментів у реальному середовищі. Це узгоджується з штрафами Ф та вимогами fail-safe. Актуальні прийоми підвищення безпеки онлайн-RL: урахування модельної невизначеності (GP-моделі/квантільні оцінки) та консервативні алгоритми; «safety wrappers» – проєкція дій у безпечну множину перед застосуванням; human/grower-in-the-loop – експертні підказки, ручне схвалення або зміна ваг у винагороді $r_t = -\left(\|x_t - x_t^*\|_Q^2 + \lambda\|u_t\|_R^2\right) - k \cdot 1_{\text{порушення}}$.

Для теплиць показова гібридизація RL+MPC: RL генерує «цільові» сетпойнти/політики, а MPC гарантує дотримання обмежень і згладжує керування. Це поєднання зберігає адаптивність RL, але накладає формальні запобіжники [14].

Лінійка «прогноз (ANN/LSTM) → MPC/RL → економічна оптимізація» забезпечує логічний місток між теорією та практикою: прогноз зменшує похибки в *J*, MPC/RL – знаходить безпечні дії під обмеженнями та тарифами c_t^{el}, c_t^{th} , a human-in-the-loop/«сейфті-воркери» підтримують надійність у реальних умовах.

Праці, що наведені нижче, демонструють, як саме реалізуються елементи теорії – від сенсорики та просторового профілю мікроклімату x_t до даних, правил і хмарної оркестрації, та показують, як під'єднати прогностичні моделі до систем підтримки рішень і економічної оптимізації: Edge/IoT для мікроклімату [1] – завершена IoT-система з віддаленим керуванням, набором сенсорів і веб-інтерфейсами; слугує референсом для вашого конвеєра «сенсори → дані → рішення». «Кейбелбот» для мікроклімату/сканування простору [2] – дає просторову дискретизацію середовища, що напряму пов'язано з розширеним вектором стану x_t (мікрозони, градієнти). Інформаційна модель теплиці [3] – формалізує структуру даних і алгоритмів для автоуправління вентиляцією/зрошенням/світлом; відповідає вашим підрозділам про референтні профілі та обмеження. Хмарна ACS-архітектура

теплиці [4] – ілюструє IoT-платформу з агентами та зменшенням людського фактора; корелює з вашою моделлю мультиагентної координації. Сховище/DaW для динамічної теплиці [15] – підтримує етапи «фільтрація → агрегація → злиття ознак → прогноз», на які ви спираєтесь у розділі обробки даних. Decision Support Tool [16] – DSS для оптимального моніторингу й керування мікрокліматом у мережі теплиць (енергія/вода, прогнози/симуляції); практичне втілення вашого критерію якості J із багатьма складовими. LSTM-прогнозування для керування HVAC [11] по енергокритерію $E = \sum_t (c_t\ P_t\ )$ – прогноз навантаження підлаштовує графіки/сетпойнти з урахуванням тарифів c_t.

Отже, інтелектуальне керування мікрокліматом базується на поєднанні:

1. формалізованої динаміки об’єкта та багатокритеріальної оптимізації;
2. архітектури IoT–Edge/Fog–Cloud із подієвою взаємодією мікросервісів;
3. методів ML/RL для прогнозування та адаптивних політик;
4. мультиагентної координації та формальної верифікації інваріантів безпеки.

2 Агентно-орієнтовані системи управління комфортом і безпекою

Архітектура системи управління розумним будинком ґрунтується на принципах мультиагентних систем (MAS), що забезпечує узгоджену роботу різнорідних підсистем – клімат-контролю, безпеки, медичного моніторингу – з мінімальною участю користувача та формальними гарантіями. Дослідження [17–23] окреслюють архітектуру, ролі агентів та механізми їхньої взаємодії. У цьому розділі узагальнено архітектурні принципи мультиагентної системи (MAS), що забезпечує комфорт і безпеку мешканців [24]. Взаємодія між компонентами системи реалізована через подієву шину (Message Bus, MQTT) за моделлю «публікація/підписка» (Рис. 2), що забезпечує слабку зв'язність і масштабованість. Агенти, сенсори, виконавчі пристрої, користувач та хмарна аналітика об'єднані в єдиний простір подій.

Архітектурне середовище включає сенсори й виконавці, IoT-мережу, інтерфейси користувача, сервер чи хмару для обробки, підсистеми безпеки та медичну інтеграцію. Усі компоненти формують єдиний простір подій і команд, де агенти взаємодіють через шину, а хмарні моделі керують локальними діями.

Агенти мають спеціалізовані ролі: повідомлень (агрегація подій, сповіщення), керування пристроями (освітлення, клімат, прилади), камери й сигналізації (виявлення руху, тривоги), побутових сценаріїв і медичного догляду (розклади, фізіологічні параметри), а також контролю клімату (температура, вологість, освітленість з урахуванням енергоефективності). Це забезпечує масштабованість і просте додавання підсистем.

Комунікація відбувається через подієву шину та хмару, дії узгоджуються правилами й політиками безпеки. Телеметрія надходить до аналітики, назад поширюються політики та сетпойнти; користувач взаємодіє через HMI.

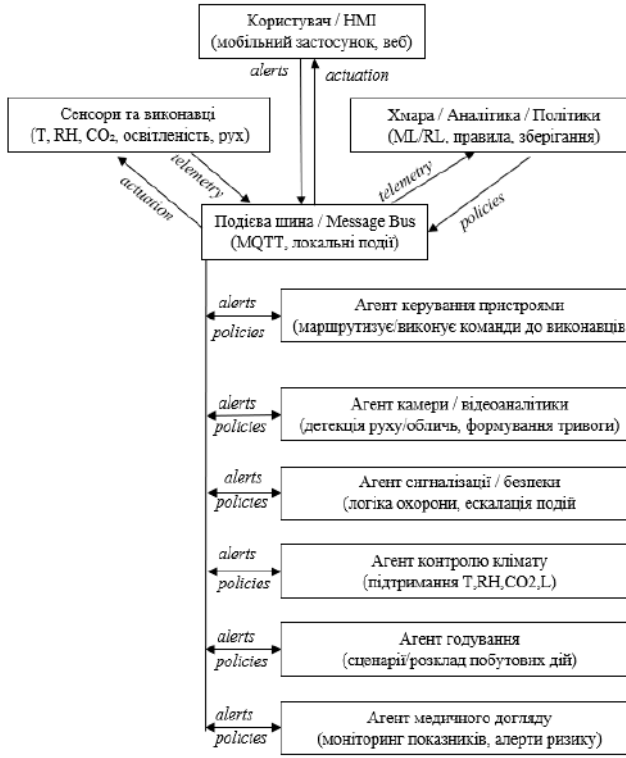


Рисунок 2. Взаємодія агентів через подієву шину (Message Bus)

Визначення агентної моделі MAS. Агентну систему подамо кортежем

$$\mathrm{M}_{MAS} = \langle A, S, R, Lab, Act, E, P, \Pi, \mathfrak{Z} \rangle, \quad (9)$$

де $A = \{a_i\}$ – множина агентів для кожного з яких задана множина дій Act, S – множина станів, яка включає стани агентів і ресурсів; відношення переходів $R \subseteq S \times S$ описує еволюцію у часі (або $P(s'|s,a)$ – стохастичний перехід для MDP/RL); $Lab: S \to 2^{AP}$ – маркування станів атомарними висловлюваннями AP; $Act = \prod_i Act_i$ – спільний простір дій;середовище подано як декартовий добуток $E = E_{phy} \times E_{net} \times E_{ui} \times E_{med} \times E_{sec}$, що охоплює фізичний простір, мережу/Інтернет речей, інтерфейси користувача, медичну та охоронну

інфраструктури; $\Pi = \{\pi_i\}$ – політики агентів (мапи зі стану/історії в дії); $\beth \subseteq \mathrm{LTL(AP)}$ – специфікації/інваріанти у LTL.

Властивості формулюються у темпоральній логіці LTL на моделі Крипке (S, R, Lab) з операторами G («завжди»), F («зрештою»), X («наступного кроку») та U («доти, доки»).

Для агента повідомлень істинною є формула $M \wedge G\, M$ – агент отримує й надалі розповсюджує повідомлення; для агента керування пристроями – $C\, F\, C$ – керування виконується й повторюється; для агента камери – $V\, X\, V$ – виявленню події слідує оновлення стану спостереження; для агента сигналізації – $A \wedge G\, A$ – охоронний режим активується і підтримується; для агента годування – $F\, X\, F$ – процес має крок-до-кроку оновлення; для агента медичного догляду – $H\, U\, H$ – моніторинг триває, доки стан не стабілізується; для властивостей середовища – $I \wedge G\, I$ – інтеграція з медичними пристроями підтримується постійно. Якщо у стані s_0 агент повідомлень отримує подію «недостатнє освітлення», а в s_1 пересилає її агенту керування пристроями або в хмарний сервіс, відбувається перехід $s_0 \rightarrow s_1$, при якому формули M та G M істинні відповідно в s_0 і на всіх наступних кроках.

У типових сценаріях агенти взаємодіють для реагування на рух, підтримки комфорту чи побутових процесів, координуючи сигнали, клімат і прилади. MAS забезпечує чіткий розподіл ролей, канали комунікації та інваріанти безпеки й комфорту, залишаючись масштабованою й придатною до верифікації. Інтеграція зі штучним інтелектом дозволяє прогнозувати, оптимізувати та адаптувати дії агентів, знижуючи енергоспоживання й підвищуючи надійність.

Для реальної експлуатації мультиагентної системи встановлюється «бюджет якості», що охоплює час відгуку, надійність, безпеку й масштабованість. Затримка «сенсор → виконавець» не повинна перевищувати 500 мс у звичайних та 100 – 200 мс у тривожних сценаріях; доступність сервісів – ≥ 99,9 %, MTTR ≤10 хв, реакція на тривогу ≤ 2 с. Система масштабується до ≥ 1000 пристроїв і ≥10 тис. повідомлень/с без деградації, підтримує rolling-update та інтероперабельність через MQTT і стандартизовані контракти даних. Безпека гарантується TLS 1.3, OAuth 2.1/OIDC, RBAC, сегментацією мережі й журналюванням; приватність – мінімізацією даних, обмеженням зберігання та шифруванням.

Ефективність MAS оцінюється за метриками клімату, комфорту/безпеки, енерговитрат і експлуатаційної якості. Для клімату: RMSE температури <0 ,5 °C, вологості < 3 %, CO_2 ≤ 1000 ppm (95-й перцентиль). Час у «зелених коридорах» має бути ≥ 95 %, при цьому HVAC споживає на 15–30 % менше енергії. Для безпеки: precision/recall ≥ 0,95, хибні тривоги < 2 %, відсутність конфліктів між агентами. Експлуатаційні SLI: затримка телеметрії в межах норм, втрата повідомлень <0,1 %, кіберінциденти дорівнюють 0, контроль

дрейфу моделей. Сукупно це підтверджує, що MAS забезпечує кращий баланс «комфорт – безпека – енерговартість», ніж традиційні системи.

3Архітектурні рішення систем забезпечення мікроклімату у приміщеннях

Мікроклімат суттєво впливає на врожайність культур, тому його контроль має враховувати взаємозалежність параметрів. У [25] запропоновано IoT-систему з трирівневою архітектурою (сенсорний, транспортний і прикладний рівні), де ключовими параметрами є температура (T), відносна вологість (RH) та дефіцит тиску пари (VPD). Дослідження [26] із застосуванням LoRaWAN підтвердило ефективність моніторингу *T*, *RH* і *VPD* у реальному часі для вирощування томатів. Концепція IoT ґрунтується на взаємодії об'єктів із унікальною адресацією [27], а забезпечення однорідності мікроклімату може підвищити врожайність і знизити енергоспоживання [28]. Окремо визначаються оптимальні параметри для тепличних томатів [29]. Запропонована архітектура орієнтована на контроль мікроклімату в закритих просторах із хмарною інтеграцією та мобільним доступом [30].

Архітектура ґрунтується на трирівневій моделі: сенсорний рівень (*T, RH,* CO_2, вологість субстрату, виконавці), транспортний (WSN/LPWAN, зокрема LoRaWAN, MQTT/HTTP) та прикладний/хмарний (БД часових рядів, розрахунок *VPD*, ML/RL-аналітика, HMI). Взаємодія організована через pub/sub із брокером повідомлень у хмарі чи на edge-шлюзі.

На рисунку 3 показано узагальнені потоки даних. Телеметрія піднімається від сенсорів через транспорт до прикладного рівня (*telemetry*). На основі аналізу формуються політики та сетпойнти (*policies*), які спускаються до транспорту і транслюються у конкретні керувальні дії для виконавців (*actuation*). Повернені події/підтвердження (*events/acks*) фіксують факт виконання або збої.

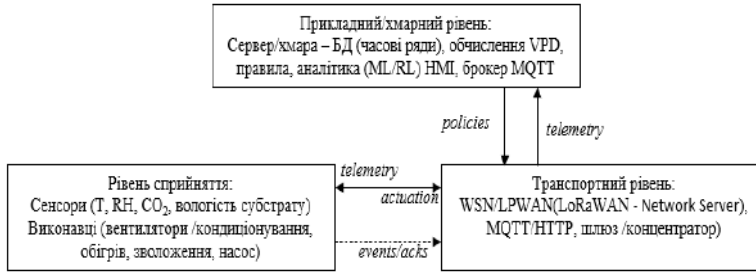


Рисунок 3. Узагальнена архітектура хмарної системи контролю мікроклімату

Так вибудовується замкнене коло «вимірювання → аналіз → політика → дія → підтвердження», яке узгоджується з агентною моделлю M_{MAS} (6): вектор стану $x_t = [T_t, RH_t, CO_2,t, L_t, v_t, \ldots]^T$ надходить як телеметрія,

прикладний рівень мінімізує похибку $\|x_t - x_t^*\|$ і обирає дії u_t, а обмеження на швидкість керувальних впливів дотримуються за умовою $\|u_t - u_{t-1}\|_\infty \le \Delta u_{max}$. У термінах MAS агент клімату генерує політики, агент керування пристроями інтерпретує їх у команди на транспорті, агент повідомлень забезпечує HMI-сповіщення, а агент безпеки блокує небезпечні дії згідно з LTL-інваріантами.

Policies визначають «що досягти» (T*, RH, VPD), а *actuation* – «як виконати» (реле, PWM, вентилятори). Розділення дає змогу працювати локально при втраті зв'язку, застосовувати safety wrappers і масштабувати систему без зміни пристроїв. Для критичних команд використовуються MQTT QoS 1/2, TLS 1.3, OAuth2/OIDC та RBAC, із затримками ≤100–200 мс (тривожні) та ≤500 мс (звичайні). Архітектура забезпечує збір параметрів, хмарний аналіз і вплив на обладнання, підтримуючи комфорт і енергоефективність.

4 Інтелектуальні підходи до керування мікрокліматом міських агросистем

Контрольоване середовище (Controlled Environment Agriculture, CEA) є основою міського продовольчого ланцюга. Для мікрозелені у замкнутих приміщеннях клімат, освітлення та зрошення підтримуються автоматизовано й економно з використанням даних та методів машинного навчання. Об'єктом керування є модульні стелажі, де вимірюють температуру повітря T, вологість RH, CO_2 та освітленість L (за потреби — вологість субстрату, швидкість повітря). Керувальні дії u_t охоплюють охолодження, обігрів, вентиляцію, зволоження й крапельне зрошення з подачею добрив. Мета — підтримання референтних профілів x_t^* для культур і фаз росту при мінімальних витратах ресурсів.

Кліматична установка включає стандартні вузли HVAC (радіатор, вентилятор, кондиціонер із випарником, компресором та конденсатором). Сенсорний контур містить датчики температури, вологості й тиску, а також положення заслінок; кількість точок вимірювання визначається конструкцією. Виконавчі пристрої — реле, приводи, вентилятори та клапани зрошення/добрив. Усі компоненти під'єднані до контролера, що збирає телеметрію й формує команди.

Логічна архітектура будується за моделлю Edge–Transport–Cloud. На Edge контролер виконує фільтрацію, швидкі правила та кешування. Транспорт реалізовано через MQTT/HTTP або LPWAN/LoRaWAN для доставки телеметрії (*telemetry*) й керувальних дій (*policies/actuation*). Хмара або локальний сервер зберігає часові ряди, обчислює показники (*VPD*), застосовує ML/RL та надає HMI.

Оптимальні режими визначають експериментально та статистично, уточнюють через ML (прогноз x_{t+1}, оцінка VPD, рекомендації) і RL під обмеженнями безпеки. Це підвищує врожайність і якість, знижуючи витрати ресурсів.

Експерименти проводили на модульній сіті-фермі з багаторівневими стелажами; обладнання включало вентилятори, обігрів, кондиціонер, заслінки, сенсори *T, RH,* CO_2, *L* та виконавчі механізми. Контролер на edge-рівні збирав телеметрію через MQTT/HTTP або LPWAN/LoRaWAN і виконував локальні правила швидкої реакції.

Дослід організовано за перехресним або рандомізованим блоковим дизайном по ярусах/зонах: кожна зона послідовно проходила всі режими керування у випадковому порядку. Передбачено щонайменше три цикли тривалістю повної фази вирощування (7–14 діб). Референтні траєкторії x_t^* (профілі дня/ночі для *T, RH*, фотоперіод, межі CO_2 і *VPD*) визначалися за агропаспортом культури; обмеження (3) і (4) задавали коридори станів та швидкість зміни дій. Порівнювали три стратегії: базову (гістерезис/ПІД), прогнозну (x_{t+1} + правила/MPC) та інтелектуальну (RL з «safety wrapper»).

Збір даних було уніфіковано: телеметрія фіксувалася з періодом 30–60 с, команди виконавцям – кожні 5 – 10 с для зменшення зносу. Перед кожною серією виконували калібрування *T/RH*, перевірку CO_2, тест зв'язку (втрата <0,1 %) і «сухий» прогін. Зрошення здійснювалося за комбінованим правилом «розклад + стан субстрату» з журналюванням усіх поливів.

Дані (*T, RH,* CO_2, *L*), стани виконавців та лічильники ресурсів зберігали у базі часових рядів. Попередня обробка охоплювала синхронізацію міток, фільтрацію викидів (правило 3σ), інтерполяцію коротких пропусків та, за потреби, згладжування для візуалізації. У прогнозних та інтелектуальних режимах додатково фіксували невизначеність прогнозів і версії політик.

Ефективність оцінювали за KPI: $RMSE_T$, $RMSE_{RH}$, частка часу в «зеленому коридорі» [x_{min}, x_{max}] для *T, RH, VPD*, 95-й перцентиль похибки; ресурси — інтегральна енерговитрата *E* (8), витрата води й добрив, частота вмикань; експлуатація — затримка «сенсор→дія», втрати повідомлень, MTTR, спрацювання «safety wrapper»; додатково – врожайність, рівномірність і відсоток браку. Статистику проводили через перевірку нормальності й дисперсій, далі ANOVA з повторними вимірюваннями або непараметричні аналоги, пост-хок з поправками, із 95% ДІ та ефектами (Cohen's d, η^2). Для енергопоказників виконували нормування або включали зовнішні умови як коварати.

Інтелектуальні політики працювали під «safety wrapper» із жорсткими обмеженнями (3)–(4), додатковими тригерами за CO_2 і температурою, таймаутами та антидребезгом. Для RL застосовували офлайн-навчання з історичних даних і поступове онлайн-оновлення з проєкцією небезпечних дій у безпечну множину. Конфігурацію системи фіксували повністю (версії моделей, топіки MQTT, KPI). Ключовим вузлом було автоматичне крапельне зрошення, інтегроване з сенсорами й контролером: подача води чи добрив виконувалася за розкладом і/або станом субстрату з журналом подій у HMI.

У агентній моделі клімат-агент формує *policies* для *T, RH* і *VPD*, агент керування пристроями реалізує їх як *actuation*, агент повідомлень відповідає за HMI, а агент безпеки застосовує «safety wrappers». Така взаємодія забезпечує

замкнений цикл «вимірювання → аналіз → дія → підтвердження». Запропонований підхід поєднує сенсорику, виконавчі механізми й аналітику, що дає відтворюваний мікроклімат та раціональне використання ресурсів у міських фермах.

5 Методи штучного інтелекту у керуванні кліматичними системами

Алгоритми контролю мікроклімату теплиць еволюціонували від класичних до методів штучного інтелекту, здатних навчатися на даних і приймати рішення без жорстких правил. Сучасні підходи включають байєсові мережі, SVM, нейронні мережі та генетичні алгоритми [31]. Нейромережі моделюють зв'язки між параметрами клімату й ростом рослин, а глибоке навчання аналізує часові ряди [32]. Для цього потрібна сенсорна інфраструктура (*T, RH,* CO_2, *L*) та виконавчі пристрої (вентиляція, зрошення, опалення), інтегровані через IoT для збору даних і регулювання в реальному часі [33]. Гібридні рішення поєднують прогноз і управління: DL-моделі прогнозують зміни, оптимізаційні алгоритми визначають дії [34], а surrogate models допомагають знаходити оптимальні параметри й підвищують ефективність [31]. Керування у закритому агровиробництві — багатокритеріальна задача з вимогами до якості, енерго- та водоспоживання й безпеки [35]. Традиційні ПІД-регулятори не враховують нелінійності й затримки, тоді як методи ШІ дають змогу перейти до прогнозно-оптимізаційної парадигми, формуючи дії ut з урахуванням обмежень (3) – (4) та енергетичного критерію (8).

У керуванні мікрокліматом застосовують три основні підходи: прогнозні моделі (ML/DL) для часових рядів *T, RH*, CO_2; оптимізацію й планування (евристики, генетичні алгоритми, MPC із сурогатами); та RL для формування політик. Додатково використовують байєсові мережі, SVM і комп'ютерний зір для аналізу стану рослин.

Дані сенсорів, події виконавців та зовнішні фактори (погода, графіки) формують конвеєр ознак, на виході якого LSTM/GRU/TCN або градієнтні ансамблі дають прогноз x_{t+1} та інтервали невизначеності. Для рослин критично оцінювати $VPD = e_s(T)\left(1 - \frac{RH}{100}\right)$, а для людей – PMV/PPD (як цільові або як штрафні складові). Прогноз $\hat{x}_{t+1}$ зменшує «помилковий» член у функції якості *J* та готує MPC/RL до проактивних дій.

MPC мінімізує J (5) з обмеженнями (3) – (4), використовуючи фізичну модель або surrogate model $f(\cdot)$. Сурогати (дерева рішень, NN) прискорюють оптимізацію на горизонті H, тоді як генетичні й роєві методи підходять для глобального пошуку сетпойнтів і графіків поливу.

RL формулює керування як $\mathcal{M}$ з виразу (6): стан $s_t=\psi(x_t,$історія,прогнози), дія a_t – уставки/режими, винагорода

$$r_t = -\left(\|x_t - x_t^*\|_Q^2 + \lambda\|u_t\|_R^2\right) - k \cdot 1_{\text{порушення}} - \eta \cdot \|u_t - u_{t-1}\|_\infty. \quad (10)$$

Для дискретних дій застосовують Q-learning/Double DQN, для неперервних – DDPG, TD3, SAC. Ризики зменшують batch/offline RL, safe-RL (проєкція дій у допустиму множину, штрафи, «safety wrapper»). Гібрид RL→MPC дозволяє RL генерувати сетпойнти, а MPC – гарантувати обмеження та згладжувати керування. Edge виконує локальні правила й фільтрацію, Transport забезпечує *telemetry/policies/actuation/events*, Cloud – зберігання даних, ML/RL-моделі та HMI. У MAS агент клімату формує policies, агент пристроїв реалізує actuation, агент безпеки застосовує LTL-інваріанти, агент повідомлень відповідає за HMI. Така структура спрощує інтеграцію й вивід рішень у контур. Ефективність оцінюють за $RMSE_T$, $RMSE_{RH}$, часом у «зелених коридорах», динамікою *VPD*, енерговитратами *E* (8), водою/добривами, а також затримками, втратою повідомлень і MTTR. Порівняння режимів (базовий, прогнозний, RL/MPC) виконують за ANOVA/Friedman із 95% ДІ.

Основні бар'єри – пояснюваність (XAI), стійкість до збоїв, узагальнюваність і стандартизація (KPI, API, безпека/приватність). Дорожня карта: конвеєр даних і профілі x_t, прогнозний модуль, MPC із сурогатом, пілотний safe-RL (off-policy), гібрид RL→MPC, XAI та регулярна перекалібровка.

6 Інтелектуальні методи керування вентиляційними процесами у приміщеннях

Сучасні будівлі потребують інтегрованих систем мікроклімату, орієнтованих на енергоефективність і комфорт [36]. Традиційні ручні чи фіксовані режими вентиляції часто спричиняють перевентиляцію або її нестачу, що підвищує витрати та погіршує якість повітря. Використання ШІ та IoT дозволяє автоматизувати контур, адаптувати його до умов експлуатації та зменшувати енергоспоживання [37]. Такі системи поєднують IoT-датчики (CO_2, *T*, *RH*, тверді частки, ЛОС), алгоритми ШІ для аналізу й прогнозування та контролери, які регулюють інтенсивність повітрообміну [38]. Робота будується як конвеєр: моніторинг → аналіз і прогноз → адаптивне керування; принцип "on-demand" задіює вентиляцію лише за потреби. Ефективність підтверджують приклади: RESPIRA® у метрополітенах та аеропортах знижує енерговитрати [39], а BrainBox AI у торговельних центрах скорочує їх на 21% [40].

Стан повітря описується вектором $x_t=[T_t, RH_t, CO_{2,t}, \ldots]^T$, а керуючі дії u_t охоплюють витрату припливного повітря/швидкість вентиляторів, положення заслінок, режими рекуперації/рециркуляції. Ціль – утримувати $CO_{2,t} \le C_{max}$ і параметри комфорту в "зелених коридорах" при мінімумі інтегральної вартості *E* ((3)–(4) та (8)). Спрощеною масообмінною моделлю для CO_2 слугуватиме дискретизована рівновага:

$$CO_{2,t+1} = CO_{2,t} + \left(\frac{G_t}{V} - \frac{Q_t}{V}\left(CO_{2,t} - CO_{2,out}\right)\right)\Delta t + \varepsilon_t. \quad (11)$$

де G_t – джерела CO_2 від людей/процесів, Q_t – витрата припливу, V – об'єм зони. Ця модель вбудовується у загальну динаміку $x_{t+1}=f(x_t,u_t,d_t)+\varepsilon_t$ з виразу (1).

Сенсори CO_2, *T, RH,* VOC та лічильники енергії формують telemetry, що на рівні Edge фільтрується й кешується при збоях. Транспортна шина (MQTT/HTTP, LoRaWAN) передає дані в аналітичний рівень, де ШІ генерує *policies* (сетпойнти, пороги), які транслюються в *actuation* (приводи, вентилятори, заслінки). Зворотні *events/acks* підтверджують виконання. У MAS агент клімату формує політики, агент пристроїв реалізує їх, агент безпеки застосовує LTL-обмеження, а агент повідомлень обслуговує HMI. Така структура підвищує надійність і масштабованість.

Базові схеми (гістерезис, ПІД) прості, але не враховують взаємодії «вентиляція–CO_2–втрати тепла». Прогнозні моделі (LSTM/GRU) дають оцінки $\widehat{CO}_{2,t+1}$, $\hat{T}_{t+1}$ з невизначеністю; MPC мінімізує критерій (5) з обмеженнями (3) – (4), використовуючи фізичну або ML-модель. RL (Q-learning, DDPG, SAC тощо) оптимізує політики "вентиляція-на-вимогу", працюючи під безпечними обмеженнями. Гібриди RL→MPC поєднують адаптивність RL із гарантіями MPC.

DCV змінює витрату Q_t за показами CO_2/заповненості, зводячи перевентиляцію до мінімуму. Енергетичну складову описує (8):

$$E = \sum_t \left(c_t^{el} P_t^{fan} + c_t^{th} Q_t^{heat/cool}\right), \quad P_t^{fan} \propto m_t^3,$$

де кубічний закон для вентиляторів дає значний виграш від частотного регулювання, а теплова складова залежить від ΔT та інфільтрацій. ШІ-модулі узгоджують Q_t із тарифним профілем c_t^{el}, c_t^{th} прогнозами навантаження і вимогами до CO_2/комфорту.

Політики працюють під «safety wrapper» із жорсткими межами станів і дій, таймаутами та fallback-режимами; канали шифруються (TLS), автентифікація здійснюється через RBAC, а сенсори регулярно калібруються. Якість визначається 95-м перцентилем $CO_2 \leq 1000$ ppm і ≥95% часу в допустимих коридорах *T, RH, VPD*; ефективність – зменшення E на 15–30% порівняно з базовим режимом. Порівняння режимів виконують за ANOVA/непараметричними тестами з 95% ДІ. Інтелектуальне керування вентиляцією поєднує сенсори, прогнозно-оптимізаційні моделі та подієву архітектуру, інтегруючись із MAS і трирівневою архітектурою для відтворюваності та масштабованості; гібрид MPC/RL додає адаптивність.

7 Застосування Q-learning у задачах оптимізації мікроклімату

Оптимізація вирощування мікрозелені здійснюється завдяки впровадженню інтелектуальних і розподілених систем контролю [41]. Використання IoT забезпечує моніторинг мікроклімату в реальному часі [42], а дані аналізуються хмарними платформами та алгоритмами ML [43–44]. Для підвищення

ресурсоощадності застосовують Q-learning [45], що у поєднанні з роботизованими системами [46], тепловими моделями та ПЗ для моделювання [47]. У такій парадигмі Q-learning автоматично формує політику керування, яка мінімізує витрати та підтримує параметри в допустимих коридорах відповідно до (3) – (4) і критерію (8) [48].

Контур керування подається як MDP (розділи 1–2): дискретизовані стани s (температура, вологість, освітленість тощо), скінченні дії a (режими вентиляції/обігріву/зволоження/світла), скалярна винагорода $r(s,a,s')$, дисконтування $\gamma \in (0,1)$. Класичне оновлення Q-функції використовує рівняння Беллмана:

$$Q_{t+1}(s,a) = (1-\alpha)Q_t(s,a) + \alpha \left[r_t + \gamma \max_{a'} Q_t(s',a') \right] \quad (12)$$

де α – крок навчання, s' – наступний стан, a' – можлива дія в s'. Алгоритм ітеративно поліпшує політику, балансуючи дослідження (ε-greedy) та експлуатацію.

Набір вимірювань охоплює T, *RH*, CO_2, *L*. Стан s формується дискретизацією цих параметрів, а дії *a* включають режими вентиляції, зволоження, обігріву, освітлення та поливу. Компактний простір дій прискорює збіжність і зменшує розмірність Q-таблиці. Винагорода визначається критерієм *J* (5): штраф за відхилення від x_t*, енергетичні витрати та «смикання» приводів, з додатковим контролем *VPD* (2) для рослин і PMV/PPD для людей. Безпека гарантується «safety-wrapper» із проєкцією дій у допустиму множину (3) – (4) та таймаутами на рівні Edge.

Реалізація базується на архітектурі Edge–Transport–Cloud: контролер обробляє телеметрію та виконує локальні правила, транспортна шина передає дані, а хмарний рівень веде базу часових рядів, обчислює *VPD*, оновлює Q-таблицю й надає HMI. У MAS (розд. 2) агент клімату формує політику Q-learning, агент керування пристроями реалізує її, агент безпеки застосовує інваріанти й «safety-wrapper», а агент повідомлень забезпечує взаємодію з користувачем. Експерименти підтвердили коректність передачі сигналів через Wi-Fi/ESP8266, стабільність Q-таблиці та зниження енергоспоживання при збереженні клімату в допустимих межах. Практичні параметри: 3–5 бінів на вимір, $\varepsilon \approx 0.2 \to 0.02$, $\alpha \approx 0.1–0.3$, $\gamma \approx 0.95–0.99$; ефективність оцінюється за $RMSE_T$, $RMSE_{RH}$, *VPD*/CO_2, питомою енергією та частотою перемикань. За потреби використовують безтабличні методи (DQN/TD3/SAC) або гібрид RL→MPC. У підсумку Q-learning забезпечує адаптивне та ресурсоощадне керування мікрокліматом сіті-ферми, інтегроване з IoT/хмарною інфраструктурою.

8 Практичні аспекти

Система реалізується на архітектурі Edge–Transport–Cloud: контролер виконує локальні правила, транспортна шина передає дані, а хмара обробляє телеметрію, веде базу часових рядів, обчислює *VPD* і оновлює Q-таблицю. У MAS агент клімату формує політику Q-learning, агент керування пристроями

реалізує її, агент безпеки застосовує інваріанти, а агент повідомлень забезпечує HMI. Експерименти підтвердили коректну роботу Wi-Fi/ESP8266, стабільність Q-таблиці та зниження енергоспоживання. Оптимальні параметри: 3–5 бінів, $\varepsilon \approx 0.2 \rightarrow 0.02$, $\alpha \approx 0.1–0.3$, $\gamma \approx 0.95–0.99$; ефективність оцінюють за $RMSE_T$, $RMSE_{RH}$, *VPD*/CO_2 та питомим споживанням. За потреби застосовуються безтабличні методи (DQN/TD3/SAC) чи гібриди RL→MPC.

Висновки

Запропоновано узгоджену рамку інтелектуального керування мікрокліматом: формальні моделі об'єкта, архітектура Edge–Transport–Cloud, мультиагентна взаємодія та алгоритми ШІ. Перехід від реактивних правил до зв'язки «прогноз (ANN/LSTM) → MPC/RL» стабілізує параметри в «зелених коридорах» і знижує витрати енергії; для вентиляції ефективними є DCV та гібриди MPC/RL під «safety wrapper».

Практичні правила — калібрування й контроль якості даних, розділення «policies ↔ actuation», прозорі KPI та коректні експериментальні дизайни - роблять результати відтворюваними та порівнюваними. Кейс Q-learning підтвердив життєздатність підходу й окреслив шлях масштабування до безтабличних методів та гібридів RL→MPC.

Обмеження пов'язані з дрейфом сенсорів, сезонністю даних і різнорідністю об'єктів; їх пом'якшують моніторинг, перекалібровка, адаптивні моделі та XAI. Подальші напрями — багатозонне керування, цифрові двійники, тарифно-обізнані політики та стандартизація API/KPI - для стійкого промислового розгортання й кращого балансу «комфорт - безпека - енерговартість».

Література

1. Prasol, N. S., Furikhata, D. V., Vakaliuk, T. A., & Regenel, T. Y. (2025, April). Integration of edge devices and IoT to create a climate monitoring system for plants. In *Proceedings of the 5th Edge Computing Workshop (DOORS 2025)* (Vol. 3943, pp. 4–19). Zhytomyr, Ukraine: CEUR-WS. https://ceur-ws.org/Vol-3943/paper01.pdf

2. Kavga, A., Bitas, D., Papastavros, K., Prapopoulos, M., & Kotsiris, G. (2020, September 24–27). Development of an integrated IoT-based greenhouse control cablebot system. In *Proceedings of the 9th International Conference on Information and Communication Technologies in Agriculture, Food and Environment (HAICTA 2020)* (pp. 518–525). Thessaloniki, Greece: CEUR-WS. https://ceur-ws.org/Vol-2761/HAICTA_2020_paper79.pdf

3. Kryvenchuk, Y., Doroschuk, R., Sundeha, R., & Shymanskyi, V. (2025, April 3–5). The system of automatic greenhouse care and its informational model. In *Proceedings of the 2nd International Conference on Smart Automation & Robotics for Future Industry (SMARTINDUSTRY 2025)* (pp. 67–76). Lviv, Ukraine: CEUR-WS. https://ceur-ws.org/Vol-3970/PAPER6.pdf

4. Khavalko, V., Baranovska, S., & Geliznyak, G. (2020). Investment feasibility of building the architecture of greenhouse automated control system based on the IoT and cloud technologies. In *Proceedings of the International Workshop on Conflict Management in Global Information Networks (CMiGIN 2019)* (pp. 311–323). CEUR-WS. https://ceur-ws.org/Vol-2588/paper26.pdf

5. Muñoz, M., Guzmán, J. L., Sánchez-Molina, J. A., Rodríguez, F., Torres, M., & Berenguel, M. (2020). A new IoT-based platform for greenhouse crop production. *IEEE Internet of Things Journal, 9*(9), 6325–6334. https://doi.org/10.1109/JIOT.2020.2996081

6. Muñoz, M., Guzmán, J. L., Sánchez, J. A., Rodríguez, F., & Torres, M. (2019). Greenhouse models as a service (GMaaS) for simulation and control. *IFAC-PapersOnLine, 52*(30), 190–195.

7. Jeon, K. E., Lin, T. N., She, J., Wong, S., Govindan, R., Al-Ansari, T., & Wang, B. (2023). LuXSensing beacon: Batteryless IoT sensor, design methodology, and field test for sustainable greenhouse monitoring. *IEEE Transactions on AgriFood Electronics, 1*(2), 86–98. https://doi.org/10.1109/TAFE.2023.3295383

8. ASHRAE. (2025, September 5). *Standards 62.1 & 62.2*. ASHRAE. https://www.ashrae.org/technical-resources/bookstore/standards-62-1-62-2?utm_source=chatgpt.com

9. Li, B., & Cai, W. (2022). A novel CO_2-based demand-controlled ventilation strategy to limit the spread of COVID-19 in the indoor environment. *Building and Environment, 219*, 109232. https://doi.org/10.1016/j.buildenv.2022.109232

10. Pandey, A., & Ramesh, V. (2024, April 18–20). Cloud cultivation: Optimizing agricultural automation practices through deep learning. In *Proceedings of the 1st International Conference on Smart Automation & Robotics for Future Industry (SMARTINDUSTRY 2024)* (pp. 209–217). Lviv, Ukraine: CEUR-WS. https://ceur-ws.org/Vol-3699/short2.pdf

11. Alden, R. E., Gong, H., Jones, E. S., Ababei, C., & Ionel, D. M. (2021). Artificial intelligence method for the forecast and separation of total and HVAC loads with application to energy management of smart and nZE homes. *IEEE Access, 9*, 160497–160509. https://doi.org/10.1109/ACCESS.2021.3129172

12. Nykyforova, L., Kiktev, N. A., Lendiel, T., Trokhaniak, V., & Ramsh, V. (2022, November 30 – December 2). Information and control system for controlling the irradiation process plants. In *Selected Papers of the IX International Scientific Conference "Information Technology and Implementation" (IT&I 2022)* (pp. 132–141). Kyiv, Ukraine. CEUR-WS. https://ceur-ws.org/Vol-3384/

13. Liu, H. Y., Balaji, B., Gao, S., Gupta, R., & Hong, D. (2022, May). Safe HVAC control via batch reinforcement learning. In *Proceedings of the 13th ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS 2022)* (pp. 181–192). https://doi.org/10.1109/ICCPS54341.2022.00023

14. Mallick, S., Airaldi, F., Dabiri, A., Sun, C., & De Schutter, B. (2025). Reinforcement learning-based model predictive control for greenhouse climate control. *Smart Agricultural Technology, 10*, 100751. https://doi.org/10.48550/arXiv.2409.12789

15. Kiktev, N. A., Lendiel, M., & Lendiel, T. (2023, December 20–21). Design of a data warehouse for a dynamic greenhouse control system. In *Selected Papers of the XX International Scientific Conference "Dynamical System Modeling and Stability Investigation" (DSMSI 2023)* (Vol. 1, pp. 69–78). Kyiv, Ukraine. CEUR-WS. https://ceur-ws.org/Vol-3687/

16. Ouammi, A., Choukai, O., Zejli, D., & Sayadi, S. (2020). A decision support tool for the optimal monitoring of the microclimate environments of connected smart greenhouses. *IEEE Access, 8,* 212094–212105. https://doi.org/10.1109/ACCESS.2020.3039889

17. Fonseca, T., Dias, T., Vitorino, J., Ferreira, L. L., & Praça, I. (2022). A low-cost multi-agent system for physical security in smart buildings. *arXiv preprint* arXiv:2209.00741. https://arxiv.org/abs/2209.00741

18. Hassan, W. H. (2019). Current research on Internet of Things (IoT) security: A survey. *Computer Networks, 148,* 283–294.

19. Berry, D., Usmani, A., Torero, J. L., Tate, A., McLaughlin, S., Potter, S., et al. (2005, September). FireGrid: Integrated emergency response and fire safety engineering for the future built environment. In *UK e-Science Programme All Hands Meeting.*

20. Çetin, A. E., Dimitropoulos, K., Gouverneur, B., Grammalidis, N., Günay, O., Habiboğlu, Y. H., et al. (2013). Video fire detection – review. *Digital Signal Processing, 23(6),* 1827–1843.

21. Harshika, G., Haani, U., Bhuvaneshwari, P., & Venkatesh, K. R. (2022). Smart Pet Insights System Based on IoT and ML. In H. S. Saini, R. K. Singh, G. K. Singh, P. K. Gupta, & K. P. Yadav (Eds.), *Innovative Data Communication Technologies and Application: Proceedings of ICIDCA 2021* (pp. 725–738). Singapore: Springer Nature Singapore.

22. Chen, Y., & Elshakankiri, M. (2020, April). Implementation of an IoT based pet care system. In *Proceedings of the 2020 Fifth International Conference on Fog and Mobile Edge Computing (FMEC)* (pp. 256–262). IEEE.

23. Qiao, B., Liu, K., & Guy, C. (2006, December). A multi-agent system for building control. In *Proceedings of the 2006 IEEE/WIC/ACM International Conference on Intelligent Agent Technology* (pp. 653–659). IEEE.

24. Аксак, Н. Г., Василевська, О. О., & Шеліхов, Ю. О. (2024, January 10–12). Архітектура мультиагентної системи для забезпечення комфорту та безпеки у розумних будинках. In *VI International Scientific and Practical Conference "The aspects of contemporary scientific research that encompass both theoretical and practical components"* (pp. 49–55). Venice, Italy: International Scientific Unity. https://isu-conference.com/wp-content/uploads/2024/01/The-aspects-of-contemporary-scientific-research-Jan-10-12-2024-Venice-Italy.pdf

25. Yan, M., Liu, P., Tong, C., Wang, X., Wen, F., Song, C., & O'Hare, G. M. (2020). A farmland microclimate monitoring system based on the Internet of Things. *International Journal of Embedded Systems, 12*(1), 81–92.

26. Rezvani, S. M. E., Abyaneh, H. Z., Shamshiri, R. R., Balasundram, S. K., Dworak, V., Goodarzi, M., … Mahns, B. (2020). IoT-based sensor data fusion for

determining optimality degrees of microclimate parameters in commercial greenhouse production of tomato. *Sensors*, *20*(22), 6474.

27. Boursianis, A. D., Papadopoulou, M. S., Diamantoulakis, P., Liopa-Tsakalidi, A., Barouchas, P., Salahas, G., Karagiannidis, G., Wan, S., & Goudos, S. K. (2020). Internet of Things (IoT) and Agricultural Unmanned Aerial Vehicles (UAVs) in smart farming: A comprehensive review. *Internet of Things*, 100187.

28. Lee, C., Chung, M., Shin, K.-Y., Im, Y.-H., & Yoon, S.-W. (2019). A study of the effects of enhanced uniformity control of greenhouse environment variables on crop growth. *Energies*, *12*, 1749.

29. Shamshiri, R. R., Bojic, I., van Henten, E., Balasundram, S. K., Dworak, V., Sultan, M., & Weltzien, C. (2020). Model-based evaluation of greenhouse microclimate using IoT sensor data fusion for energy efficient crop production. *Journal of Cleaner Production*, 121303.

30. Шеліхов, Ю. О., & Аксак, Н. (2023, February). Архітектура системи контролю мікроклімату у замкнутому приміщенні. *International scientific journal Grail of Science, 24,* 296–301. https://doi.org/10.36074/grail-of-science.17.02.2023.055

31. Hemming, S., de Zwart, F., Elings, A., Righini, I., & Petropoulou, A. (2019). Remote control of greenhouse vegetable production with artificial intelligence–Greenhouse climate, irrigation, and crop production. *Sensors, 19*(8), 1807.

32. Данильців, О. Б. Система штучного інтелекту на основі нейронних мереж для оцінювання стану рослин у “розумних” теплицях. *ELARTU – Інституційний репозитарій ТНТУ імені Івана Пулюя.* https://elartu.tntu.edu.ua/handle/lib/37915

33. Bhat, S. A., Huang, N.-F., Hussain, I., et al. (2021). On the classification of a greenhouse environment for a rose crop based on AI-based surrogate models. *Sustainability*, *13*(21), 12166. https://ideas.repec.org/a/gam/jsusta/v13y2021i21p12166-d671932.html

34. Lee, M.-H., Yao, M.-H., Kow, P.-Y., Kuo, B.-J., & Chang, F.-J. (2024). An artificial intelligence-powered environmental control system for resilient and efficient greenhouse farming. *Sustainability*, *16*(24), 10958. https://doi.org/10.3390/su162410958

35. Шеліхов, Ю. О., & Аксак, Н. Г. (2025). Використання методів штучного інтелекту для управління кліматичними системами в закритому агровиробництві. *29-й Міжнародний молодіжний форум «Радіоелектроніка та молодь у XXI столітті»* (т. 5, сс. 42–44). Харків: ХНУРЕ.

36. Шеліхов, Ю. О. (2025, 25–27 березня). Інтелектуальне керування вентиляцією для систем контролю мікроклімату в приміщенні. *Комп’ютерні інтелектуальні системи та мережі: матеріали XVIII Всеукраїнської науково-практичної WEB конференції аспірантів, студентів та молодих вчених* (сс. 350–353). Кривий Ріг: КНУ. https://sites.google.com/view/kicm/?pli=1

37. Altayeva, A. B., Omarov, B. S., & Cho, Y. I. (2016). Intelligent microclimate control system based on IoT. *International Journal of Fuzzy Logic & Intelligent Systems, 16*(4), 254–261.

38. Smart ventilation for better homes. (2024). *ebm-papst Magazine.*

39. Sener. (2021). RESPIRA®: Intelligent HVAC management system – Case study. *Sener Group.*

40. Ling, P. (2023). AI takes on growing role in HVAC system efficiencies. *Avnet Silica Tech Articles.*

41. Zhou, N. (2020). Intelligent control of agricultural irrigation based on reinforcement learning. *Journal of Physics: Conference Series, 1601*(5), 052031. https://doi.org/10.1088/1742-6596/1601/5/052031

42. Lodge, J. (2019). Controlled environment agriculture: Using intelligent systems on the next level. In *Agriculture and Environment Perspectives in Intelligent Systems* (pp. 1–33). IOS Press. https://doi.org/10.3233/AISE190002

43. Ashokkumar, K., Chowdary, D. D., & Sree, C. D. (2019, October). Data analysis and prediction on cloud computing for enhancing productivity in agriculture. *IOP Conference Series: Materials Science and Engineering, 590*(1), 012014. https://doi.org/10.1088/1757-899X/590/1/012014

44. Ashcraft, C., & Karra, K. (2021). Machine learning aided crop yield optimization. *arXiv preprint, arXiv:2111.00963.* https://doi.org/10.48550/arXiv.2111.00963

45. Ali, N., Wahid, A., Shaw, R., & Mason, K. (2024). A reinforcement learning approach to dairy farm battery management using Q learning. *Journal of Energy Storage, 93,* 112031. https://doi.org/10.48550/arXiv.2403.09499

46. Chougule, M. A., & Mashalkar, A. S. (2022). A comprehensive review of agriculture irrigation using artificial intelligence for crop production. In *Computational Intelligence in Manufacturing* (pp. 187–200). https://doi.org/10.1016/B978-0-323-91854-1.00002-9

47. Choab, N., Allouhi, A., El Maakoul, A., Kousksou, T., Saadeddine, S., & Jamil, A. (2019). Review on greenhouse microclimate and application: Design parameters, thermal modeling and simulation, climate controlling technologies. *Solar Energy, 191,* 109–137. https://doi.org/10.1016/j.solener.2019.08.042

48. Axak, N., Kushnaryov, M., & Shelikhov, Yu. (2024). The intelligent control of the city-farm microclimate based on the Q-learning algorithm. *Advanced Information Technology, 1*(3), 12–24. https://doi.org/10.17721/AIT.2024.1.02 https://ait.knu.ua/wp-content/uploads/2025/03/%D0%A1%D0%86%D0%A2-132024_28_02_25-13-22.pdf

INTELLIGENT MICROCLIMATE CONTROL SYSTEMS IN SMART ENVIRONMENTS

Dr.Sci. N. Aksak[1][0000-0001-8372-8432], **Ph.D. M. Kushnaryov**[2][0000-0002-3772-3195], **Yu. Shelikhov**[3][0009-0009-8970-6571]
Kharkiv National University of Radioelectronics, Ukraine
EMAIL: [1]nataliia.axak@nure.ua, [2]maksym.kushnarov@nure.ua, [3]yurii.shelikhov@nure.ua

Abstract. *This дослідження addresses the design and deployment of intelligent microclimate control systems across smart environments, including residential/commercial buildings, enclosed industrial spaces, and city-farms. We present a unified Edge–Transport–Cloud architecture with an event-driven data bus, time-series storage, and human–machine interfaces, alongside a multi-agent model that formalizes agent roles (climate, devices, safety, messaging) and safety invariants. The work surveys and integrates AI methods for forecasting and optimization: supervised ML/DL for short-term state prediction and comfort indices (e.g., VPD/PMV), Model Predictive Control (MPC), and Reinforcement Learning (RL), with a focus on Q-learning for policy synthesis under uncertainty. Special attention is given to intelligent ventilation (DCV), energy modeling, and evaluation criteria (tracking accuracy, time-in-band, energy use, reliability). Case studies span smart homes, closed spaces, and city-farms, illustrating how models are embedded into a microservice-based IoT platform and outlining practical deployment issues: sensor calibration, safety wrappers, secure channels (TLS), and human-in-the-loop operation. Overall, the work bridges theory and practice, demonstrating how combining IoT infrastructure with AI enables resilient comfort and resource efficiency.*

Keywords. *Intelligent systems; IoT; cloud computing; multi-agent systems; agents; machine learning, distributed computing; Q-learning; controlled environment; microclimate.*

UDC 681.5
DOI https://doi.org/10.36059/978-966-397-538-2-16

УДОСКОНАЛЕНА ДИНАМІЧНА ПРОГНОЗУЮЧА МОДЕЛЬ ПРОДУВКИ КИСНЕВОГО КОНВЕРТЕРА

Ph.D. Ю. Маріяш[1][0000-0002-0812-8960], **Ph.D. О. Степанець**[2][0000-0002-0812-8960]
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Україна
EMAIL: [1]y.mariiash@kpi.ua, [2]o.stepanets@kpi.ua

Анотація. *У цій роботі було розроблено динамічну модель режиму продувки в процесі кисневої печі, що є ключовим питанням для підвищення енерго- та ресурсо-ефективності сучасного виробництва сталі. Було розглянуто дуттєвий режим киснево-конвертерної плавки як технологічний об'єкт керування, виконано аналіз проблематики регулювання параметрів дуття в умовах нестаціонарності швидкості зневуглецювання металу. Під час продувки конвертерної ванни система керування вирішує завдання синхронізації процесів рафінування та нагріву металу при надійному дуттьовому режимі, а в кінці продувки – завдання визначення моменту її припинення. Встановлено, зміна ступеня окиснення вуглецю до CO_2 у порожнині конвертера залежить від зміни швидкості зневуглецювання яка, в свою чергу, залежать від відстані фурми до рівня спокійної ванни. Процес зміни швидкості зневуглецювання від зміни відстані фурми до рівня спокійної ванни є нестаціонарним, описується диференційним рівнянням першого порядку, стала часу якого залежить від періоду продувки; досліджено вплив інтенсивності подачі дуття на швидкість зневуглецювання металу. Перехідний процес зміни ступеня окиснення вуглецю до CO_2 від зміни положення пневмоклапану дуття кисню описується диференційним рівнянням третього порядку; отримана модель режиму продувки киснево-конвертерного процесу у вигляді керованої канонічної форми дискретної моделі в просторі станів в залежності від зміни відстані фурми до рівня спокійної ванни та інтенсивності дуття, яка використана в якості прогнозуючої моделі модельно-прогнозуючого регулятора.*

Ключові слова: *прогнозуюча модель, керування, модель в просторі станів, кисневий конвертер.*

1. Технологічні особливості керування режимом дуття киснево-конвертерного процесу

Киснево-конвертерна плавка відзначається складністю фізико-хімічних процесів, протікає з великою швидкістю і при високій температурі, характеризується багаторежимністю функціонування й великою розмірністю розв'язувальних задач. Якість одержуваної сталі визначається її складом і температурою. Конвертер можна розглядати як хімічний реактор, у якому відбуваються реакції окиснення різних елементів і процеси перерозподілу домішок і теплоти між металом та шлаком, що утворюється. Дослідження технологічних закономірностей проводили за даними конвертерного цеху ПАТ "Арселор-Міталл" Кривий Ріг (КМЗ) з конвертерами ємністю 160 тон. У конвертерах переплавляли переробний чавун з вмістом (%) силіцію 0,4…1,0; мангану 0,3…0,6; сірки 0,02…0,07; фосфору 0,02…0,15. Температура чавуну коливалась в діапазоні 1200…1400 ˚С [1]. В завалку завантажували металевий брухт у кількості 0…30 % від маси чавуну. Рідкий чавун із міксера подавали у 140 т ковшах. Інтенсивність подання кисню становила 2,5…3,0 м3/(т×хв). Сортамент марок сталі характеризувався вмістом вуглецю 0,09…0,40 % і температурою випуску 1580…1630 ˚С. Виплавка сталі є інтенсивним

процесом, тому оператор конвертера фізично не має можливості опрацювати великий об'єм інформації, вибрати найкращий режим та оперативно втрутитись у хід технологічно процесу плавки [2]. При ручному управлінні хід продувки часто відхиляється від оптимального, порушується процес шлакоутворення, у результаті чого шлак або звертається, або спінюється, що призводить до виносів та викидів. Тільки 45 – 50 % плавок, а іноді і менше, випускають при ручному управлінні з першої спроби. Одним з основних параметрів режиму дуття є інтенсивність продувки, від якої залежить хід процесів окиснення домішок і шлакоутворення [3]. Однак підвищення інтенсивності продувки призводить до зменшення окиснення заліза і переходу його в шлак (рис. 1), що пов'язано зі збільшенням механічного перемішування ванни та прискоренням розвитку вторинних реакцій окиснення домішок за рахунок оксидів заліза. Особливо це виявляється наприкінці продувки за низької швидкості зневуглецювання. Ступінь допалювання оксиду карбону (II) в порожнині конвертера (рис. 2) підвищується зі зростанням інтенсивності продування, що пов'язано з перерозподілом потоку кисню, який вивільняється від зменшення окиснення заліза [4].

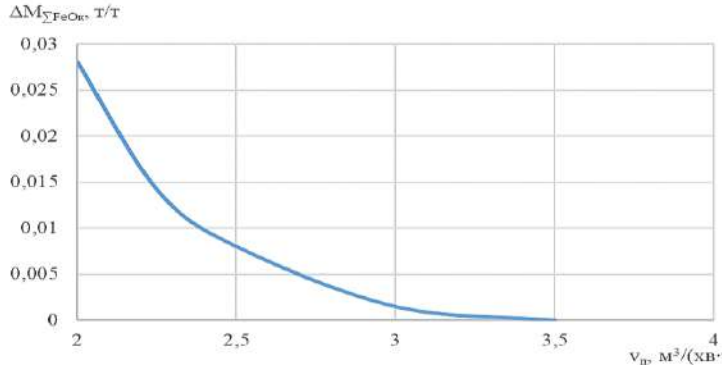


Рисунок 1. Вплив інтенсивності продування на питому масу оксидів заліза в шлаці

За експериментальними даними для 160-тонного кисневого конвертера складено залежність між інтенсивністю продувки і зміною основності кінцевого шлаку (рис. 3). Зі збільшенням інтенсивності продувки основність шлаку зменшується. З підвищенням інтенсивності продувки, знижується масова частка оксидів заліза в шлаку, отже, підвищується вміст мангану в металі.

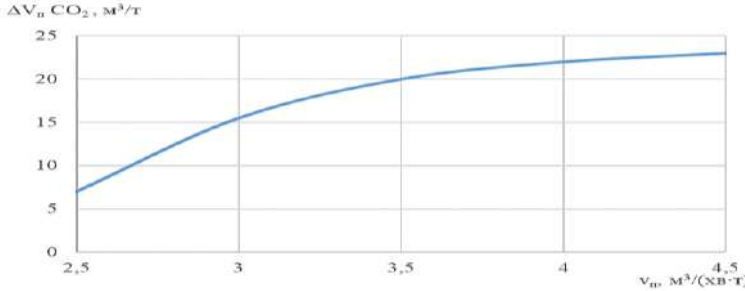


Рисунок 2. Вплив інтенсивності продування на підвищення питомого об'єму CO_2 у порожнині конвертера

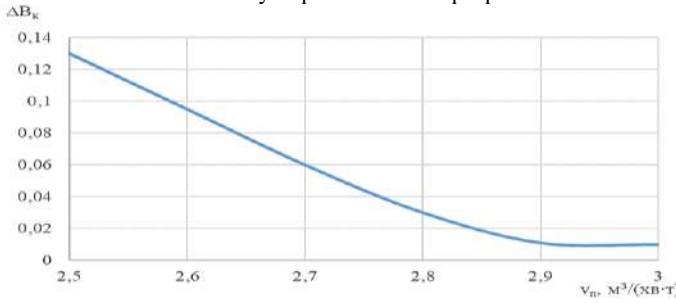


Рисунок 3. Вплив інтенсивності продування на основність кінцевого шлаку

Залежність між інтенсивністю продувки і зміною масової частки мангану в кінцевому металі зображено на рис. 4.

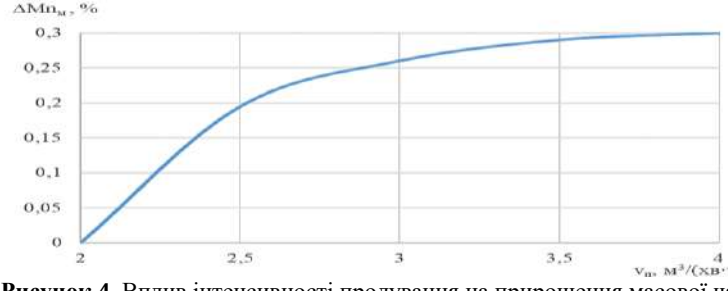


Рисунок 4. Вплив інтенсивності продування на приростання масової частки мангану в кінцевому металі

Залежність зношення футерівки, що визначається за складом оксиду магнію в кінцевому шлаку (останній переходить у шлак лише із футерівки), від інтенсивності продувки для 130-тонного конвертера зображено на рис. 5.

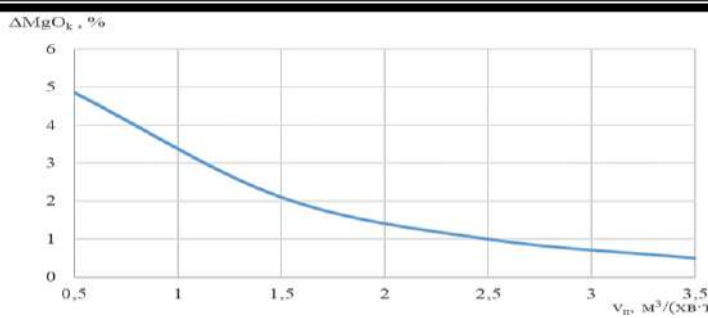


Рисунок 5. Вплив інтенсивності продування на масову частку оксиду магнію в кінцевому шлаці

З підвищенням інтенсивності продувки зношення футерівки зменшується, що пов'язано зі скороченням як тривалості продувки, так і контактування вогнетривів з агресивним шлаком і високотемпературним факелом. Характер залежності зміни питомої на 1т виплавленої сталі маси плавикового шпату від інтенсивності продувки (рис. 6) зумовлюється розріджувальним впливом на кінцевий шлак флюсу. Аналогічно впливає на кінцевий шлак підвищення в ньому вмісту оксидів заліза.

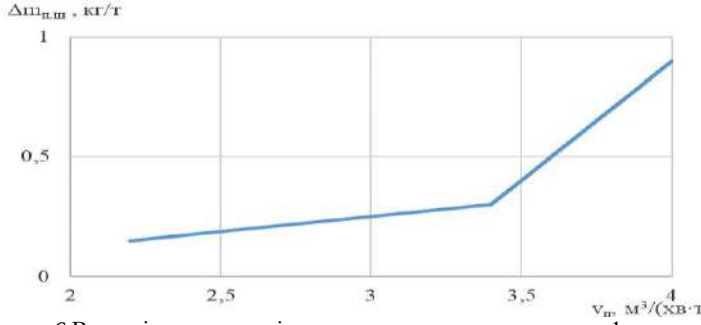


Рисунок.6 Вплив інтенсивності продування на питому масу на 1 т виплавленої сталі плавикового шпату

Однак значення останнього зменшується у разі збільшення інтенсивності продувки, що компенсується уведенням плавникового шпату. Тому підвищення інтенсивності продувки зумовлює збільшення ступеня допалювання CO в порожнині конвертера, масової частки мангану в металі наприкінці продувки і витрат плавикового шпату та зменшення оксидів заліза в кінцевому шлаку, його основності та зношення футерівки.

Висота розміщення фурми над рівнем спокійної ванни також впливає на хід процесів окиснення домішок і шлакоутворення. Збільшення масової частки оксидів заліза в кінцевому шлаку, що перераховані в еквівалентне значення

оксиду заліза (II), зі збільшенням висоти розміщення фурми над рівнем спокійної ванни (рис. 7) зумовлене зниженням швидкості масоперенесення при зменшені глибини реакційної зони.

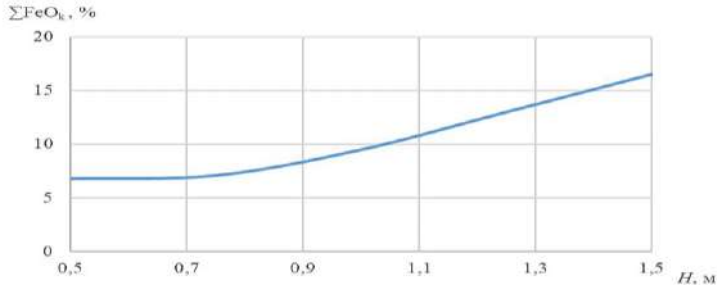


Рисунок 7. Вплив висоти розміщення фурми над рівнем спокійної ванни на окисненість кінцевого шлаку

Ступень допалювання CO до CO_2 в порожнині конвертера залежить від висоти розміщення фурми над рівнем спокійної ванни (рис. 8, де штрихова лінія – область нетехнологічної продувки). Ефективність допалювання оксиду карбону (II) в порожнині конвертера збільшується у разі підвищення висоти розміщення фурми над рівнем спокійної ванни, що пов'язано з великим потоком кисню у верхній зоні агрегату [5]. Залежність $\Delta B_к = f(H)$ (рис. 9) пояснюється збільшенням масової частки оксидів заліза у шлаку в разі зростання висоти розміщення фурми над рівнем спокійної ванни, що сприяє поліпшенню умов шлакоутворення. Наближення залежності до рівня насичення пов'язано з повним засвоєнням добавок вапна за підвищення значень висоти розміщення фурми.

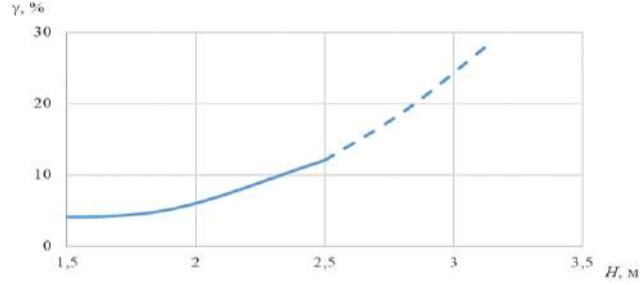


Рисунок 8. Вплив висоти розміщення фурми над рівнем спокійної ванни на ступінь допалювання CO до CO_2 у порожнині конвертера

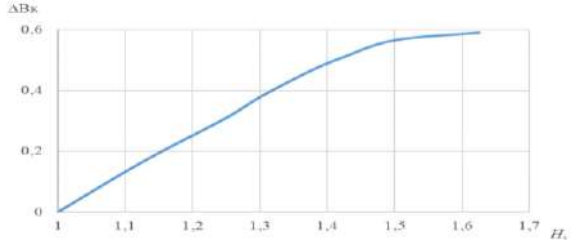


Рисунок 9. Вплив висоти розміщення фурми над рівнем спокійної ванни на зміну основності кінцевого шлаку

Фізико-хімічна природа залежності між зміною маси плавикового шпату на плавку і висотою розміщення фурми над рівнем спокійної ванни (рис. 10) пояснюється процесом формування рідкотекучості шлаку, яка забезпечується, крім добавок плавикового шпату, також зростанням масової частки оксидів заліза в шлаку. Що вища висота розміщення фурми, то більше міститься оксидів заліза в шлаку, тобто для підтримання такої самої рідкотекучості шлаку потрібна менша витрата плавикового шпату.

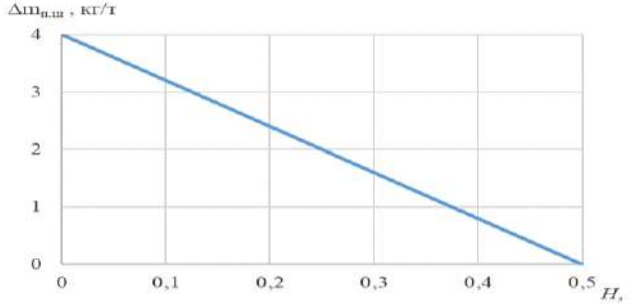


Рисунок 10. Вплив висоти розміщення фурми над рівнем спокійної ванни на зміну питомої маси плавикового шпату

Зростання масової частки оксидів заліза в шлаку з підвищенням висоти розміщення фурми можна пояснити залежність між цим параметром і зміною масової частки мангану в металі наприкінці продувки (рис. 11). Як уже зазначалось, стійкість футерівки можна оцінювати за масою оксиду магнію в шлаку. Залежність зміни масової частки оксиду магнію в кінцевому шлаку від висоти розміщення фурми над рівнем спокійної ванни зображено на рис. 12.

Що нижча висота розміщення фурми, то більше зношення футерівки конвертера, оскільки струмені дуття розмивають днище. Крім того, виникають технологічні труднощі наведення високоосновного шлаку. Зі збільшенням висоти розміщення фурми над рівнем спокійної ванни поліпшуються умови процесу шлакоутворення, підвищується основність шлаку і, як наслідок, знижується зношення футерівки.

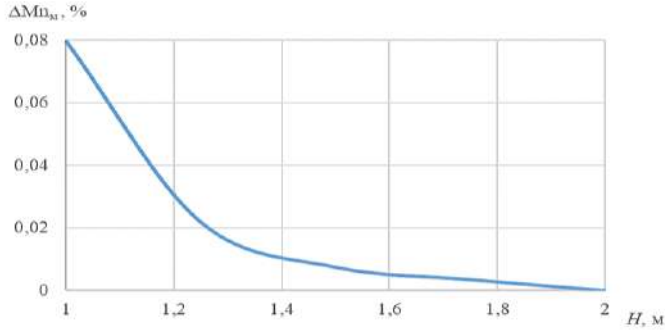


Рисунок 11. Вплив висоти розміщення фурми над рівнем спокійної ванни на зміну масової частки мангану наприкінці продувки

Отже, підвищення розміщення фурми над рівнем спокійної ванни призводить до збільшення основності та окиснення кінцевого шлаку, ступеня допалювання CO в порожнині конвертера, зменшення масової частки мангану в металі наприкінці продувки, витрат плавикового шпату й зношення футерівки.

Використання частини утвореного конвертерного газу в якості палива в порожнині конвертера для розплавлення металевого брухту дозволить збільшити частку брухту (рис. 13) у шихті, що в результаті призведе до зниження собівартості киснево-конвертерної сталі. Враховуючи, що гази, які відходять з конвертера, складаються приблизно з 90 % CO і 10 % CO_2, а питома теплота згоряння CO становить 12,7 МДж/м^3, великі резерви в збільшенні частки брухту криються в підвищенні ступеня згоряння CO в порожнині конвертера [6]. Реакція горіння CO в порожнині конвертера (спалахує при t>700 °C) екзотермічна $CO + \frac{1}{2}O_2 = CO_2 + Q$, теплова енергії якої дає можливість розплавити більше металобрухту та знизити частку чавуну.

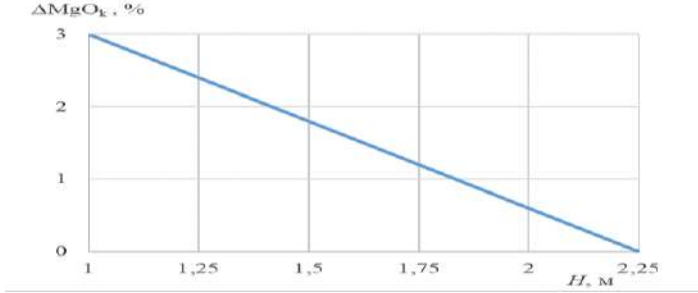


Рисунок 12. Вплив висоти розміщення фурми над рівнем спокійної ванни на зміну масової частки оксиду магнію в кінцевому шлаці

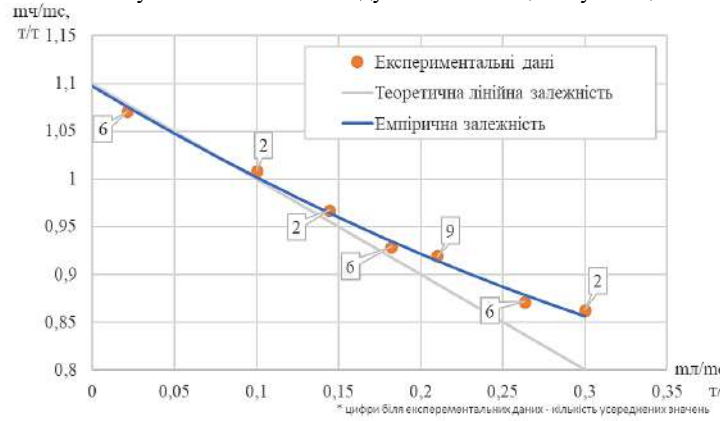


Рисунок 13. Залежність питомої (на тонну сталі) маси рідкого чавуну від питомої маси лому

Найбільш розповсюдженим способом збільшення ступеня допалювання CO у порожнині конвертера є регулювання відстані фурми над рівнем спокійної ванни. Ступінь згоряння CO в порожнині конвертера визначають параметри режиму дуття, зокрема відстань фурми від рівня спокійної ванни [7]. Регулюючи відстань, можна забезпечити оптимальну кількість тепла, що виділяється в конвертері від окиснення CO до CO_2. Дослідження показали, що при використанні системи керування для одноярусної фурми, яка спрямована на регулювання CO_2, його вміст в середньому можна підвищити до 12,7 %, що дає можливість збільшити частку брухту у шихті на 2,7 %. Найкращі ж результати можна отримати при застосуванні двоярусної фурми, в цьому випадку можна максимально підвищити кількість CO_2 до 25 %, та відповідно брухту у шихті на 5,3 %, не порушуючи умови шлакоутворення.

2 Розробка прогнозуючої моделі режиму продувки киснево-конвертерного процесу

У сучасних умовах розвитку металургійного виробництва актуальними являються задачі по розробці ресурсозберігаючих технологічних режимів виплавки сталі, теоретичних і практичних аспектів нових енергозберігаючих способів продувки сталеплавильної ванни технологічним газом та підвищення ефективності теплової роботи печей [8]. Одним із шляхів зниження витратних показників є утилізація фізичної та хімічної енергії газів, які відходять із конвертера, а саме допалювання CO до CO_2. Керування режимом продувки ККП потребує вимірювання масової частки CO_2 та інтенсивності продувки. В силу технологічних особливостей процесу вміст CO_2 розраховують по

балансовому рівнянню вмісту газів у дутті, повітрі та газоході. Кількість газів у газоході (1) визначається по балансовому рівнянню [9] вмісту аргону та азоту:

$$\begin{cases} Ar_г v_г = Ar_п v_п + Ar_д v_д \\ N_{2г} v_г = N_{2п} v_п + N_{2д} v_д, \end{cases} \quad (1)$$

де v – витрати газів, м^3/хв; “г”, “п”, “д” – індекси відповідно газоходу, повітря і дуття.

З (1) виразимо витрати газів у газоході (2) та повітря (3):

$$v_г = \frac{Ar_п N_{2д} - Ar_д N_{2п}}{Ar_п N_{2г} - Ar_г N_{2п}} v_д \quad (2)$$

$$v_п = \frac{Ar_г N_{2д} - Ar_д N_{2п}}{Ar_п N_{2г} - Ar_г N_{2п}} v_д \quad (3)$$

Гази, що відходять із конвертера, які складаються в основному із CO і CO_2, в газоході змішуються з повітрям. Реакція окиснення CO до CO_2 є екзотермічною реакцією $CO + \frac{1}{2}O_2 = CO_2 + Q$. При цьому частково витрачається кисень (4) повітря, підсмоктаний у газохід. Складемо баланс по кисню:

$$O_{2п} v_п = O_{2г} v_г + v_{O_2 р}, \quad (4)$$

де $v_{O_2 р}$ – витрати кисню на реакцію окиснення CO, м^3/хв.

Витрати кисню (5) на реакцію окиснення (6) при цьому складають:

$$v_{CO р} = 2v_{O_2 р} = 2(O_{2п} v_п - O_{2г} v_г). \quad (5)$$

$$\begin{cases} v_{CO_2} + v_{CO_р} = CO_{2г} v_г \\ v_{CO} - v_{CO р} = CO_г v_г. \end{cases} \quad (6)$$

Із системи (6) знаходимо (7):

$$\begin{cases} v_{CO_2} = CO_{2г} v_г - v_{CO р} \\ v_{CO} = CO_г v_г + v_{CO_р}. \end{cases} \quad (7)$$

Використовуючи (1) – (7), знаходимо коефіцієнт (8) допалювання CO в CO_2 у порожнині конвертера:

$$\gamma = \frac{v_{CO_2 к}}{v_{CO к} + v_{CO_2 к}} = \frac{CO_{2г} v_г - v_{CO р}}{(CO_г + CO_{2г}) \times v_г}. \quad (8)$$

Підставляючи (2) в (8) отримуємо (9):

$$\gamma = \frac{CO_{2г}\left(\frac{Ar_п N_{2д} - Ar_д N_{2п}}{Ar_п N_{2г} - Ar_г N_{2п}} v_д\right) - v_{CO р}}{(CO_г + CO_{2г}) \times \left(\frac{Ar_п N_{2д} - Ar_д N_{2п}}{Ar_п N_{2г} - Ar_г N_{2п}} v_д\right)} \quad (9)$$

Математична модель динамічного контролю продувки, заснована на врахуванні розподілу дуттьового кисню між металом, шлаком і конвертерним газом – представляє собою систему диференційних рівнянь, що характеризує матеріальний і тепловий баланс в конвертері і газоході охолоджувача

конвертерних газів. При створенні динамічної моделі процесу нехтуємо градієнтами керуючих параметрів, вважаючи, що просторова неоднорідність в ванній як по хімічному складі, так і по температурі внаслідок інтенсивного перемішування відсутня. Основний вклад в масообмін і енергетику процесу вносять термохімічні реакції окиснення вуглецю і заліза ванни. Вважаємо, що конвертерний газ як продукт зневуглецювання ванни складається з CO та CO_2. У робочому просторі конвертера оксид вуглецю частково спалюється в діоксид. Ця реакція, як і реакція горіння заліза, приводить до зменшення коефіцієнта засвоєння кисню вуглецем ванни і знижує його швидкість вигоряння. З врахуванням вище сказаного виразимо швидкість зневуглецювання ванни через об'ємну витрату кисню дуття (10):

$$\frac{dG_c}{d\tau} = 10^{-3}\frac{2{\cdot}12}{22{,}4}\cdot\left[v\gamma_1(1-\gamma_2) - 10^3\frac{22{,}4}{2{\cdot}12}(1-\gamma_{CO})\frac{dG_c}{d\tau} - 10^3\frac{22{,}4}{2{\cdot}56}\frac{dG_{Fe}}{d\tau}\right], \quad (10)$$

де $\frac{dG_c}{d\tau}$ – масова швидкість зневуглецювання ванни, т/хв; v – інтенсивність подачі дуття, м^3/хв; γ_1 – коефіцієнт, що характеризує чистоту дуття; γ_2 – коефіцієнт, що характеризує втрати дуття; γ_{CO} – масова доля карбону ванни, що окислюється до CO в порожнині конвертера за рахунок кисню дуття; $\frac{dG_{Fe}}{d\tau}$ – масова швидкість окиснення заліза ванни, т/хв. Виразимо швидкість зневуглецювання (11), враховуючи, що $\gamma_{CO_2} = 1 - \gamma_{CO}$ та $v_{O_2Fe} = 10^3\frac{22{,}4}{2{\cdot}56}\frac{dG_{Fe}}{d\tau}$:

$$\frac{dG_c}{d\tau} = 10^{-3}\frac{2{\cdot}12}{22{,}4}\frac{v\gamma_1(1-\gamma_2)-v_{O_2Fe}}{1+\gamma_{CO_2}}, \quad (11)$$

де v_{O_2Fe} – інтенсивність витрати кисню на окиснення заліза ванни, м^3/хв; γ_{CO_2} – масова доля карбону ванни, що окислюється до CO_2 в порожнині конвертера за рахунок кисню дуття. Інтенсивність витрати кисню на окиснення заліза ванни визначається залежністю (12):

$$v_{O_2Fe} = 10m_ч\gamma_{шл}\frac{16}{72}\frac{22{,}4}{32}\gamma_{FeO}\tau_n^{-1}, \quad (12)$$

де $m_ч$ – маса чавуну, т; $\gamma_{шл}$ – частка шлаку від маси металу; γ_{FeO} – вміст окису заліза в шлаку, %; τ_n – середня тривалість продувки, хв.

Величини γ_{FeO} (13) та γ_{CO_2} (14) є функціями від відстані фурми до рівня спокійної ванни:

$$\gamma_{FeO} = 16{,}34H - 5{,}63 \quad (13)$$

$$\gamma_{CO_2} = [10{,}2(H-1{,}5)^2 + 3{,}1]10^{-2}, \quad (14)$$

де H – положення фурми над рівнем спокійної ванни, м.

Для 160 тонного конвертера при частці шлаку 0.1 та середньою тривалістю продувки 20 хв $v_{O_2Fe} = 10 \cdot 160 \cdot 0{,}1 \cdot \frac{16}{72}\frac{22{,}4}{32}\gamma_{FeO}\frac{1}{20} = 1{,}244\gamma_{FeO}$. Підставимо (13) і (14) в (11). Для інтенсивності подачі дуття 400 м^3/хв, чистоті дуття кисню 0.99 та втратах 0.01 отримаємо залежність швидкості зневуглецювання (рис. 14) від положення фурми над рівнем спокійної ванни (15):

$$\frac{dG_c}{d\tau} = \frac{427{,}55 - 21{,}78H}{102(H-1{,}5)^2 + 1031}. \quad (15)$$

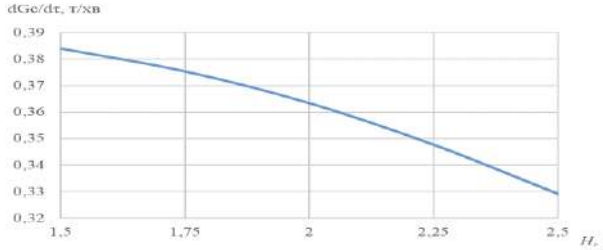


Рисунок 14. Вплив висоти розміщення фурми над рівнем спокійної ванни на зміну швидкості зневуглецювання

Перехідний процес зміни швидкості зневуглецювання $v_c = \frac{dG_c}{d\tau}$ від зміни відстані фурми до рівня спокійної ванни H описується диференційним рівнянням (16):

$$T_{v_c}^H \frac{dv_c(t)}{dt} + v_c(t) = k_{v_c}^H H(t), \quad (16)$$

де $k_{v_c}^H$ – коефіцієнт передачі по каналу відстань фурми до рівня спокійної ванни – швидкість зневуглецювання, $\frac{м}{хв \cdot м}$; $T_{v_c}^H$ – стала часу, с. Значення коефіцієнта передачі знайдемо зі залежності (2.15) $k_{v_c}^H = \frac{\Delta v_c}{\Delta H} = \frac{(0{,}329 - 0{,}383)\frac{м}{хв}}{(2{,}5-1{,}5)м} \approx -0{,}054\frac{м}{хв \cdot м}$. При знаходженні $T_{v_c}^H$ виникають труднощі, що пов'язані з перехідними процесами у вимірювачі швидкості зневуглецювання. Тому для визначення постійної часу використовували імпульсні характеристики та аналіз акустичних коливань через вимірювання тиску газів у перехідному газоході конвертера. Величина постійної часу нестаціонарна (рис. 15) та залежить від періоду плавки, залежність описана функцією Гауса третього порядку ($R^2 = 0.989$) (17):

$$T_{v_c}^H(\tau) = 7{,}05 \cdot e^{-\left(\frac{\tau - 8{,}47}{2{,}9}\right)^2} + 6{,}61 \cdot e^{-\left(\frac{\tau - 15{,}57}{2{,}6}\right)^2} + 11{,}48 \cdot e^{-\left(\frac{\tau - 9{,}78}{6{,}0}\right)^2}, \quad (17)$$

де τ – час від початку продувки, хв.

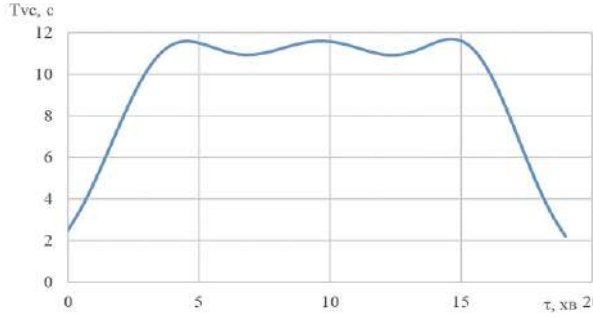


Рисунок 15. Залежність сталої часу $T_{v_c}^{H}$ від часу з початку продувки

Зміна швидкості зневуглецювання приводить до зміни ступеня окиснення вуглецю до CO_2 у порожнині конвертера. Цей процес також можна описати диференційним рівнянням першого (18) порядку виду:

$$T_{\gamma CO_2}^{v_c} \frac{d\gamma_{CO_2}(t)}{dt} + \gamma_{CO_2}(t) = k_{\gamma CO_2}^{v_c} v_c(t), \tag{18}$$

де $k_{\gamma CO_2}^{v_c}$ – коефіцієнт передачі по каналу швидкість зневуглецювання – ступінь окиснення вуглецю до CO_2, $\frac{хв}{т}$; $T_{\gamma CO_2}^{v_c}$ – стала часу, с.

Значення коефіцієнта передачі знайдемо зі залежності (11): $k_{\gamma CO_2}^{v_c} = \frac{\Delta\gamma_{CO_2}}{\Delta v_c} = \frac{0{,}133-0{,}0565}{(0{,}329-0{,}363)\frac{т}{хв}} \approx -2{,}25\frac{хв}{т}$, згідно результатів дослідження [10] отримано $T_{\gamma CO_2}^{v_c} \approx 2{,}15\ с.$

Перехідний процес зміни ступеня окиснення вуглецю до CO_2 від зміни відстані фурми до рівня спокійної ванни H утворений послідовним з'єднанням (16) та (18) та описується диференційним рівнянням (19):

$$T_{1\gamma CO_2}^{H}(\tau)\frac{d^2\gamma_{CO_2}(t)}{d^2t} + T_{2\gamma CO_2}^{H}(\tau)\frac{d\gamma_{CO_2}(t)}{dt} + \gamma_{CO_2}(t) = k_{\gamma CO_2}^{H} H(t),$$
$$k_{\gamma CO_2}^{H} = k_{v_c}^{H} \cdot k_{\gamma CO_2}^{v_c} = \left(-0{,}054\frac{т}{хв\cdot м}\right)\cdot\left(-2{,}25\frac{хв}{т}\right)\cdot 100\% = \tag{19}$$

де $12{,}15\frac{\%}{м}$;

$$T_{1\gamma CO_2}^{H}(\tau) = T_{v_c}^{H} T_{\gamma CO_2}^{v_c} = 15{,}16 \cdot e^{-\left(\frac{\tau-8{,}47}{2{,}9}\right)^2} + 14{,}21 \cdot e^{-\left(\frac{\tau-15{,}57}{2{,}6}\right)^2} + 24{,}68 \cdot e^{-\left(\frac{\tau-9{,}78}{6{,}0}\right)^2} [c];$$

$$T_{2\gamma CO_2}^{H}(\tau) = T_{v_c}^{H} + T_{\gamma CO_2}^{v_c} = 7{,}05 \cdot e^{-\left(\frac{\tau-8{,}47}{2{,}9}\right)^2} + 6{,}61 \cdot e^{-\left(\frac{\tau-15{,}57}{2{,}6}\right)^2} + 11{,}48 \cdot e^{-\left(\frac{\tau-9{,}78}{6{,}0}\right)^2} + 2{,}15\,[c].$$

Представимо процес (19) у вигляді керованої канонічної форми моделі в просторі станів (20):

$$\begin{cases} \begin{bmatrix} x_1'(t) \\ x_2'(t) \end{bmatrix} = \begin{bmatrix} 0, & 1 \\ -\frac{1}{T_{1\gamma CO_2}^{H}(\tau)}, & -\frac{T_{2\gamma CO_2}^{H}(\tau)}{T_{1\gamma CO_2}^{H}(\tau)} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{T_{1\gamma CO_2}^{H}(\tau)} \end{bmatrix} H(t), \\ \gamma_{CO_2}(t) = \begin{bmatrix} k_{\gamma CO_2}^{H} & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}. \end{cases} \quad (20)$$

Відомо, що швидкість зневуглецювання металу також визначається інтенсивністю подачі дуття [6]. На початку продувки при високій масовій долі вуглецю швидкість його окиснення змінюється від зміни інтенсивності подачі кисню в зону реакції, так як зневуглецювання протікає в основному в зоні контакту дуттьової струї з ванною. В кінці продувки, коли масова доля вуглецю у ванній досягає так званого «критичного» рівня, швидкість зневуглецювання знижується, оскільки лімітуючою ланкою процесу стає дифузія окиснюваного елемента в зоні реакції. У відповідності з уявленнями про два кінетичні періоди процесу окиснення вуглецю, перший описується рівнянням (21):

$$-\frac{dC}{d\tau} = \frac{K_1 \cdot \eta \cdot v}{G_M}, \quad (21)$$

де $\frac{dC}{d\tau}$ – швидкість зневуглецювання ванни, %/хв; K_1 – коефіцієнт, що характеризує перший кінетичний період, $\frac{m \cdot \%}{м^3}$; η – коефіцієнт, що залежить від об'ємної долі кисню в дутті і ступені його використання на зневуглецювання; v – об'ємна витрата кисневого дуття, м^3/хв; G_M – маса металевої ванни, т.

У другому кінетичному періоді, який наступає при рівномірності дифузійних потоків вуглецю і кисню, швидкість зневуглецювання описується рівнянням (22):

$$-\frac{dC}{d\tau} = \frac{\beta S C}{V_M}, \quad (22)$$

де β – коефіцієнт масопереносу вуглецю у ванній, м/хв; S – поверхня, на якій відбувається процес окиснення вуглецю, м^2; C – масова доля вуглецю в ванній, %; V_M – об'єм металевої ванни, м^3. Встановлена залежність ступеня засвоєння кисню ванни від інтенсивності дуття. Дослідження були проведені на 130-тоному конвертері при продувці через чотирьохсоплову фурму з кутом

нахилу осі сопла до вертикалі 13^0. Залежність середньої швидкості окиснення вуглецю від питомої витрати кисню ($\sigma = 0{,}012\frac{\%}{хв}$, R>0.95) зображена на рисунку 16.

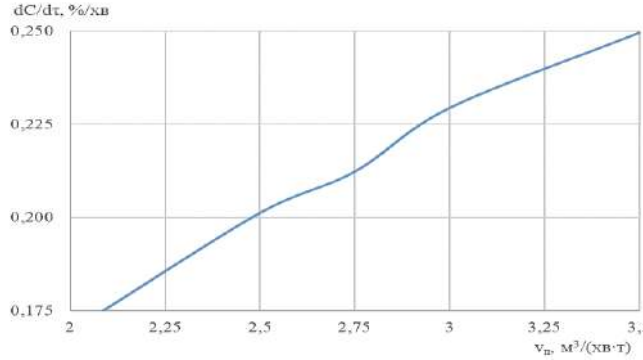


Рисунок 16. Вплив інтенсивності продування на швидкість зневуглецювання ванни

Перехідний процес зміни швидкості зневуглецювання v_c від інтенсивності дуття кисню можна описати диференційним рівнянням першого порядку виду (23):

$$T_{v_c}^v \frac{dv_c(t)}{dt} + v_c(t) = k_{v_c}^v v(t), \quad (23)$$

де $k_{v_c}^v$ – коефіцієнт передачі по каналу витрата кисню – швидкість зневуглецювання, $\frac{\frac{m}{м^3}}{хв}$; $T_{v_c}^v$ – стала часу, с. Значення динамічних властивостей знайдемо із досліджень описаних вище (рис. 14). Для 160т конвертера отримуємо $k_{v_c}^v = \frac{\Delta v_c}{\Delta v} = \frac{(0{,}367-0{,}322)\frac{m}{хв}}{(480-400)\frac{м^3}{хв}} \approx 0{,}56\frac{кг}{м^3}$; $T_{v_c}^v \approx 3{,}7\ с.$ Перехідний процес зміни ступеня окиснення вуглецю до CO_2 від зміни інтенсивності дуття кисню утворений послідовним з'єднанням (18) та (23) та описується диференційним рівнянням (24):

$$T_{v_c}^v T_{\gamma CO_2}^{v_c} \frac{d^2\gamma_{CO_2}(t)}{d^2t} + \left(T_{v_c}^v + T_{\gamma CO_2}^{v_c}\right)\frac{d\gamma_{CO_2}(t)}{dt} + \gamma_{CO_2}(t) = k_{\gamma CO_2}^v v(t), \quad (24)$$

$$k_{\gamma CO_2}^v = k_{v_c}^v \cdot k_{\gamma CO_2}^{v_c} = \left(0{,}56\frac{кг}{м^3}\right)\cdot\left(-2{,}25\cdot 10^{-3}\frac{хв}{кг}\right)\cdot 100\% = -0{,}126\frac{(\%\cdot хв)}{м^3}$$

де .

ТОУ інтенсивності дуття кисню кисневого конвертера утворений фізичним з'єднанням пневмоклапана та витратоміра, що представляє ємкість кисню, яка створює опір потоку речовини. Об'єкт, вхідною величиною якого

є положення пневмоклапану, а вихідною – витрата кисню, описується диференційним рівнянням першого порядку (25):

$$T_v^{u_{O_2}} \frac{dv(t)}{dt} + v(t) = k_v^{u_{O_2}} u_{vo_2}(t), \quad (25)$$

де u_{vo_2}– положення пневмоклапану, %; $k_v^{u_{O_2}}$ – коефіцієнт передачі по каналу положення пневмоклапану – інтенсивність дуття, $\frac{м^3}{(хв \cdot \%)}$; $T_v^{u_{O_2}}$ – стала ,с. Коефіцієнт передачі: $k_v = \frac{\Delta v}{\Delta u_{vo_2}} = \frac{600 \frac{м^3}{хв}}{100\%} = 6 \frac{м^3}{(хв \cdot \%)}$. З довідника [11] стала часу об'єкту $T_v^{u_{O_2}} = 1{,}2\ c$.

Представимо процес (25) у вигляді керованої канонічної форми моделі в просторі станів (26):

$$\begin{cases} [x_1'(t)] = \left[-\frac{1}{T_v^{u_{O_2}}}\right][x_1(t)] + \left[\frac{1}{T_v^{u_{O_2}}}\right] u_{vo_2}(t), \\ v(t) = [k_v^{u_{O_2}}][x_1(t)]. \end{cases} \quad (26)$$

Перехідний процес зміни ступеня окиснення вуглецю до CO_2 від зміни положення пневмоклапану дуття кисню утворений послідовним з'єднанням (24) та (25) та описується диференційним рівнянням (27):

$$T_{1\gamma CO_2}^{u_{O_2}} \frac{d^3 \gamma_{CO_2}(t)}{d^3 t} + T_{2\gamma CO_2}^{u_{O_2}} \frac{d^2 \gamma_{CO_2}(t)}{d^2 t} + T_{3\gamma CO_2}^{u_{O_2}} \frac{d\gamma_{CO_2}(t)}{dt} + \gamma_{CO_2}(t) = k_{\gamma CO_2}^{u_{O_2}} u_{vo_2}(t), \quad (27)$$

$$k_{\gamma CO_2}^{u_{O_2}} = k_{\gamma CO_2}^{v} \cdot k_v^{u_{O_2}} = \left(-0{,}126 \frac{(\%_{CO_2} \cdot хв)}{м^3}\right) \cdot \left(6 \frac{м^3}{(хв \cdot \%_{u_{O_2}})}\right) =$$

де $-0{,}756 \frac{\%_{CO_2}}{\%_{u_{O_2}}}$;

$$T_{1\gamma CO_2}^{u_{O_2}} = T_v^{u_{O_2}} T_{v_c}^{v} T_{\gamma CO_2}^{v_c} = 9{,}55[c];$$

$$T_{2\gamma CO_2}^{u_{O_2}} = T_{v_c}^{v} T_{\gamma CO_2}^{v_c} + T_{v_c}^{v} T_v^{u_{O_2}} + T_v^{u_{O_2}} T_{\gamma CO_2}^{v_c} = 14{,}98[c];$$

$$T_{3\gamma CO_2}^{u_{O_2}} = T_v^{u_{O_2}} + T_{v_c}^{v} + T_{\gamma CO_2}^{v_c} = 7{,}05[c].$$

Представимо процес (27) у вигляді керованої канонічної форми моделі в просторі станів (28):

$$\begin{cases} \begin{bmatrix} x_1'(t) \\ x_2'(t) \\ x_3'(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -\frac{1}{T_{1\gamma CO_2}^{u_{O_2}}} & -\frac{T_{3\gamma CO_2}^{u_{O_2}}}{T_{1\gamma CO_2}^{u_{O_2}}} & -\frac{T_{2\gamma CO_2}^{u_{O_2}}}{T_{1\gamma CO_2}^{u_{O_2}}} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{T_{1\gamma CO_2}^{u_{O_2}}} \end{bmatrix} u_{vo_2}(t), \\ \gamma_{CO_2}(t) = \begin{bmatrix} k_{\gamma CO_2}^{u_{O_2}} & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix}. \end{cases} \quad (28)$$

Виконаємо моделювання режиму продувки киснево-конвертерного процесу (рис. 17-20) в середовищі Matlab Simulink. Було обрано алгоритм

вирішення рівнянь ode23s (stiff/mod. Rosenbrock) зі зміною величиною кроку (variable-step). Абсолютна і відносна точність розрахунків – 0,00001. Механізм переміщення фурми опишемо інтегральною ланком першого порядку, швидкість переміщення фурми становить 0,2 м/с.

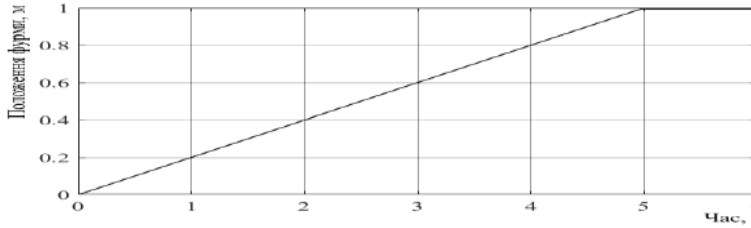


Рисунок 17. Перехідний процес моделі переміщення фурми

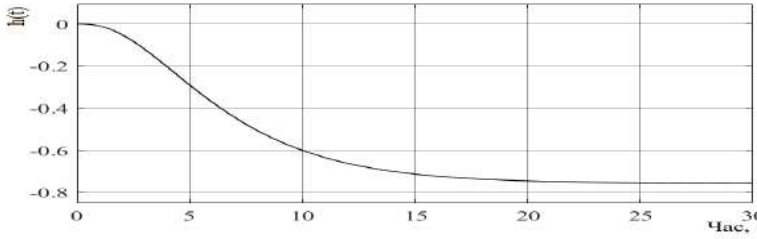


Рисунок 18. Перехідна характеристика системи по каналу зміна положення пневмоклапану кисню – вміст вуглекислого газу

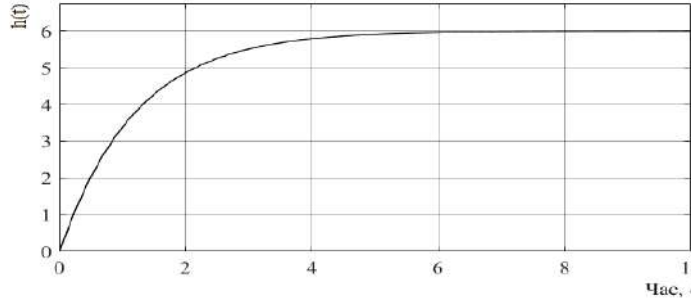


Рисунок 19. Перехідна характеристика системи по каналу зміна положення пневмоклапану кисню – витрата кисню

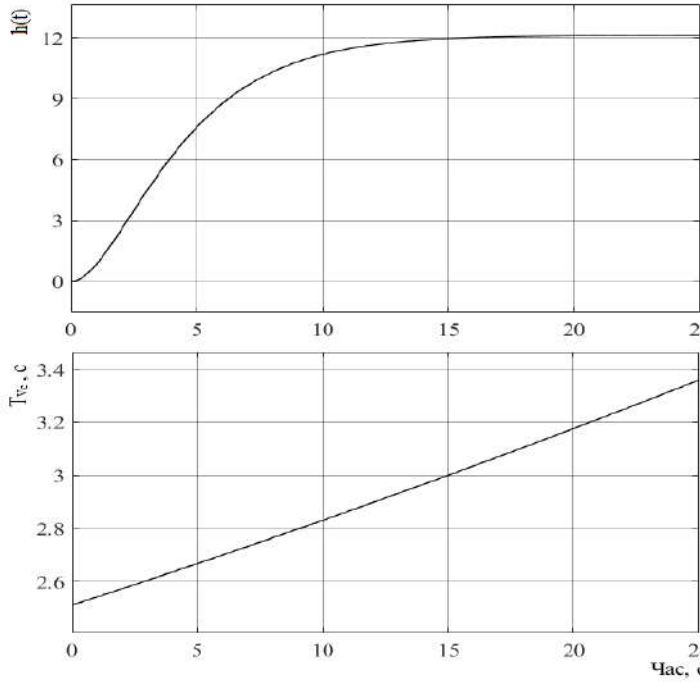


Рисунок 20. Перехідна характеристика системи по каналу зміна положення фурми – вміст вуглекислого газу

Було проаналізовано технологічні особливості керування параметрами режиму дуття ККП та розроблено модель в просторі станів даного процесу: встановлено, що одним з основних параметрів режиму дуття є інтенсивність продувки, від якої залежить хід процесів окиснення домішок і шлакоутворення. Однак підвищення інтенсивності продувки призводить до зменшення окиснення заліза і переходу його в шлак та зношення футерівки зменшується, що пов'язано зі скороченням як тривалості продувки, так і контактування вогнетривів з агресивним шлаком і високотемпературним факелом; проаналізовано вплив розміщення фурми над рівнем спокійної ванни, а саме підвищення розміщення фурми призводить до збільшення основності та окиснення кінцевого шлаку, ступеня допалювання CO в порожнині конвертера, зменшення масової частки мангану в металі наприкінці продувки, витрат плавикового шпату й зношення футерівки. Регулюючи відстань, можна забезпечити оптимальну кількість тепла, що виділяється в конвертері від окиснення CO до CO_2. Дослідження показали, що при використанні системи керування, яка спрямована на регулювання CO_2, його вміст в середньому можна підвищити до 12,7%, що дає можливість збільшити

частку брухту у шихті на 2,7 %; встановлено, зміна ступеня окиснення вуглецю до CO_2 у порожнині конвертера залежить від зміни швидкості зневуглецювання яка, в свою чергу, залежать від відстані фурми до рівня спокійної ванни. Процес зміни швидкості зневуглецювання від зміни відстані фурми до рівня спокійної ванни є нестаціонарним, описується диференційним рівнянням першого порядку, стала часу якого залежить від періоду продувки; досліджено вплив інтенсивності подачі дуття на швидкість зневуглецювання металу. Перехідний процес зміни ступеня окиснення вуглецю до CO_2 від зміни положення пневмоклапану дуття кисню описується диференційним рівнянням третього порядку; отримана модель режиму продувки киснево-конвертерного процесу у вигляді керованої канонічної форми дискретної моделі в просторі станів в залежності від зміни відстані фурми до рівня спокійної ванни та інтенсивності дуття, яка використана в якості прогнозуючої моделі модельно-прогнозуючого регулятора. Наведені чисельні значення динамічних властивостей отриманої моделі.

Література

1. Bogushevsky, V., & Skachok, A. (2016). The influence of uncontrolled disturbance actions on control of converter melting. *Metallurgical and Mining Industry*, (5), 128–133

2. Singha, P. (2023). Insights Thermodynamic in Basic Oxygen Steel Making Process. *ISIJ International*, *63*(8), 1343 – 1350. doi.org/10.2355/isijinternational.isijint-2023-087

3. Wang, R., Mohanty, I., Srivastava, A., Roy, T. K., Gupta, P., & Chattopadhyay, K. (2022). Hybrid Method for Endpoint Prediction in a Basic Oxygen Furnace. *Metals*, *12*(5), 801. https://doi.org/10.3390/met12050801

4. Xin, Z., Liu, Q., Zhang, J., & Lin, W. (2025). Analysis of Multi-Zone Reaction Mechanisms in BOF Steelmaking and Comprehensive Simulation. *Materials*, *18*(5), 1038. https://doi.org/10.3390/ma18051038

5. Kačur, J., Flegner, P., Durdán, M., & Laciak, M. (2022). Prediction of Temperature and Carbon Concentration in Oxygen Steelmaking by Machine Learning: A Comparative Study. *Applied Sciences*, *12*(15), 7757. https://doi.org/10.3390/app12157757i

6. Mariiash, Y. I., Stepanets, O. V., & Safonyk, A. P. (2024). Model predictive control for the blowing regime of the steelmaking process. *CEUR Workshop Proceedings*, 3790, 331–341.

7. Nenchev, B., Panwisawas, C., Yang, X., Fu, J., Dong, Z., Tao, Q., Gebelin, J.-C., Dunsmore, A., Dong, H., Li, M., Tao, B., Li, F., Ru, J., & Wang, F. (2022). Metallurgical Data Science for Steel Industry: A Case Study on Basic Oxygen Furnace. *steel research international*. https://doi.org/ 10.1002/srin.202100813

8. Niyayesh, M., & Uygun, Y. (2024). Predicting endpoint parameters of electric arc furnace–based steelmaking using artificial neural network. *The International Journal of Advanced Manufacturing Technology*. https://doi.org/10.1007/s00170-024-14502-x

9. Богушевський, В. С., Зубова, К. М., & Сухенко, В. Ю. (2012). Спосіб керування режиму дуття у кисневому конвертері (Патент України № 71152). НТУУ «КПІ»

10. Stepanets, O., & Mariiash, Y. (2020). MODEL PREDICTIVE CONTROL APPLICATION IN THE ENERGY SAVING TECHNOLOGY OF BASIC OXYGEN FURNACE. *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*, *10*(2), 70–74. https://doi.org/10.35784/iapgos.931

11. Кравченко, В. П., Койфман, О. О., & Сімкін, О. І. (2023). *Автоматизація технологічних процесів і виробництв у чорній металургії: навчальний посібник*. Одеса: Олді+

IMPROVED DYNAMIC PREDICTIVE MODEL FOR OXYGEN CONVERTER BLOWING

Ph.D. Y. Mariiash[1][0000-0002-0812-8960], **Ph.D. O. Stepanets**[2][0000-0002-0812-8960]
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Ukraine
EMAIL: [1]y.mariiash@kpi.ua, [2]o.stepanets@kpi.ua

Abstract. *This paper develops a dynamic model of the blowing mode in the oxygen furnace process, which is a key issue for improving the energy and resource efficiency of modern steel production. The blowing mode of oxygen converter smelting was considered as a technological control object, and an analysis of the problems of regulating blowing parameters under conditions of non-stationary metal decarburization rates was performed. During the blowing of the converter bath, the control system solves the problem of synchronizing the processes of refining and heating the metal with a reliable blowing mode, and at the end of the blowing, the problem of determining the moment of its termination. It has been established that the change in the degree of carbon oxidation to CO_2 in the converter cavity depends on the change in the decarburization rate, which, in turn, depends on the distance between the tuyere and the level of the calm bath. The process of changing the decarburization rate from the change in the distance between the tuyere and the level of the calm bath is non-stationary and is described by a first-order differential equation, the time constant of which depends on the blowing period; the influence of the intensity of the blast supply on the metal decarburization rate has been investigated. The transition process of changing the degree of carbon oxidation to CO_2 from changing the position of the oxygen blast pneumatic valve is described by a third-order differential equation; The resulting model of the oxygen converter process purging mode in the form of a controlled canonical form of a discrete model in state space depending on the change in the distance from the tuyere to the level of the calm bath and the intensity of blowing is used as a predictive model of a model-predictive controller.*

Keywords: *predictive model, control, state space model, oxygen converter.*

UDC 004.9
DOI https://doi.org/10.36059/978-966-397-538-2-17

NEURAL NETWORK MODEL FOR OPTIMIZED MANAGEMENT OF MEDICINES IN A UNIVERSAL FIRST AID KIT

I. Fedorchenko[1][0000-0003-1605-8066], **Dr.Sci. A. Oliinyk**[2][0000-0002-6740-6078],
K. Panychuk[3][0009-0009-4340-6688], **O. Korshun**[4][0009-0009-9178-793X],
Ph.D. A. Stepanenko[5][0000-0002-0267-7440]
[1,2,3,5]*National University "Zaporizhzhia Polytechnic", Ukraine,*
[4]*"OPTIME", Georgia*
EMAIL: [1]evg.fedorchenko@gmail.com, [2]olejnikaa@gmail.com,
[3]panichuck.kat@gmail.com, [4]oleh.korshun@gmail.com, [5]alex@zntu.edu.ua

Abstract. *A modified convolutional neural network model MediPackNet was developed with an accuracy of 92%, which correctly recognized all 5 test images of medicines. It showed good results at the level of 6 models based on already known ones, namely: InceptionV3, Xception, ResNet50V2, MobileNetV2, NASNetMobile and DenseNet169. In addition, AES, RSA data encryption methods and a combination of these algorithms were implemented. Based on the results of the analysis, it was concluded that hybrid encryption is the best for the developed software. A mobile application for the accounting of medicines for a universal first aid kit was created, with stable performance and the possibility for further development. The developed application has the potential to significantly facilitate the process of managing medicines stocks and contributes to more efficient use of medical resources and procurement optimization. In addition, the ability to track the course of treatment contributes to a more accurate following of medical recommendations and ensures more effective health monitoring. The use of the mobile application also has a significant impact on the environment, as it reduces the amount of hazardous waste associated with improper storage and disposal of expired medicines.*

Keywords: *accounting, medicines, first aid kit, packaging, recognition, mobile application, .NET MAUI, Python.*

Introduction

Medicines play an extremely important role in a person's life, helping to maintain health and improve quality of life. However, people often skip taking their medications or overpay for medications they already have at home, but don't remember about them. The World Health Organization has classified medication non-adherence as a major global problem [1]. It is estimated that 20% to 50% of patients do not take their medications properly [1]. The reasons for this are quite diverse, but the most common reasons are unintentional, such as confusion or simple forgetfulness. These problems can have serious health consequences and increase treatment costs [2]. In this regard, there is a need for a convenient and

efficient medication management tool. The universal first aid kit medicine accounting software is a relevant solution to solve these problems. It allows you to add, delete, and edit first aid kits' medicines, monitor their expiration dates, and also makes it possible to set up reminders to take medicines. This is especially useful for people who take a lot of medications or have chronic diseases [3].

Using a mobile application for accounting is convenient and affordable, as a mobile phone is usually always with the owner. It also has a significant impact on the environment, as it reduces the amount of hazardous waste associated with improper storage and disposal of expired medicines.

1 Analysis of known methods and available software tools

In order for a first aid kit to be effective and safe, it is necessary to keep proper records of the medicines it contains. This is extremely important, since human life and health depend on the correct use and proper condition of these medicines. The subject area of medicines accounting in the first aid kit is the pharmaceutical industry. To develop it, it is necessary to have knowledge of various medicines, their effects, classification, dosage forms, expiration dates, dosages, and routes of administration.

1.1 Review of existing methods

The task of accounting for medicines in a first aid kit is not new, as people have long been using first aid kits to store medicines. The technology development has made its implementation more convenient, efficient, and accessible to users.

There are several methods of performing the task of accounting for medicines in the first aid kit, namely: manual accounting, scanning barcodes, recognizing images of medicines packages. Manual accounting involves entering data about medicines manually. The user can enter the name, expiration date, quantity, and other information about each medicine as needed. This method is simple but can be time-consuming and labor-intensive.

Mobile applications can use a smartphone camera to scan barcodes located on drug packages [4]. After scanning, the application automatically finds the medicine in the database and adds it to the list of medicines. This method is convenient and fast, as it avoids manual data entry. However, it depends on the availability of barcodes on packages and the database with medicines and their code values.

The software can use an image recognition system to identify medicines [5]. The user can take a photo of the medicine's packaging, and the application will automatically add this medicine to the list of medicines. This method makes it easy to add new medicines to the list, but the accuracy of the recognition may depend on the quality of the image, the model, and the recognition database.

Manual accounting and package image recognition methods were chosen for further implementation because they make the accounting process more accessible and convenient for a wide range of users. A person will be able to choose a method of accounting that meets their needs, namely, entering data manually, recognizing the medicinal product by its packaging, or combining these methods.

Image recognition can be a more versatile method than barcode scanning, as it allows you to keep track of a variety of medicines, regardless of the type of barcode or its presence, packaging damage, differences in design or localization.

1.2 Review of image classification approaches and techniques

Traditional classification is a key data analysis approach that focuses on sorting data points into predefined classes or categories using specific rules and established features. Prior to the rise of deep learning, various conventional techniques such as Decision Trees, Support Vector Machines (SVM), Naive Bayes, and k-Nearest Neighbors (k-NN) were commonly applied for this task [6]. These methods were used and described in studies [6, 7, 8, 9, 10, 11, 12].

A Decision Tree (DT) is a hierarchical, rule-based method that utilizes a non-parametric approach [7]. It determines class membership by recursively dividing a dataset into homogeneous subsets. As a hierarchical classifier, it allows the acceptance or rejection of class labels at each intermediate step. The process consists of three main stages: partitioning the nodes, identifying the terminal nodes, and assigning class labels to these terminal nodes.

Kernel SVMs implicitly transform input feature vectors into a higher-dimensional space through the use of a kernel function, such as the Gaussian kernel [8]. In this transformed space, a maximal separating hyperplane is constructed, particularly for a two-class problem. Two parallel hyperplanes are then created symmetrically on either side of the separating hyperplane. The goal is to maximize the distance, known as the margin, between these two outer hyperplanes. It is believed that the larger the margin, the lower the generalization error of the classifier. SVMs are based on the principle of structural risk minimization, which aims to minimize an upper bound on the generalization error, unlike many classifiers that focus on minimizing empirical risk, or the error on the training set. The SVM algorithm works to find a decision function that minimizes a specific functional. Moreover, SVMs are capable of training nonlinear classifiers in high-dimensional spaces even with a small training set, thanks to the selection of a subset of vectors, known as support vectors, which define the optimal boundaries between the classes [8].

The Naive Bayes classifier operates on a probabilistic framework, assigning the class with the highest estimated posterior probability to the feature vector derived from the region of interest (ROI) [9]. This method is optimal when the attributes are independent (orthogonal), but it still performs well even without this assumption. Its simplicity enables strong performance with small training sets, and by constructing probabilistic models, it remains robust to outliers. Additionally, Naive Bayes creates soft decision boundaries, helping to prevent overfitting. However, the arbitrary choice of the distribution model for estimating probabilities P(x) and the limited flexibility of its decision boundaries can reduce its effectiveness in more complex multiclass problems [9].

The k-Nearest Neighbor classifier defines hyperspheres within the instance space by assigning the majority class of the k-nearest instances based on a specific

metric. It is asymptotically optimal and allows fast testing [10]. However, this method has several limitations. It is highly sensitive to the curse of dimensionality, as increasing the dimensionality tends to disperse the feature space, causing the local homogeneous regions representing the class prototypes to spread out [11]. The classification performance strongly depends on the chosen metric. Additionally, selecting a small value for k can lead to erratic decision boundaries, making the classifier more susceptible to outliers.

Although these methods are effective in many situations, they often require manual feature engineering, which can be labor-intensive and may not capture the complex patterns and relationships within sophisticated datasets. The chosen features are then used as inputs for the classification algorithms, which follow set criteria to assign data points to the appropriate classes [12].

With the rise of deep learning, Convolutional Neural Networks (CNNs) emerged as a powerful alternative, offering significant advancements in processing and analyzing data with complex structures, such as images and videos. Studies [6, 13, 14] highlight the effectiveness of CNNs in various applications, demonstrating their ability to achieve superior results.

Convolutional Neural Networks are a regularized form of multilayer perceptrons, which are typically fully connected networks where each neuron in one layer is linked to every neuron in the next [13]. CNNs differ by employing a mathematical operation called convolution instead of standard matrix multiplication in at least one of their layers. As a type of feedforward neural network, CNNs are particularly suited for handling data with grid-like structures. They learn features and patterns within the data using convolutional layers. The neurons in a CNN have learnable weights and biases, where each neuron processes inputs, performs a dot product, and may apply a non-linearity [14]. CNNs are inspired by biological processes in the visual cortex of the brain and have become a key solution for many computer vision tasks in artificial intelligence, such as image and video analysis.

Unlike traditional methods, CNNs leverage layered architectures to automatically learn and extract features from raw input data, significantly improving performance in tasks involving visual recognition and pattern detection.

1.3 Review of available software tools

First aid kits are an integral part of our lives. These are small containers that contain a variety of medical supplies and medications needed for minor medical interventions or first aid. They can be present at home, at work, in schools, cars, etc.

It is important to properly maintain the first aid kits and keep good records of the medicines they contain. This is necessary to ensure the safety and effective use of medicines in case of emergency. It is also important to have a clear list of medical supplies and update it in a timely manner to have complete information about available resources when needed.

Currently, there are already software tools [15, 16, 17, 18, 19, 20, 21, 22] that ensure the accounting of medicines. These programs allow you to accurately track the availability and quantity of medicines, control expiration dates, etc.

These software tools typically provide the ability to create lists of medicines, enter information about each product, and indicate the number of units available. Some of them even allow you to scan barcodes on drug packages to automatically fill in the data. In addition, the programs can display alerts about the approaching expiration date or shortage of certain medicines, which allows you to replenish stocks in a timely manner.

But in addition to the main advantages, some of them have disadvantages. For example, there is no tracking of the course of treatment and reminders to take medications if the user needs it, and no import of their own first aid kits.

The implemented mobile application has the following advantages over existing analogues: tracking the course of treatment, medication reminders, and recognition of certain medicines by their packaging.

2 Development of a modified convolutional neural network model MediPackNet

To implement the function of recognizing medicines from images of their packages in the software, a modified convolutional neural network model MediPackNet and 6 more models based on the already known ones were created, namely: InceptionV3, Xception, ResNet50V2, MobileNetV2, NASNetMobile, and DenseNet169 [23, 24, 25, 26, 27, 28, 29, 30]. The training results of these models were used to compare, analyze, and improve the models.

The following medicines were selected for recognition: Flucold-N, Gofen 200, No-Spa, Ortophen-Zdorovye Forte, and Phosphalugel.

The created model contains convolutional layers, maximum pooling layers, flatten layers, dropout layers, and fully connected layers. In addition, the images were normalized and randomly rotated, zoomed, and flipped horizontally to increase the diversity of the data (in all created models) [31, 32, 33]. ELU (Exponential Linear Unit) was chosen as the activation function because it showed the best results among the available functions. The output layer uses the Softmax activation function (5 classes). In the fully connected layers, L2-regularization was used (encourages smaller, more evenly distributed weights by adding a penalty), which improved the resulting model. The structure of this model is shown in Figure 1.

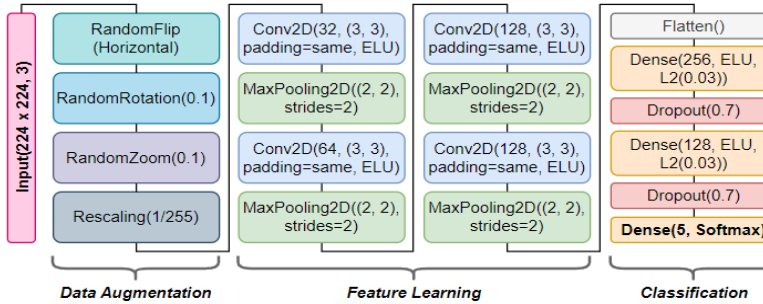


Figure 1. Structure of the MediPackNet model

The other 6 models include Inception V3, Xception, ResNet50V2, MobileNetV2, NASNetMobile, and DenseNet169, a global average pooling layer, and fully connected layers. The structure of these models using the example of the InceptionV3-based model is shown in Figure 2.

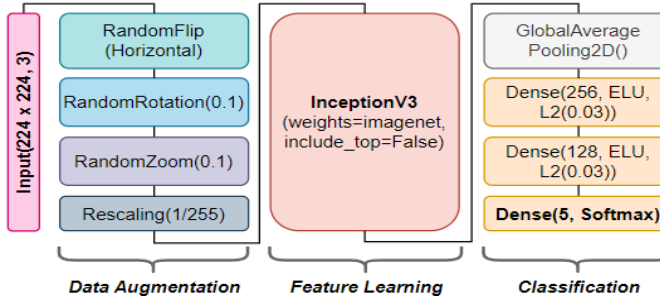


Figure 2. Structure of the model with Inception V3

3 Software implementation

3.1 Software structure

To design the architecture of the mobile application, the Model-View-ViewModel (MVVM) design pattern was used [34].

Software data is stored in a database that has been created and interacted with using SQLite.

The structure of a mobile application consists of 27 classes and 6 XAML (eXtensible Application Markup Language) files with 6 classes associated with them. The classes of view models and views have ViewModel and View in their names, respectively. All program files are structured into folders according to their purpose. The classes in the General folder were designed to support the interaction of views with view models.

The structure of the server part of the program consists of 5 files, 3 of which are responsible for creating a model for medicine recognition, and the rest for processing HTTP requests, encryption, decryption, and image recognition.

The architecture of a mobile application contains three functional parts: model, view, and view model. The diagram of classes representing the modules of the part containing the models is shown in Figure 3.

The diagram of classes representing the modules of the part containing the view models is shown in Figure 4.

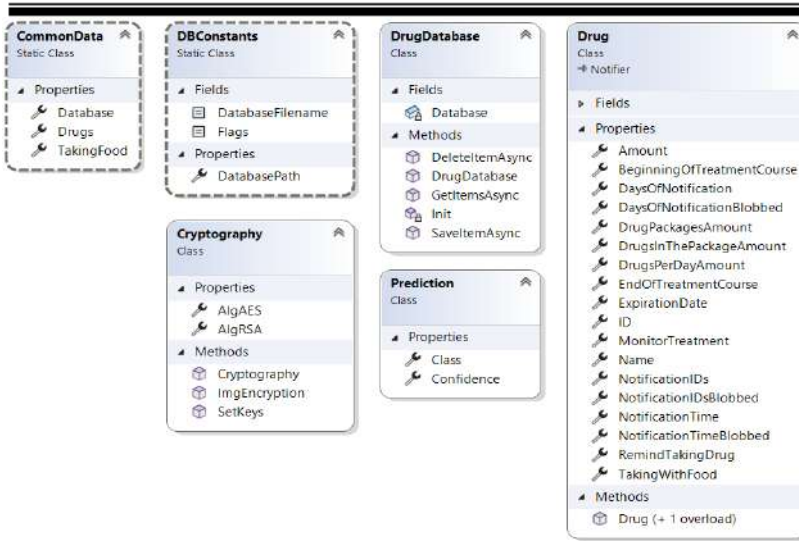


Figure 3. Diagram of classes that belong to models

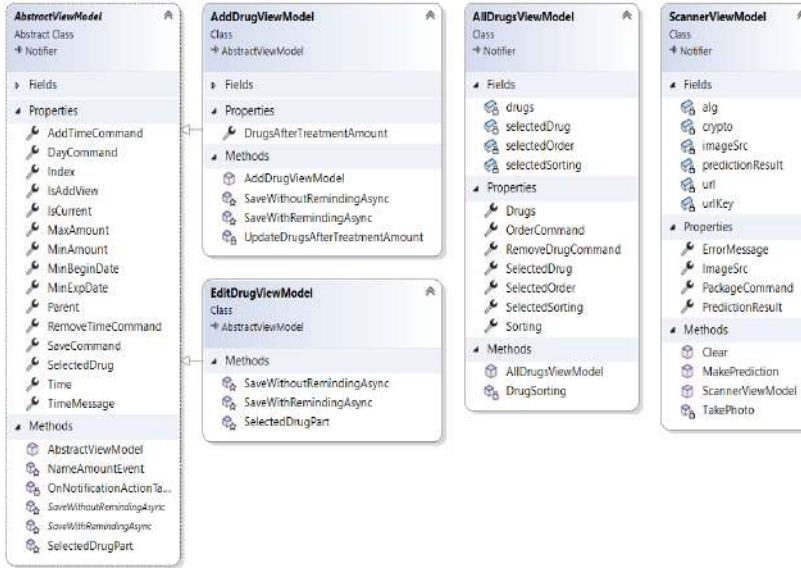


Figure 4. Diagram of classes that belong to view models

The diagram of classes representing the modules of the part containing the views is shown in Figure 5.

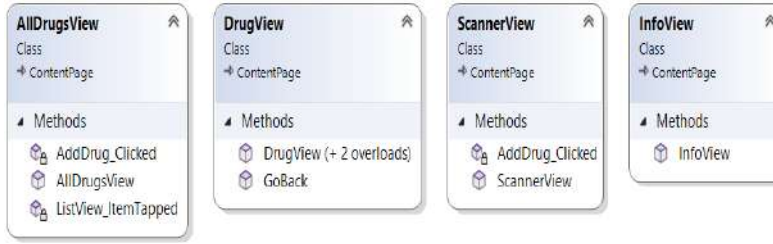


Figure 5. Diagram of classes that belong to views

The server part consists of three modules: cryptography, recognition, and creation of a model for recognition.

3.2 Image encryption

To implement the image encryption function in the software, modules were created that encrypt and decrypt data using the symmetric AES algorithm with CFB mode, the asymmetric RSA algorithm, and a combination of these algorithms [35, 36, 37, 38, 39]. The results of implementing these methods were used to compare, analyze, and select the best approach to protecting visual information [40, 41, 42].

In this application, encryption is essential because the images of medicine packages may include sensitive data such as personal annotations, prescription labels, QR codes, or barcodes that can be associated with specific users. If transmitted in plain form, such images could potentially be intercepted and analyzed, leading to privacy breaches, exposure of medical conditions, or manipulation of user medication data.

Using the System.Security.Cryptography library, the Cryptography class was created to encrypt images of medicine packages using AES and RSA algorithms and decrypt the shared key (encrypted with RSA in the hybrid approach). After encryption, the images are transmitted to the server.

On the server side, the Cryptography module (built with PyCryptodome) decrypts the received images and encrypts shared keys using RSA for secure communication. Keys are transmitted via HTTP requests, making asymmetric and hybrid encryption more suitable due to their resilience to insecure channels.

To assess the efficiency of the implemented encryption strategies, test sessions were conducted using typical medicine package images transmitted over a network. The hybrid approach (AES for image data + RSA for key exchange) showed a balanced trade-off between performance and security, with an execution time of 1.44 seconds, which is acceptable for real-time usage.

The characteristics of the encryption methods are summarized in Table 1.

Table 1

Comparison of implemented encryption methods

Comparison criteria	AES	RSA	AES and RSA
Keys	One shared key	Private and public keys	Key combination
Key size	256 bits	2048 bits	256 and 2048 bits
Time of execution	1.09 s	51.47 s	1.44 s
Complexity of key management	High	Low	Low
Transfer of keys	Difficult	Easy	Easy
Complexity	Low	High	Medium

The table shows that each method has its advantages and disadvantages. AES is known for its speed and efficiency in encrypting large amounts of data, while RSA provides secure key exchange and a high level of security, but may be less efficient for large amounts of data. The hybrid method leverages the strengths of both: using AES for data encryption and RSA for key exchange, ensuring security without compromising performance. Therefore, hybrid encryption is best suited for protecting visual medical data in a client-server application.

3.3 Recognition models training

In addition to ensuring data security, an important component of the project is to recognize medicines by their packaging. For this purpose, the created recognition models were trained and the results are shown in the form of graphs in Figures 6-12.

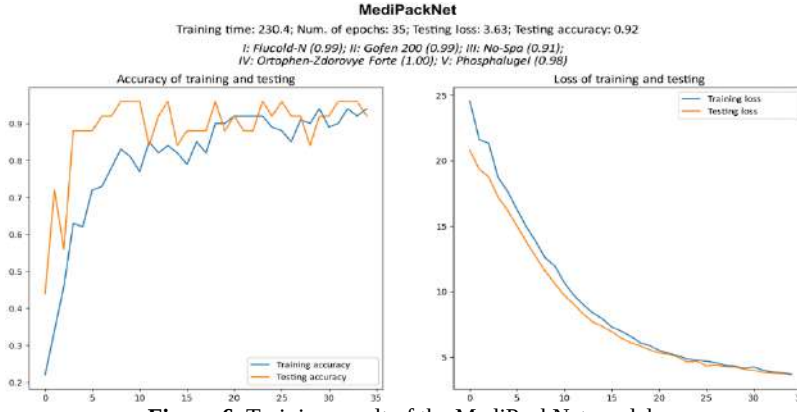


Figure 6. Training result of the MediPackNet model

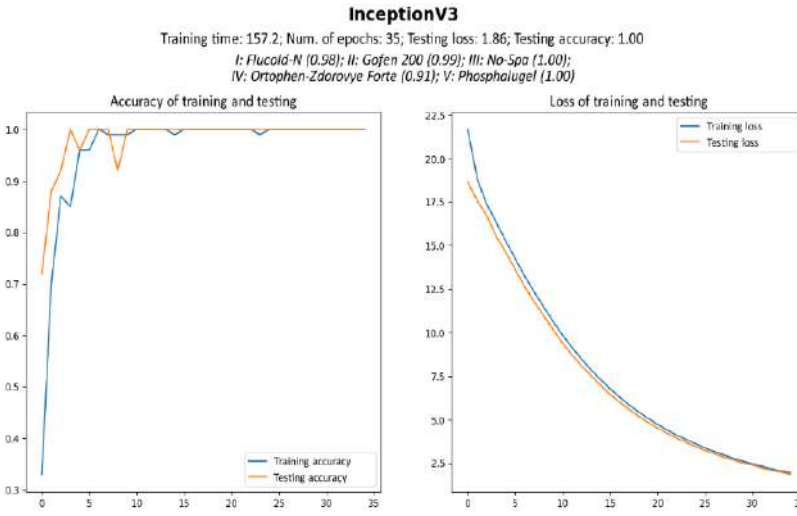


Figure 7. Training result of the model with Inception V3

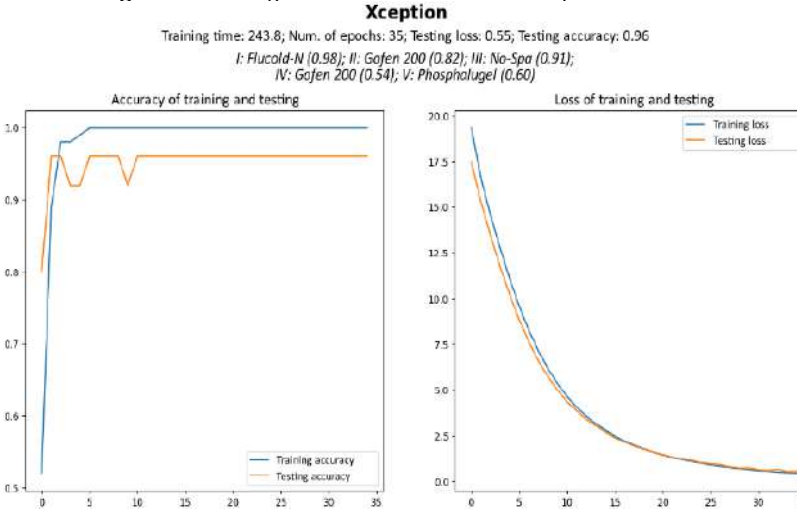


Figure 8. Training result of the model with Xception

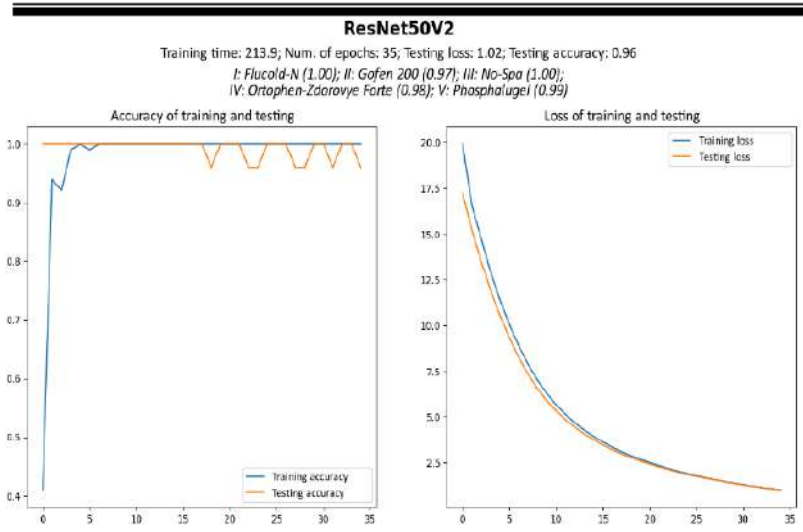


Figure 9. Training result of the model with ResNet50V2

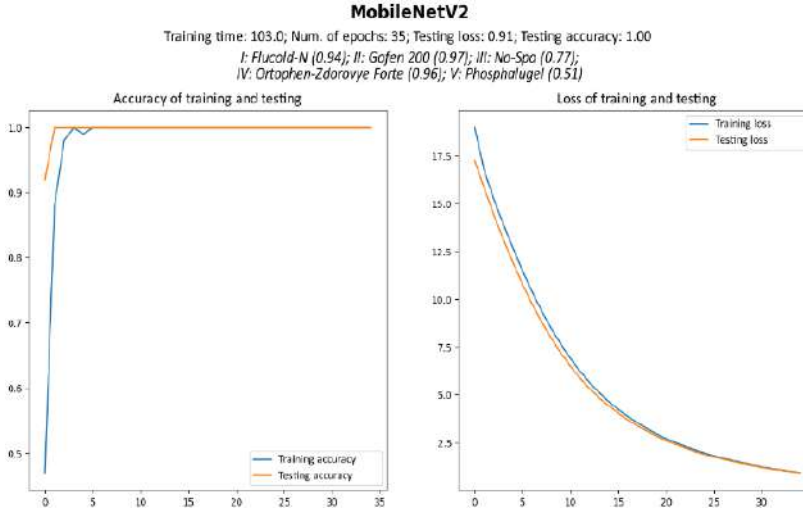


Figure 10. Training result of the model with MobileNetV2

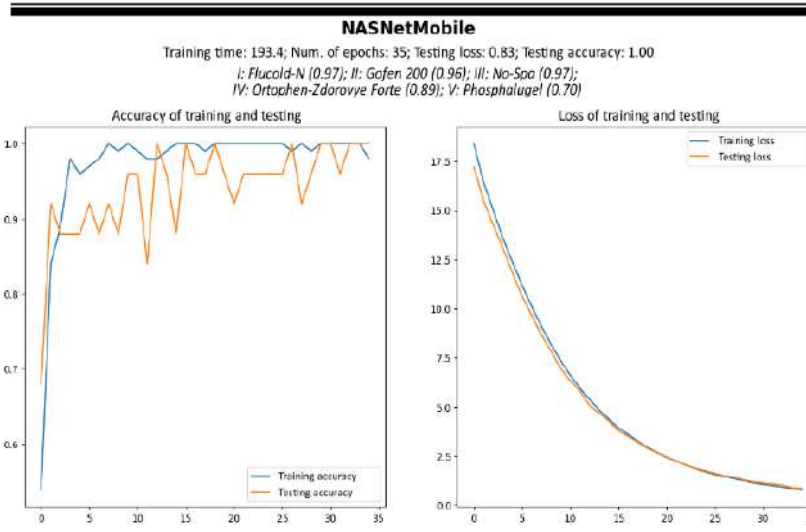


Figure 11. Training result of the model with NasNetMobile

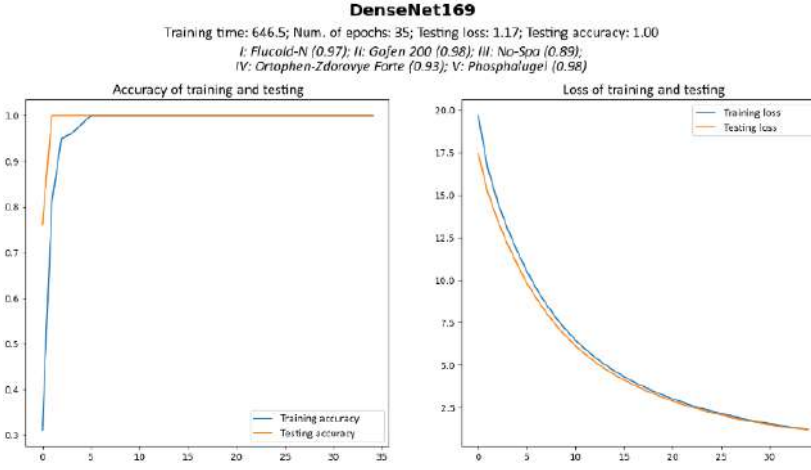


Figure 12. Training result of the model with DenseNet169

3.4 Graphical user interface

During the development of the mobile application, a user-friendly graphical interface was implemented to facilitate intuitive interaction with the core functionality of the system:

"My Medicines" tab: This tab serves as the main screen where users can view and manage their list of medications. Medicines are displayed in a sorted list. The "Add Medicine" button allows the user to quickly navigate to the medicine input page;

"Add Medicine" page: This interface enables users to input detailed information about a new medicine. It also allows users to configure reminders for each medication. Upon completing the form, the user can save the medicine entry using the "Save" button;

"View Medicine" page: Once a medicine is added, it can be viewed and edited in this interface. All previously entered fields are displayed, and users can adjust the treatment parameters or notification settings as needed;

"Recognition" tab: This tab provides a convenient way to identify medications using image recognition. Users can take a photo of the medicine packaging via the "Packaging" button. The recognized name is displayed on-screen, and users can then proceed to the "Add Medicine" page with pre-filled data based on the recognition result;

"Info" tab: This section contains information about the application and developer.

Figure 13 presents examples of the key interface screens.

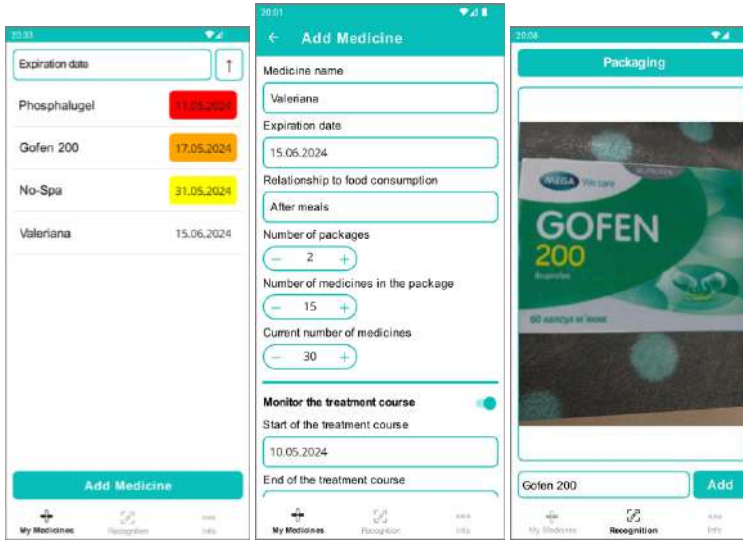


Figure 13. Key interface screens.

Testing the developed software and studying the effectiveness of the developed recognition models

For the training set, 100 images of 5 different medicine packages (20 images per medicine) were used, and for the test set, 25 images (5 per medicine). To ensure data variability and model robustness, the images were captured under diverse real-world conditions: both good and poor lighting, from different angles and distances, and with packaging in different states (e.g., in cardboard boxes and without them, if applicable). The dataset includes variations in background and orientation to simulate common usage scenarios in mobile applications.

Table 2 shows the results of training the model over 35 epochs. To further validate the model's performance, 5 additional images per class that were not included in the training or test set were used. The number of correctly classified samples is shown in Table 2, and their class probabilities are presented in Table 3.

To provide a more comprehensive assessment of model performance, standard classification metrics - accuracy, precision, recall, and F1-score- were calculated for each model on the test set. A comparison of these metrics is provided in Table 4. MediPackNet achieved an accuracy of 92%, with a precision of 93% and an F1-score of 91%. However, confusion matrix analysis revealed that MediPackNet misclassified two samples of the third class (No-Spa), leading to false negatives.

Table 2

Comparison of model training results

Model	Train. time (s)	Test loss	Test acc. (%)	Size (MB)	Number of correctly classified samples
MediPackNet	230.4	3.63	92	78.3	5 з 5
InceptionV3	157.2	1.86	100	97.5	5 з 5
Xception	243.8	0.55	96	90.8	4 з 5
ResNet50V2	213.9	1.02	96	101	5 з 5
MobileNetV2	103.0	0.91	100	18.0	5 з 5
NASNetMobile	193.4	0.83	100	39.9	5 з 5
DenseNet169	646.5	1.17	100	67.7	5 з 5

Table 3

Image classification probabilities in percentage

Model	Flucold-N	Gofen 200	No-Spa	Ortophen-Zdorovye Forte	Phosphalugel
MediPackNet	99	99	91	100	98
InceptionV3	98	99	100	91	100
Xception	98	82	91	–	60
ResNet50V2	100	97	100	98	99
MobileNetV2	94	97	77	96	51
NASNetMobile	97	96	97	89	70
DenseNet169	97	98	89	93	98

Table 4

Comparative evaluation of models using key classification metrics (%)

Model	Accuracy	Precision	Recall	F1-score
MediPackNet	92	93	92	91
InceptionV3	100	100	100	100
Xception	96	97	96	96
ResNet50V2	96	97	96	96
MobileNetV2	100	100	100	100
NASNetMobile	100	100	100	100
DenseNet169	100	100	100	100

The data demonstrates that the custom MediPackNet model was able to learn effective feature representations from a relatively small but diverse dataset. Although some pre-trained models achieved perfect accuracy, MediPackNet provided reliable performance with minimal misclassifications and offers advantages in flexibility and deployment. Its robust behavior in real-world-like conditions and interpretability made it suitable for integration into the server-side application, with further improvements expected as the dataset and architecture evolve.

The software was tested on an emulator of a mobile device with Android OS with the following characteristics: model: Google Pixel 5; OS version: Android 14.0 (API 34); screen size: 6.0 inches; screen resolution: 1080x2340 pixels, 440 dpi; processor: x86_64; memory: 1 GB; network access; access to front and back cameras; sensor support [41, 42, 43, 44].

During the testing of the mobile application, various scenarios were executed, including checking the functionality, stability, interaction with the server side, correct data storage and processing.

4 Discussion of the research results

The software was created for which a modified convolutional neural network model MediPackNet was developed, which is 92% accurate and correctly recognized all 5 test images of medicine packages. It showed good results at the level of 6 models based on already known ones, namely: InceptionV3, Xception, ResNet50V2, MobileNetV2, NASNetMobile and DenseNet169. In addition, AES, RSA data encryption methods and a combination of these algorithms were implemented. Based on the results of the analysis, it was concluded that hybrid encryption is the best for the developed software. All the planned functions were implemented in the software. Testing of the program proved that all the developed functionality works correctly.

The results of the comparative analysis show that the proposed MediPackNet model demonstrates high accuracy and efficiency. It is important to note that the model architecture was optimized to achieve this level of accuracy. Successful recognition of all test images confirms the reliability and stability of the developed

software. The implementation of data encryption methods ensures a high level of security of information transmission, which is critical for the medical field. Further research can be aimed at optimizing computational complexity and reducing data processing time, as well as expanding the functionality of the software. The results of the study are of great practical importance and can be used in various industries that require the ability to keep records of medicines and high accuracy of package image recognition.

The disadvantage of using package recognition is the need to constantly update the training set and additional training of the created model due to periodic changes in medicines packaging.

The developed application has the potential to significantly facilitate the process of managing medicines stocks and contributes to more efficient use of medical resources and optimization of procurement. In addition, the ability to track the course of treatment contributes to more accurate implementation of medical recommendations and ensures more effective health monitoring. The use of the mobile application also has a significant impact on the environment, as it reduces the amount of hazardous waste associated with improper storage and disposal of expired medicines.

In the future, it is possible to improve the created software by adding notifications about the need to purchase a medicine because it is about to expire, implementing barcode or serial number scanning, improving the package recognition model, security mechanisms, and expanding the list of medicines for recognition, providing the ability to view instructions for use of medicines, etc.

Conclusions

The subject area was analyzed, the relevance and feasibility of software development were presented, the known methods of performing the task of accounting for medicines of a universal first aid kit were considered. In the course of analyzing the analogues, their advantages and disadvantages were identified and the functionality for implementation was determined.

The chosen design pattern and the architecture of the developed software were described. The MediPackNet model for recognizing medicines by their packaging was developed, which contains convolutional layers, maximum pooling layers, flatten layers, dropout layers, and fully connected layers. The images were normalized and randomly rotated, zoomed, and horizontally flipped to increase the diversity of the data (in all created models). ELU was selected as the activation function, and the output layer uses the Softmax activation function. L2 regularization was used in the fully connected layers. There were 6 more models created, which include Inception V3, Xception, ResNet50V2, MobileNetV2, NASNetMobile, and DenseNet169, a global average aggregation layer, and fully connected layers. Based on the results of training and testing, it was decided to use the MediPackNet model, which is 92% accurate and correctly recognized all 5 test images of medicine packages. A conceptual model of the database with the “Medicine” entity was developed. Then the implemented data encryption methods

AES, RSA and a combination of these algorithms were described. Based on the results of the analysis, it was concluded that hybrid encryption is the best for the developed software. The main decisions regarding the development of the graphical user interface were also presented.

References

1. Pérez-Jover, V., Sala-González, M., Guilabert, M., & Mira, J. J. (2019). Mobile Apps for Increasing Treatment Adherence: Systematic review. Journal of Medical Internet Research, 21(6), e12505. https://doi.org/10.2196/12505.

2. Nova Advertising. (2023). Why medication dosage monitoring is critical for your health. Texas Star Pharmacy. https://www.texasstarpharmacy.com/why-medication-dosage-monitoring-is-critical-for-your-health/.

3. Hladka, O. (2022). Complete overview of medication tracking software — Intellectsoft blog. Intellectsoft Blog. https://www.intellectsoft.net/blog/medication-tracking-software/.

4. Abdulhamid, M., & Matongo, A. (2020). Mobile phone and barcode scanner. Scientific Bulletin of Electrical Engineering Faculty, 20(1), 25–28. https://doi.org/10.2478/sbeef-2020-0105.

5. Avola, D., Cinque, L., Fagioli, A., Foresti, G. L., Marini, M. R., Mecca, A., & Pannone, D. (2022). Medicinal boxes recognition on a deep transfer learning augmented reality mobile application. In Lecture notes in computer science (pp. 489–499). https://doi.org/10.1007/978-3-031-06427-2_41.

6. Archana, R., & Jeevaraj, P. S. E. (2024). Deep learning models for digital image processing: a review. Artificial Intelligence Review, 57(1). https://doi.org/10.1007/s10462-023-10631-z.

7. Lu, L., Di, L., & Ye, Y. (2014). A Decision-Tree Classifier for Extracting Transparent Plastic-Mulched Landcover from Landsat-5 TM Images. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 7(11), 4548–4558. https://doi.org/10.1109/jstars.2014.2327226.

8. Wu, N. J. (2012). Efficient HIK SVM learning for image classification. IEEE Transactions on Image Processing, 21(10), 4442–4453. https://doi.org/10.1109/tip.2012.2207392.

9. Taherkhani, A. (2010). Recognizing Sorting Algorithms with the C4.5 Decision Tree Classifier. In IEEE 18th International Conference on Program Comprehension (pp. 72–75). https://doi.org/10.1109/icpc.2010.11.

10. Xu, N. Y., Sonka, M., McLennan, G., Guo, N. J., & Hoffman, E. (2006). MDCT-based 3-D texture classification of emphysema and early smoking related lung pathologies. IEEE Transactions on Medical Imaging, 25(4), 464–475. https://doi.org/10.1109/tmi.2006.870889.

11. Srensen, L., Shaker, S. B., & De Bruijne, M. (2010). Quantitative analysis of pulmonary emphysema using local binary patterns. IEEE Transactions on Medical Imaging, 29(2), 559–569. https://doi.org/10.1109/tmi.2009.2038575.

12. Ponnusamy, R., Sathyamoorthy, S., & Manikandan, K. (2017). A Review of Image Classification Approaches and Techniques. International Journal of Recent

Trends in Engineering and Research, 3(3), 1–5. https://doi.org/10.23883/ijrter.2017.3033.xts7z.

13. Pineda-Jaramillo, J. D. (2019). A review of Machine Learning (ML) algorithms used for modeling travel mode choice. DYNA, 86(211), 32–41. https://doi.org/10.15446/dyna.v86n211.79743.

14. Alnuaimi, A. F., & Albaldawi, T. H. (2024). An overview of machine learning classification techniques. BIO Web of Conferences, 97, 00133. https://doi.org/10.1051/bioconf/20249700133.

15. Medical Directory of Diseases and Medicines – Apps on Google Play. (2024). https://play.google.com/store/apps/details?id=ru.medical.spravochnik.

16. First aid kit. Accounting for medicines. – Apps on Google Play. (2024). https://play.google.com/store/apps/details?id=ru.decsoft.myaidkit.

17. Medication Control – Apps on Google Play. (2024). https://play.google.com/store/apps/details?id=ua.com.hitmax.liki.

18. My Medicine – Health Log. – Apps on Google Play (2024). https://play.google.com/store/apps/details?id=health.doctor.pill.my_medicine.

19. Medscape – Apps on Google Play. (2024). https://play.google.com/store/apps/details?id=com.medscape.android.

20. Medisafe Pill & Med Reminder – Apps on Google Play. (2024). https://play.google.com/store/apps/details?id=com.medisafe.android.client.

21. MyTherapy Pill Reminder – Apps on Google Play. (2024). https://play.google.com/store/apps/details?id=eu.smartpatient.mytherapy.

22. Medsbit Medication Tracker – Apps on Google Play. (2024). https://play.google.com/store/apps/details?id=all.pillreminder.medicationtracker.medsalarm.

23. Saha, S. (2023). A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way. https://saturncloud.io/blog/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way/.

24. Li, H., Krček, M., & Perin, G. (2020). A comparison of weight initializers in Deep Learning-Based Side-Channel analysis. In Lecture notes in computer science (pp. 126–143). https://doi.org/10.1007/978-3-030-61638-0_8.

25. Narein, A. (2021). Inception V3 Model Architecture. https://iq.opengenus.org/inception-v3-model-architecture/.

26. Fabien, M. (2019). XCeption Model and Depthwise Separable Convolutions. https://maelfabien.github.io/deeplearning/xception/.

27. Datagen Tech. ResNet-50: The Basics and a Quick Tutorial. https://datagen.tech/guides/computer-vision/resnet-50/.

28. Badgujar, S. (2021). Creating MobileNetsV2 with TensorFlow from scratch. https://medium.com/analytics-vidhya/creating-mobilenetsv2-with-tensorflow-from-scratch-c85eb8605342.

29. Reda, M., Suwwan, R., Alkafri, S., Rashed, Y., & Shanableh, T. (2022). AgroAID: a mobile app system for visual classification of plant species and diseases using deep learning and TensorFlow Lite. Informatics, 9(3), 55. https://doi.org/10.3390/informatics9030055.

30. Vulli, A., Srinivasu, P. N., Sashank, M. S. K., Shafi, J., Choi, J., & Ijaz, M. F. (2022). Fine-Tuned DenseNET-169 for breast cancer metastasis prediction using FastAI and 1-Cycle Policy. Sensors, 22(8), 2988. https://doi.org/10.3390/s22082988.

31. Alshannaq, O., Alsayaydeh, J. A. J., Hammouda, M. B., Ali, M. F., Alkhashaab, M. A. R., Zainon, M., & Jaya, A. S. M. (2022). Particle swarm optimization algorithm to enhance the roughness of thin film in tin coatings. ARPN Journal of Engineering and Applied Sciences, 17(2), 186–193.

32. Oliinyk, A., Fedorchenko, I., Stepanenko, A., Rud, M., & Goncharenko, D. (2020). Implementation of evolutionary methods of solving the travelling salesman problem in a robotic warehouse. In Lecture notes on data engineering and communications technologies (pp. 263–292). https://doi.org/10.1007/978-3-030-43070-2_13.

33. Izonin, I., Tkachenko, R., Shakhovska, N., & Lotoshynska, N. (2021). The Additive Input-Doubling Method Based on the SVR with Nonlinear Kernels: Small Data Approach. Symmetry, 13(4), 612. https://doi.org/10.3390/sym13040612.

34. Syromiatnikov, A., & Weyns, D. (2014). A Journey through the Land of Model-View-Design Patterns. In 2014 IEEE/IFIP Conference on Software Architecture (WICSA) (pp. 21–30). https://doi.org/10.1109/wicsa.2014.13.

35.Franklin, R. (2022). AES vs. RSA Encryption: What Are the Differences? https://www.precisely.com/blog/data-security/aes-vs-rsa-encryption-differences.

36.Froehlich, A. (2021). What is ciphertext feedback (CFB)? https://www.techtarget.com/searchsecurity/definition/ciphertext-feedback.

37. Hammod, D. N., Al-Rawi, M., & Abdulah, H. S. (2016). An enhancement method based on modifying CFB mode for key generation in AES algorithm. Engineering and Technology Journal, 34(6B), 759–768. https://doi.org/10.30684/etj.34.6b.5.

38. Medium, Introduction to RSA. (2019). https://medium.com/@c0D3M/introduction-to-rsa-e8cb39af508e.

39. LinkedIn. (2024). How do you use AES and RSA together for hybrid encryption schemes? https://www.linkedin.com/advice/0/how-do-you-use-aes-rsa-together-hybrid-encryption-schemes.

40. Rudkovskyi, O. R., & Kirichek, G. G. (2020). Interaction support system of network applications. In 3rd Workshop for Young Scientists in Computer Science & Software Engineering (pp. 11–23).

41. Alshannaq, O., Alsayaydeh, J. A. J., Hammouda, M. B. A., Ali, M. F., Alkhashaab, M. A. R., Zainon, M., & Jaya, A. S. M. (2022). Particle swarm optimization algorithm to enhance the roughness of thin film in tin coatings. ARPN Journal of Engineering and Applied Sciences, 17(2), 186–193.

42. Alsayaydeh, J. A. J., Khang, A. W. Y., Indra, W. A., Shkarupylo, V., Hossain, A. K. M. Z., Saravanan, N., & Pusppanathan, J. B. (2019). Development of Vehicle Door Security using Smart Tag and Fingerprint System. International Journal of Engineering and Advanced Technology, 9(1), 3108–3114. https://doi.org/10.35940/ijeat.e7468.109119.

43. Alsayaydeh, J. A. J., Aziz, A., Rahman, A. I. A., Abbasi, M. I., & Khang, A. W. Y. (2021). Development of Programmable Home Security using GSM System for Early Prevention. ARPN Journal of Engineering and Applied Sciences, 16(1), 88–97.

44.Oliinyk, A., Fedorchenko, I., Donenko, V., Stepanenko, A., Korniienko, S., & Kharchenko, A. (2020). Development of an evolutionary optimization method for financial indicators of pharmacies. International Journal of Computing, 19(3), 449–463. https://doi.org/10.47839/ijc.19.3.1894.

НЕЙРОМЕРЕЖЕВА МОДЕЛЬ ДЛЯ ОПТИМІЗОВАНОГО КЕРУВАННЯ ЛІКАРСЬКИМИ ЗАСОБАМИ В УНІВЕРСАЛЬНІЙ АПТЕЧЦІ

Є. Федорченко[1][0000-0003-1605-8066], Dr.Sci. А. Олійник[2][0000-0002-6740-6078], К. Паничук[3][0009-0009-4340-6688], О. Коршун[4][0009-0009-9178-793X], Ph.D. О. Степаненко[5][0000-0002-0267-7440]

[1,2,3,5]Національний університет «Запорізька політехніка», Україна
[4]«OPTIME», Грузія
EMAIL: [1]evg.fedorchenko@gmail.com, [2]olejnikaa@gmail.com, [3]panichuck.kat@gmail.com, [4]oleh.korshun@gmail.com, [5]alex@zntu.edu.ua

Анотація. *Було розроблено модифіковану модель згорткової нейронної мережі MediPackNet з точністю 92%, яка правильно розпізнала всі 5 тестових зображень лікарських засобів. Вона продемонструвала хороші результати на рівні 6 моделей, побудованих на основі відомих архітектур, а саме: InceptionV3, Xception, ResNet50V2, MobileNetV2, NASNetMobile та DenseNet169. Крім того, були реалізовані методи шифрування даних AES, RSA та їх комбінація. За результатами аналізу встановлено, що найефективнішим для розробленого програмного забезпечення є гібридне шифрування. Створено мобільний застосунок для обліку лікарських засобів універсальної аптечки, забезпечено його стабільну роботу та можливість для подальшого розвитку. Розроблений застосунок здатен суттєво полегшити процес управління запасами лікарських засобів, сприяє ефективнішому використанню медичних ресурсів і оптимізації закупівель. Крім того, можливість відстеження курсу лікування забезпечує точніше дотримання медичних рекомендацій і сприяє ефективнішому моніторингу стану здоров'я. Використання мобільного застосунку також має значний вплив на навколишнє середовище, оскільки зменшує кількість небезпечних відходів, пов'язаних із неналежним зберіганням і утилізацією прострочених лікарських засобів.*

Ключові слова: *облік, лікарські засоби, аптечка, пакування, розпізнавання , мобільний застосунок, .NET MAUI, Python.*

Section 3. Modeling and software engineering

UDC 004.03
DOI https://doi.org/10.36059/978-966-397-538-2-18

INTEGRATED MODELING OF RELIABILITY AND MAINTENANCE OF SHIP'S POWER PLANT EQUIPMENT CONSIDERING DEGRADATION AND OPERATIONAL CONDITIONS

Dr.Sci. V. Vychuzhanin[1][0000-0002-6302-1832], **Ph.D. A. Vychuzhanin**[2][0000-0001-8779-2503]
Odessa Polytechnic National University, Ukraine
EMAIL: [1]v.v.vychuzhanin@op.edu.ua, [2]vychuzhanin.o.v@op.edu.ua

Abstract. *Enhancing the reliability and economic efficiency of marine vessels requires the development of failure prediction tools capable of accounting for real operating conditions and equipment degradation processes. This paper presents an integrated approach to the quantitative assessment of the reliability of key components of a ship's power plant (SPP) over a 25,000-hour operational interval. The methodology combines probabilistic, degradation-based, and simulation models while incorporating operational parameters such as temperature, relative load, and maintenance intervals. Four classes of failure models are developed and compared: an exponential model, a variable-intensity model (non-stationary Weibull process), a four-state Markov scheme, and an event-driven Monte Carlo simulation model. The calculations are performed for the main engine, generator, cooling system, and shipboard power station. The root mean square error (RMSE) of failure prediction was 0.05 for the simulation model, 0.08 for the Markov model, and 0.12 for the exponential model. An integrated model quality criterion incorporating RMSE, AIC, BIC, and χ^2 confirmed the advantage of the hybrid simulation-Markov approach. A comparative economic analysis showed that regular maintenance at 5,000-hour intervals reduces total costs by more than 4.5 times compared to reactive repair strategies. The practical value of the method lies in its applicability within digital twins and intelligent decision support systems. Future developments include expanding the component base of the model, integrating with real-time data streams from ship monitoring systems, and applying machine learning techniques for automatic parameter adjustment.*

Keywords: *predictive diagnostics; digital twin; remaining useful life; interval-based maintenance; Markov model; simulation forecasting; economic failure assessment*

1 Introduction

Modern ship's power plants operate under elevated operational loads, thermal and vibrational stresses, which necessitate reliable prediction of their technical condition and maintenance requirements [1, 2, 3]. As the duration of autonomous voyages

increases and onboard power systems become more complex, the demands for fault tolerance and cost-effective maintenance continue to grow. Under these conditions, an integrated approach to long-term reliability assessment of SPP components accounting for degradation, probabilistic failure characteristics, and operational influences becomes especially relevant [4].

In recent years, there has been a growing interest in the digitalization of technical diagnostics in the maritime industry [5]. One of the key directions in modern monitoring is the use of digital twins, which enable near-real-time modeling of marine systems and prediction of potential failures. While this study does not focus on the development of a digital twin as a software platform, it does establish a mathematical foundation for its prognostic module. The developed models incorporate physically interpretable dependencies of failure intensity on operational factors. In particular, the model integrates temperature effects (e.g., via the Arrhenius exponential function for generators), load-related parameters (coefficients reflecting nominal value exceedance), and environmental conditions (e.g., salinity and coolant temperature for the cooling system). These dependencies are implemented as parameterized functions calibrated against field data and reflect key degradation mechanisms such as thermal aging, fatigue damage accumulation, and aggressive environmental exposure. Zocco et al. [6] emphasize that digital twins allow integration of monitoring data with predictive algorithms; however, the practical implementation of such solutions remains limited, in part due to the absence of a unified methodology. Stadtmann et al. [7] demonstrate the application of digital twins for offshore wind turbines, highlighting the potential of the technology, though the focus is primarily on renewable energy rather than marine systems. Special attention in the literature is given to the use of machine learning in diagnostics and prediction of equipment condition. Polverino et al. [8], in a systematic review, show that machine learning methods are successfully applied for estimating remaining useful life (RUL) and anomaly detection. However, these approaches are often detached from real risk evaluation and cost considerations. Studies focusing on the integration of digital solutions in the maritime sector, such as Kaklis et al. [9] underscore the need for comprehensive analysis encompassing not only failure modeling but also lifecycle management. Some research highlights the resilience of ship systems under intensive operation. Nezhad et al. [10] stress the importance of predictive maintenance based on big data analysis, while also noting the lack of quantitative models that consider both degradation dynamics and the economic consequences of technical decisions. Similarly, Mavrakos et al. [11] propose digital tools to support energy-saving strategies, pointing to the need for adaptive models capable of considering operational constraints. Additional recent studies support the relevance of a systemic approach to the prediction of technical condition in marine components. Liang et al. [12], in a review from a classification society perspective, emphasize that implementing prognostics and health management (PHM) methods requires integration of degradation models with regulatory frameworks. Han et al. [13] demonstrate how a variational autoencoder based on LSTM can detect marine component failures; however, their model is

mostly oriented toward anomaly detection rather than quantitative reliability prediction. Xiao et al. [14], in their PHM research for industrial assets, propose a digital twin architecture that combines remaining life prediction with risk-based maintenance planning - a concept applicable to marine systems as well. Finally, Cui et al. [15] develop a digital twin for a marine diesel engine and demonstrate its capability to enhance maintenance efficiency and reduce downtime, though their focus lies in platform-level integration rather than formal reliability modeling.

A review of current publications shows that despite the active development of digital diagnostics technologies, the issue of long-term reliability of SPP components under real-world wear and overload conditions remains insufficiently addressed. Moreover, there is a noticeable lack of studies that integrate failure prediction with economic evaluation of maintenance strategies. Unlike most existing research focusing on localized degradation scenarios or isolated diagnostic aspects, this article centers on the holistic integration of reliability and economic analysis, providing a foundation for informed decision-making under real marine operating conditions.

This study aims to fill this gap by offering a comprehensive analysis of the reliability of core SPP components over a 25,000-hour operating horizon. The approach is based on simulation modeling, Markov processes, and degradation models that account for wear dynamics. Special attention is given to the influence of operational factors (load, temperature, maintenance intervals) on failure probability, as well as the comparative economic efficiency of various maintenance strategies. The results obtained can be used in the development of predictive maintenance programs, resource planning, and life cycle optimization of equipment in marine engineering.

The objective of this study is to develop and justify an integrated approach to the long-term reliability analysis of SMPP components, taking into account degradation dynamics, the influence of operational factors, and the economic efficiency of maintenance strategies.

To achieve this objective, the following tasks are addressed:

1. Develop mathematical models for reliability prediction of SPP components, including exponential, degradation-based, Markov, and simulation-based approaches applicable to extended operational intervals.

2. Describe and implement component-specific failure rate dependencies on operational factors such as mechanical load, temperature, and maintenance parameters.

3. Construct a hybrid Markov–degradation model accounting for transitions between technical states (operational, degrading, pre-failure, and failed), with parameters that depend on accumulated wear.

4. Implement simulation modeling of operational scenarios using the Monte Carlo method to estimate the distribution of failure times and the variability of technical life.

5. Formulate a model selection criterion that combines prediction accuracy (RMSE), information-theoretic metrics (AIC, BIC), and agreement with empirical data (χ^2).

6. Evaluate the economic efficiency of different maintenance strategies by comparing total costs under regular and reactive servicing regimes for key components.

7. Develop recommendations for model application based on operating conditions and data availability, and assess their applicability as part of prognostic modules in digital decision support systems.

2 Materials and Methods

The objects of this study are the key components of the SPP, including the main engine, generator, cooling system, and shipboard power station. These elements are subject to long-term wear, vibrational, and thermal loads, which makes the analysis of their reliability over an operational interval of up to 25,000 hours particularly relevant. This duration is typical for resource planning and scheduled maintenance.

The initial data for the analysis are generalized statistical records of failure frequencies documented in maritime practice and technical literature, including the OREDA failure databases. Additionally, typical operational modes, maintenance intervals, and expert assessments reflecting the influence of load and temperature conditions on equipment degradation were taken into account.

Four different approaches were used to model reliability. The exponential model served as a baseline and assumed a constant failure rate, without accounting for wear accumulation. More realistic scenarios were described using analytical degradation models, in which the failure intensity increases over time following a power-law relationship. The third method involved a Markov model that represents probabilistic transitions between technical states from operational to degrading, then to pre-failure and failure states. Finally, simulation modeling was applied to reproduce complex operational conditions and to construct failure scenarios under the stochastic nature of external influences. Within this approach, modeling was implemented using the Monte Carlo method with variation of operational parameters.

The comparative accuracy of the listed models was assessed using the root mean square error (RMSE), which allows for a quantitative comparison of forecasts against reference scenarios. The analysis results showed that simulation modeling demonstrated the lowest error, whereas the exponential model exhibited the greatest deviations over extended operational periods.

Special attention in the study was given to analyzing the influence of operational factors on the reliability of SPP components. Three key factors were considered: load regimes (nominal, elevated, emergency), thermal impacts (coolant temperature, oil temperature, cylinder gas temperature), and maintenance frequency. Graphs were constructed showing the dependence of reliability on each of these factors, and components were ranked according to their sensitivity to various operational conditions.

Finally, an evaluation of the economic efficiency of different maintenance strategies was conducted. Two scenarios were compared: absence of preventive measures and regular maintenance at 5,000-hour intervals. The calculation included both direct costs of failure remediation and indirect losses associated with forced downtime. The results showed that a systematic maintenance approach reduces total costs by a factor of 4 to 5 compared to a reactive maintenance model.

The proposed methodological approach enables not only the assessment of MPP component reliability over a long time horizon, but also the justification of economically efficient maintenance decisions based on modeling, statistical data, and simulation scenarios.

3 Results

To assess the long-term reliability of SPP components, the following reliability prediction models are used: exponential reliability model, applied to components with a constant failure rate, where the probability of failure depends only on operating time; degradation models - account for the accumulation of damage and changes in failure intensity over time; Markov failure model - tracks transitions of components between different operable states, considering probabilistic changes; simulation-based reliability models used for analyzing long-term operational scenarios, simulating the impact of various operational factors.

Exponential model with a constant failure rate. For components operating under stable conditions without pronounced degradation or aging, the simplest failure model based on the exponential law is applicable. This model describes non-repairable processes with a constant failure rate λ, which corresponds to the steady-state operational phase where the failure intensity is assumed to remain constant [16]:

$$R(t) = e^{-\lambda t},\ t \geq 0,$$

where $R(t)$ is the probability of failure-free operation at time t;

λ is the failure rate (h^{-1}), assumed to be constant over time

The parameter λ is estimated based on the total operating time T_Σ and the number of observed failures k during this period. A biased maximum likelihood estimator is used [13]:

$$\hat{\lambda} = \frac{k}{T_\Sigma}, \quad Var[\hat{\lambda}] = \frac{k}{T_\Sigma^2}$$

Based on the estimated failure rate, the mean time to failure (MTTF) is calculated using the formula:

$$MTTF_{\exp} = \frac{1}{\hat{\lambda}}$$

Despite its simplicity, the exponential model serves as a useful baseline for comparison with more advanced approaches. It is applied, in particular, to components with high reliability operating under stable conditions. However, this model does not account for degradation processes, recovery after failure, or

variations in operating loads, which limits its applicability for long-term prediction under real marine operating conditions.

Degradation models for SPP equipment

The long-term reliability assessment of SPP components requires the inclusion of damage accumulation processes. In this study, component-specific degradation models are applied, reflecting the dependence of failure rate on time and operational factors.

General approaches to degradation modeling

The failure rate of a component at time *t*, denoted $\lambda(t)$, is modeled using various functional forms:

$$\lambda(t) = \lambda_0 + \alpha t^{\beta},$$

where λ_0 - initial failure rate at $t = 0$, reflecting baseline component quality [1/h];

α - *degradation growth coefficient* ($h^{-1} \cdot h^{-\beta}$);

β - power-law exponent ($\beta>1$ - indicates accelerated degradation, $\beta<1$ - indicates deceleration);

α, β - parameters obtained by regression on failure data

Weibull-Based Degradation Model (NHPP):

$$\lambda_{\deg}(t) = \frac{\beta}{\Theta}\left(\frac{t}{\Theta}\right)^{\beta-1}, \ \beta > 1$$

Parameter estimates are obtained using the maximum likelihood method:

$$\hat{\beta} = \left[\frac{\sum_{i=1}^{n} \lg t_i^{-\frac{n}{\beta}}}{\sum_{i=1}^{n} t_i^{\beta}}\right]^{-1}, \quad \hat{\Theta} = \left(\frac{1}{n}\sum_{i=1}^{n} t_i^{\beta}\right)^{1/\hat{\beta}}$$

Combined load and temperature model:

$$\lambda(t, L, T) = \lambda_0\left[1 + k_L\left(L / L_{nom}\right)^{m}\right]\exp\left\{T - T_{ref}\right\},$$

where L - relative load (0…1);

T - current operating temperature of the working medium (°C);

L_{nom}, T_{ref} - nominal values of load and temperature;

m, k_L, η - calibration parameters obtained from experimental data

The degradation of SPP equipment depends simultaneously on load and temperature. The rate of damage accumulation or increase in failure intensity is not constant but is a function of operational impacts. In the main engine, degradation affects the piston group, crankshaft, and cylinder liners. The load is characterized by propeller resistance torque, overload, and rotation frequency. Temperature-related factors include oil, combustion gases in the cylinders, and cooling water. Elevated oil temperature reduces viscosity, accelerates wear of journal bearings and

crankshaft surfaces, increases clearances, and leads to higher vibration levels - all of which contribute to engine failure.

In the cooling system, degradation affects heat exchangers, pumps, and pipe joints. The load is defined by pressure differentials and start-stop frequency. The critical external parameters are seawater temperature and overheating of the circulating water. These factors promote scale formation, corrosion, cavitation, and loss of tightness. Thus, the degradation model for the cooling system depends on both temperature and environmental aggressiveness.

In the generator, degradation primarily occurs in the stator/rotor windings, insulation, and bearings. The winding temperature governs insulation aging. Load is defined by overcurrent conditions and frequent on/off cycles, which accelerate thermal aging and thermal cycling, leading to insulation breakdown.

For each subsystem of the SPP, a dedicated degradation model is applied that accounts for the corresponding operational impacts (mechanical, thermal, electrical, etc.) using a generalized functional form of the failure intensity $\lambda_i(t, X_i)$, where X_i is the vector of external influences on the i-th component.

Main engine (ME) [16]:

$$\lambda_{ME} = (\lambda_0 + \rho t)\left[1 + k_L\left(L/L_{nom}\right)^m\right]\exp\{\eta(T_{oil} - T_{ref})\},$$

where ρ - coefficient of failure rate growth with runtime (h^{-2});

T_{oil} - oil temperature (°C);

λ_0 - baseline failure rate

An integral wear accumulation model is also used:

$$z(t) = \alpha_1 \cdot L(t)^m + \alpha_2 \cdot \exp[b \cdot (T_{oil} - T_{norm})], \qquad \lambda(t) = \lambda_0(1 + z(t))$$

Cooling system (CS) [17]:

$$\lambda_{CS} = \lambda_0 \cdot \left[1 + k_T(T_{CW} - T_0)^\alpha\right]\left[1 + k_{NaCl}C_{NaCl}\right],$$

where T_{CW} - temperature of the circulating water (°C);

T_0 - reference temperature;

C_{NaCl}- salt concentration (ppm);

k_T, k_{NaCl}, α - empirical parameters

Generator (GEN):

$$\lambda_{GEN} = \lambda_0 \exp\left[\frac{E_a}{k_\beta}\left(\frac{1}{T_w} - \frac{1}{T_{ref}}\right)\right](1 + \xi \cdot I/I_{nom}),$$

where T_w - winding temperature (K);

E_a - activation energy of insulation aging;

k_β - Boltzmann constant;

I - load current;

ξ - overload coefficient;

T_w, T_{ref} - winding temperature and reference (baseline) temperature (in Kelvin)

Ship Power Station ((ISO 12110-2:2013 Metallic materials) [18]:

$$\lambda_{SPS} = \lambda_0 + \alpha_v \cdot \nu(t)^2 1 + \alpha_f \cdot f(t),$$

where $v(t)$ - vibration amplitude;

$f(t)$ - switching frequency (on/off cycles);

α_v, α_f - empirical parameters reflecting the impact of vibration and switching loads

The proposed models allow: accounting for the influence of operational factors on component reliability; flexible adaptation to different subsystems and operating conditions; easy integration into simulation and Markov-based prognostic frameworks; suitability for implementation in predictive modules of digital twins. Model parameters are identified based on field data (failure logs, OREDA, onboard recorders), and accuracy is validated using RMSE, χ^2, and information criteria such as AIC/BIC.

Markov model

The Markov model describes transitions between states:operational → degrading → pre-failure → failure.The transition probability matrix P_{ij} is constructed based on historical data. A SPS component is modeled as a Continuous-Time Markov Chain (CTMC) with four states:

S = {0 – operational, 1 – degrading, 2 – pre-failure, 3 – failure}.

The infinitesimal intensity matrix Q (Hoyland & Rausand, 2004) [19]:

$$Q=\begin{pmatrix} -(\mu_0+\gamma_0) & \mu_0 & 0 & \gamma_0 \\ 0 & -(\mu_1+\gamma_1) & \mu_1 & \gamma_1 \\ 0 & 0 & -\gamma_2 & \gamma_2 \\ 0 & 0 & 0 & 0 \end{pmatrix}, Q \in R^{4X4}$$

where μ_0,μ_1 - gradual degradation transitions: $0 \to 1$ and $1 \to 2$;

γ_0,γ_1,γ_2 - abrupt failures from states 0, 1, and 2, respectively

Mean Time to Failure (MTTF) [19]. For stationary Q, MTTF is computed using the fundamental matrix N:

$$MTTF_{MC} = e_0^T e_0 \left(-Q_{3x3}^{-1}\right)1,$$

where Q_{3x3} - upper left 3×3 submatrix of Q;

1^T=[111]. - vector of ones;

$e_0^T = [100]$ - initial state vector (component starts in operational state)

Non-stationary (degradation-based) CTMC. In this case, transition intensities depend on accumulated wear $z(t)$:

$$z(t) = q(L(t), T(t)), z(0),$$

$$\mu_0(t) = \mu_0^*[1 + \alpha \cdot z(t)], \gamma_0(t) = \gamma_0^*[1 + \beta \cdot z(t)],$$

where $q(\cdot)$ - function linking current load *L(t)* and temperature *T(t)* with wear accumulation;

μ_0^*, γ_0^* - nominal failure intensities under base conditions;

α,β - degradation acceleration coefficients

General CTMC definition. A Continuous-Time Markov Chain is defined by: a state space *S* = {0, 1, 2, ..., n}; a transition intensity matrix *Q* = [q_{ij}], where: $q_{ij} \geq 0$ if $i \neq j$, representing the transition intensity from state *i* to state *j*; $q_{ii} = -\sum_{i \neq j} q_{ij}$, i.e., the diagonal elements are negative and equal to the negative sum of outgoing intensities

Interpretation of β:

β<1 - decelerated increase in failure intensity (e.g., under passive degradation);

β=1 - linear increase: failure rate grows proportionally with time;

β>1 - accelerated increase: typical for fatigue, aging, fouling, and wear

The parameters α and β reflect the physics of degradation and are either assigned empirically (based on operational data) or calibrated via regression.

When *Q* = *Q(t)*, the state probabilities are determined by a time-ordered matrix exponential:

$$P(t) = e_0^T \cdot T \cdot \exp(\int_0^t Q(t)dt), R(t) = 1 - [P(t)]_4,$$

where *T* - time-ordering operator;

$T \cdot \exp(\int_0^t Q(t)dt) \in R^{nxn}$ - transition probability matrix;

$[P(t)]_4$ - probability of being in the absorbing "failure" state

Numerical computation is performed using piecewise constant interval approximation or the uniformization algorithm. A stationary CTMC is recommended for stable conditions, while a non-stationary model is better suited for variable loads and temperatures.

Simulation-Based Model

The simulation model is implemented using the Monte Carlo method [20] with N = 10,000 runs. In each simulation run, the following variables are randomly sampled: *L* - relative load (as a fraction of nominal); *T* - operating medium temperature; *ΔTO* - preventive maintenance interval.

The simulation aims to compute: estimated reliability function $\hat{R}_{sim}(t)$; Expected failure rate $\hat{z}_{sim}(t)$; accuracy metrics (e.g., RMSE) by comparing predictions to observed data.

For the *j*-th run (j = 1, ..., N), the condition vector is formed:

$$X^{(j)} = (L^{(j)}, T^{(j)}, \Delta T0^{(j)}),$$

where $(L^{(j)}, T^{(j)}, \Delta T0^{(j)})$ are drawn from empirical distributions based on operational logs

The failure intensity function is selected accordingly:

$$X^{(j)}(t) = f\left(X^{(j)}, t\right),$$

A composite model selection criterion Ψ is used, minimizing a weighted sum of RMSE, AIC, BIC, and χ^2. This ensures a balanced evaluation of prediction accuracy, model complexity, and statistical fit over long-term reliability forecasts.

Random sequences of load and temperature are modeled as Rainflow histograms:

Cyclegrams of the form (L_{jk}, τ_{jk}) (k=1…K_j) are generated. Then:

1. Cumulative fatigue damage (Miner's rule) is calculated:

$$D^{(j)}(t) = \sum_{k=1}^{K_j(t)} t_{j,k} \Big/ N_f(L_{j,k}), N_f(L_j) = K \cdot L^{-m} \ ;$$

2. A failure is recorded if $D(t) \geq 1$, or if the simulation reaches an absorbing failure state in the CTMC.

3. N = 10^4 Monte Carlo runs are executed to estimate $\hat{R}_{sim}(t)$ and confidence bounds.

Model selection criterion. For each subsystem, the following were computed: RMSE, AIC, BIC, and χ^2 [21], reflecting the model's agreement with observed failure statistics.

The composite criterion:

$$\psi = \omega_1 RMSE + \omega_2 AIC + \omega_3 BIC + \omega_4 \chi^2, \sum_{i=1}^{4} \omega_i = 1,$$

where ω_i - weights are set by experts, provides a balanced selection of the optimal model based on:

$RMSE$ - accuracy of failure prediction on test data;

AIC - tradeoff between goodness of fit and model complexity;

BIC - Bayesian Information Criterion;

$\chi 2$ - goodness-of-fit test comparing model predictions to actual failure observations

The combined criterion Ψ enables the justified comparison of models with different structures, allowing for a balanced evaluation of accuracy, complexity, and realism. It supports a transparent selection of the most suitable failure prediction model for SPPs. In this study, the generalized criterion Ψ, which integrates prediction accuracy, information criteria (AIC/BIC), and agreement with field data, was used for optimal model selection.

Economic validation. For the main engine, as the most critical unit, a life-cycle cost model was constructed:

$$C = C_{TO}(\Delta) + C_{rep} + C_{down}[1 - R(t)],$$

where $C_{TO}(\Delta)$ - scheduled maintenance costs with interval Δ;

C_{rep} - capital repair costs;

C_{down} - downtime losses associated with unrealized reliability levels

The optimal value of Δ was found numerically. Regular maintenance every 5,000 hours reduces total costs by 4–5 times compared to a reactive repair scenario. The integral criterion Ψ confirms the superiority of the simulation-based scheme across all SPP components. This method accounts for the nonlinear effects of load and temperature, allows for the construction of reliability confidence intervals, and enables economic optimization of the maintenance schedule.

The application of four different reliability modeling approaches is not redundant but rather a necessary strategy, driven by the diversity of technical conditions, operating regimes, and required prediction accuracy. First, each model targets its specific domain of applicability.

The exponential reliability model is effective for components in the stable operation phase, with constant failure intensity. It is easy to implement and applicable when data availability is limited.

Degradation models allow for the consideration of damage accumulation and changing failure intensity, which is critical for components exposed to variable loads and temperatures, fatigue, or aging.

Markov models are useful when there is a need to describe discrete health states from operable to failed accounting for intermediate transitions with different probabilities. Simulation models provide the capability to analyze complex operational scenarios involving multiple random factors, such as load cyclegrams, maintenance intervals, temperature variability, and environmental aggressiveness. Second, model choice directly impacts prediction accuracy. The comparative analysis showed that RMSE values can vary by more than a factor of two between methods, and a model yielding the best accuracy for one component may be unsuitable for another. The composite criterion Ψ, combining RMSE, AIC, BIC, and χ^2, confirmed that there is no universally superior model. Third, maintaining multiple models allows for flexible adaptation to the available data, the criticality of the equipment, and the required prediction horizon. In a practical maintenance system based on a digital twin, the use of a model bank enables automatic selection of the most appropriate model type for each component and current operational condition. In summary, the proposed approach is based on the integration of four predictive models: exponential, degradation-based, Markovian, and simulation-based. Their comparative analysis made it possible to evaluate the advantages and limitations of each method. As a result, the simulation model was chosen as the core computational scheme, providing the best balance between accuracy, adaptability, and realism. This makes the proposed reliability forecasting system not only scientifically grounded but also practically applicable under real-world SPP operating conditions.

The comparative analysis of the reliability models presented above allows us to move from theoretical justification to their practical evaluation. At this stage, we consider the specific results of applying each model to the key components of the SPPs under various configurations of input parameters and operating scenarios. To this end, calculations were carried out using a unified set of metrics (RMSE, AIC, BIC, χ^2), and a quantitative assessment of the probability of failure-free operation

over 25 000 hours was performed. For each piece of SPP equipment-the generator, the main engine, the cooling system, and the electrical power unit—reliability forecasts were generated and compared with the actual failure statistics.

Table 1

Comparison of different reliability prediction models

Prediction model	RMSE	Forecast accuracy (%)
Exponential distribution	0.12	85
Degradation models	0.07	91
Markov processes	0.08	90
Simulation (Monte Carlo)	0.05	95

Table 1 provides comparative data for four reliability prediction models applied to SPP components: exponential, degradation, Markov, and simulation (Monte Carlo). The evaluation criteria are the RMSE and the forecast accuracy on a hold-out sample (as a percentage of actual observed failures). Exponential Model (constant failure intensity) yielded the poorest performance: RMSE = 0.12 and forecast accuracy 85 %. This confirms its limitation when failures result from accumulated wear or thermal degradation. It is best suited as a baseline model for rough estimates of simple, low-wear components.Degradation Model (e.g. Weibull-type with time-varying intensity $\lambda(t)=\lambda_0+\alpha t^{\beta}$) showed improved results: RMSE = 0.07 and accuracy 91 %. Its strength lies in capturing non-stationary aging processes and the effect of loads on component wear. It is especially effective for parts undergoing monotonic degradation—such as generators, heat exchangers, and bearings.

Markov Model (discrete state transitions) achieved comparable accuracy: RMSE = 0.08 and accuracy 90 %. Its advantage is formalizing the phase structure of degradation and accounting for both gradual and sudden transitions (e.g., “operational → degrading → pre-failure → failure”). It is well suited for diagnosing and forecasting complex assemblies undergoing typical wear stages. Simulation (Monte Carlo) provided the best performance: RMSE = 0.05 and accuracy 95 %. By modeling many probabilistic scenarios-including variations in load, temperature, and maintenance intervals-it captures the combined effects of multiple factors. This method is ideal for components sensitive to operating regimes and systems where failures arise from factor combinations. It also enables analysis of confidence intervals and cost-consequences of failures. In summary, each model has its own domain of applicability: exponential: for simple, stable components without evident degradation; degradation: for parts with monotonic wear and damage accumulation; Markov: for components featuring distinct degradation phases; simulation: for complex systems under variable load and environmental conditions.

Figure 3.1 presents a graph that illustrates the results of comparing various failure prediction methods based on the RMSE in reliability estimation of SPP components.

The simulation model demonstrates the highest accuracy. The degradation model outperforms both the Markov and exponential approaches, confirming the importance of accounting for cumulative wear when analyzing SPP components. To systematically compare reliability prediction methods, it is reasonable to consider three key criteria: whether the model incorporates a degradation mechanism; whether it can represent discrete transitions between component states (e.g., "operational → degraded → pre-failure → failure"); forecast accuracy, expressed through RMSE. Table 2 provides a comparative summary of these characteristics and includes practical recommendations for the application of each model depending on operating conditions and the required prediction horizon.

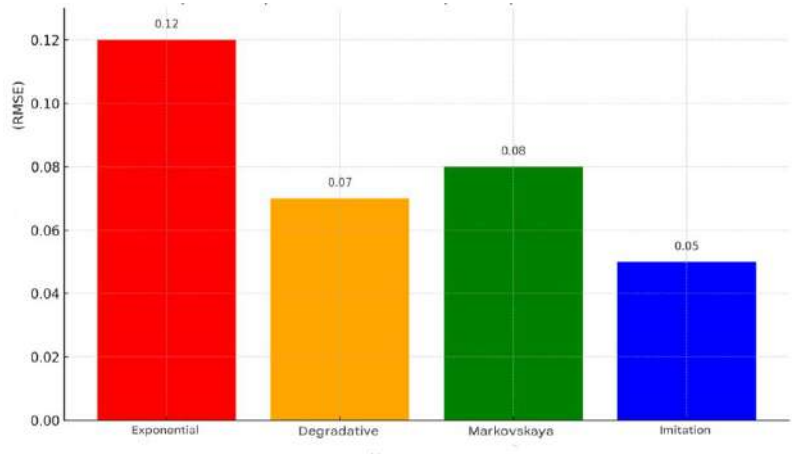


Figure 1. Comparison of reliability prediction models by root-mean-square error

Table 2

Comparative accuracy of reliability prediction models (by RMSE)

Prediction model	Accounts for degradation	Describes state transitions	RMSE	Recommended applications
Exponential	No	No	0.12	Basic estimates, preliminary assessments
Markov	Yes	Yes	0.08	Mid-term forecasting, transient state analysis
Simulation modeling	Yes	Yes	0.05	Accurate long-term assessments, complex operational scenarios
Degradation models (analytical)	Yes	No	0.09	Condition monitoring with known wear functions

The comparison clearly shows that simulation modeling offers the highest accuracy (RMSE = 0.05). This is due to its ability to capture variability in operating conditions, stochastic events, and cumulative degradation effects. As a result, this method is especially effective for long-term reliability forecasting and residual life assessment under complex operational scenarios. Markov models, while slightly less accurate (RMSE = 0.08), offer a key advantage in structural clarity. They enable formal representation of state transitions ("operational → degraded → pre-failure → failure") and are well-suited for rapid risk assessment. These models can be readily integrated into onboard predictive diagnostics systems and are applicable when moderate volumes of input data are available. Analytical degradation models provide acceptable accuracy (RMSE in the range of 0.07–0.09) and are most effective when there is a priori knowledge of wear mechanisms. Their use is particularly appropriate when combined with environmental monitoring (e.g., temperature, vibration, chemical aggressiveness), allowing for modeling the nonlinear increase in failure intensity. Despite its simplicity, the exponential model poorly reflects the behavior of most SPP components over intervals exceeding 10 000 hours, as it does not account for degradation processes. Its use is justified only for preliminary assessments or when no reliable data is available on the component's condition. Therefore, the selection of a prediction model should be based on a balance between: the availability of input data; acceptable model complexity; the required forecasting horizon. In practical operational environments, the most rational approach is a hybrid strategy, combining simulation modeling with Markov processes. This allows for simultaneously capturing probabilistic dynamics and concrete failure scenarios.

General reliability equation and parameter estimation

For all models considered, the reliability function $R(t)$ is derived from the following general equation:

$$R(t) = e^{-\int_0^t \lambda(t)dt},$$

where $R(t)$ is the probability of failure-free operation at time t;

$\lambda(\tau)$ is the time-dependent failure rate function

This integral expression accounts for the cumulative impact of component degradation over time.

With a constant failure rate $\lambda(\tau)=\lambda_0$, the model corresponds to the classical exponential distribution. For components experiencing increasing wear, a power-law dependency is used $\lambda(\tau)=\lambda_0(1+\alpha\tau^n)$. In the Markov scheme, $\lambda(t)$ is equivalent to the sum of outgoing transition rates from non-absorbing states. In the simulation model, $\lambda(t)$is calculated step-by-step for each operational scenario $X^{(j)}$. The parameters λ_0, α, n, as well as the transition rates μ, γ, are calibrated using: the OREDA field failure database; identification of CTMC parameters from operational logs; prior dependencies (Bayesian networks) in the case of limited data. The resulting reliability functions $R(t)$ are used for: estimating the remaining useful life; optimizing maintenance intervals; calculating economic losses due to downtime.

Analysis of component reliability dynamics

Reliability assessment of SPP requires not only the estimation of overall failure probability, but also an understanding of how reliability evolves over time under operational stresses.

This subsection presents a comparative analysis of the behavior of key SPP components over an extended operational period (up to 25,000 hours). Particular attention is paid to the dynamics of the reliability function, failure frequency, and the influence of load, temperature, and maintenance intervals on remaining useful life.

Figure 2 shows the reliability dynamics of the main SPP components over 25,000 hours of operation. Figure 2 shows the evolution of reliability (function R(t)) for four key components of the SPP over a 0–25,000 hour interval, calculated using Monte Carlo simulation (10,000 iterations) with variations in operational factors: relative load, cooling medium temperature, and maintenance frequency. The graph presents the probability of failure-free operation over time, obtained from the event-driven simulation model. For each component, degradation scenarios were modeled based on empirical distributions of operating parameters and failure intensity functions calibrated from historical data.

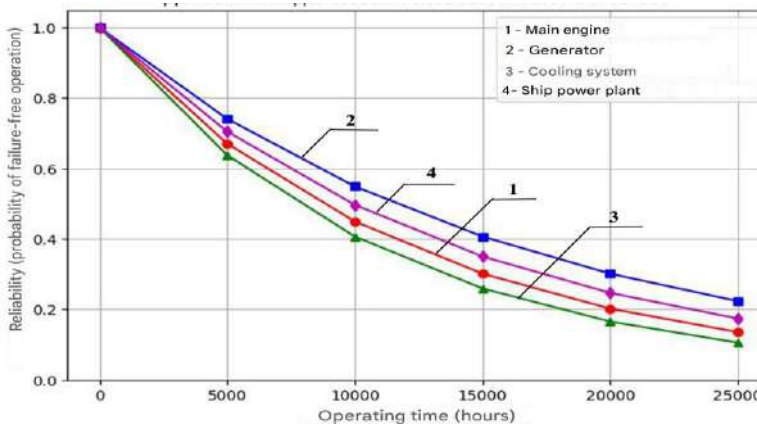


Figure 2. Reliability dynamics of key SPP components

The main engine (ME) demonstrates the steepest reliability decline: from 1.0 to approximately 0.3 by 25,000 hours, indicating the need for overhaul after 20,000 hours. The generator degrades more slowly, reaching approximately 0.45 at 25,000 hours, and its operational life can be extended with regular maintenance. The cooling system is sensitive to thermal impacts: reliability drops to about 0.55 by 15,000 hours and to ≈ 0.35 by 25,000 hours, requiring preventive actions every 10,000–12,000 hours. The ship power station shows moderate degradation: by 25,000 hours, its reliability is around 0.42 sufficient for scheduled diagnostics without urgent intervention. Thus, the primary candidates for accelerated maintenance are the main engine and cooling system. The generator and ship power

station require monitoring after 20,000 hours, when their reliability falls below 0.5. The optimal interval for preventive repair for most components is between 10,000 and 15,000 hours of operation.

To quantitatively support the graphical trends presented in Figure 2, Table 3 provides the values of the reliability function *R(t)* for key SPP components at critical stages of the operating cycle. These data are used for estimating remaining useful life and for scheduling maintenance interventions.

Table 3 presents discrete values of the probability of failure-free operation *R(t)* for the four main SPP components, calculated using Monte Carlo simulation with 10,000 runs. The model incorporated empirically determined distributions of load, temperature, and maintenance frequency, along with failure intensity functions calibrated against degradation data and state transition behavior.

These values complement the graphical interpretation in Figure 3.21 and enable precise identification of critical intervals of reliability loss. The main engine and cooling system reach $R(t) < 0.5$ between 15,000 – 20,000 hours, while the generator and ship power station remain reliable until approximately 23,000 – 24,000 hours, after which they also require major intervention.

Table 3

Long-term reliability (failure probability) of SPP components

Time (h)	Main engine	Generator	Cooling system	Shipboard power station
0	1.00	1.00	1.00	1.00
5,000	0.93	0.96	0.92	0.95
10,000	0.85	0.90	0.82	0.87
15,000	0.70	0.78	0.68	0.75
20,000	0.50	0.60	0.52	0.58
25,000	0.30	0.45	0.35	0.42

The derived data can be directly applied for residual life estimation and planning of maintenance schedules. The subsequent sections explore failure frequencies under various operating modes and the impact of maintenance intervals on component reliability.

Although the reliability function R(t) reflects the probability of failure-free operation of components over time, an important complementary metric is the normalized failure rate the expected number of failures per 1,000 operating hours depending on operating conditions. This indicator provides insight into how rapidly the risk of failure increases under varying external loads, thermal conditions, and maintenance frequencies. It is important to note that the presented values are not based on direct observations but represent average failure intensities obtained through Monte Carlo simulation, accounting for usage scenarios under three modes: nominal, high load, and emergency conditions.

Table 4 summarizes the simulated failure rates of SPP components under three operational regimes: nominal, elevated load, and emergency conditions. These

results, derived from simulation modeling, complement the previously presented *R(t)* values, offering a more detailed perspective on component sensitivity to operational factors. The resulting dependencies are used for comparative assessment of component vulnerability and to justify the need for predictive maintenance strategies when shifting toward harsher operation profiles.

Table 4

Failure frequency of components under different operating modes

Component	Nominal mode (failures/1000 h)	Increased load (failures/1000 h)	Emergency conditions (failures/1000 h)
Main engine	0.8	1.5	3.2
Generator	0.5	1.1	2.8
Cooling system	1.0	2.3	4.1
Ship power station	0.6	1.4	3.0

Based on the data presented in Table 4, the following conclusions can be drawn. The main engine shows a low failure rate under nominal conditions (0.8 failures per 1,000 operating hours), but this rate rises sharply to 3.2 failures per 1,000 hours under emergency conditions, indicating high sensitivity to increased operational loads. The generator demonstrates strong reliability in stable conditions (0.5 failures per 1,000 hours), yet under emergency scenarios the failure rate increases fivefold (2.8 failures per 1,000 hours), which emphasizes the need for enhanced monitoring. The cooling system exhibits the highest sensitivity to operating conditions. Its failure rate escalates from 1.0 to 4.1 failures per 1,000 hours under critical thermal loads and hydraulic stress, highlighting the importance of timely maintenance. The shipboard power station also experiences a decline in reliability under high loads, although its resilience remains higher compared to the main engine, making it relatively less vulnerable.

Following the assessment of overall component reliability and failure rates across different modes of operation, it becomes essential to analyze the influence of individual operational factors such as load, temperature, and maintenance intervals on the time-dependent degradation of reliability. The modeling framework includes the following variables: relative load (e.g., propeller shaft torque for the main engine, current load for the generator); temperature conditions (e.g., oil temperature for the main engine, coolant temperature for the cooling system, and winding temperature for the generator); and the maintenance schedule, which determines the accumulation of residual risk. The simulation scenarios span from nominal operational parameters to overloads and emergency conditions. Based on these inputs, reliability functions *R(t)* were computed using Monte Carlo simulation to represent the system's aggregated response to variable environmental and operational influences.

Figure 3.50 visualizes these dependencies, providing a clear picture of how the reliability of shipboard power systems changes in response to variations in

operational conditions. This analysis helps identify components that are more vulnerable under elevated loads or thermal stress and supports decisions regarding adaptive maintenance planning.

The simulation results presented in Figure 3 clearly demonstrate that the reliability of SPP components varies significantly depending on operating conditions. In nominal mode, with standard loading and normal thermal conditions, the reliability function of the components remains above 0.80 during the first 15,000 hours, which corresponds to the planned operational phase without signs of accelerated degradation. Under increased (intensive) load such as a ~20% rise in propeller resistance or generator current the rate of failure intensifies: by 15,000 hours, the probability of failure-free operation drops to 0.55–0.60, which is 30–35% lower than the baseline level. In emergency conditions a combination of overloads, cooling water overheating (above 85 °C), and infrequent maintenance leads to rapid deterioration *R(t)* falls below 0.40 as early as 12,000 –15,000 hours, and by 25,000 hours, reliability decreases to 0.20 – 0.25. This indicates critical wear and the urgent need for major overhaul. Based on the results of the simulation model, the following practical recommendations are proposed: adaptive load control: For the main engine and cooling system, it is advisable to reduce operating loads by 10 – 15% when oil or coolant temperatures increase.

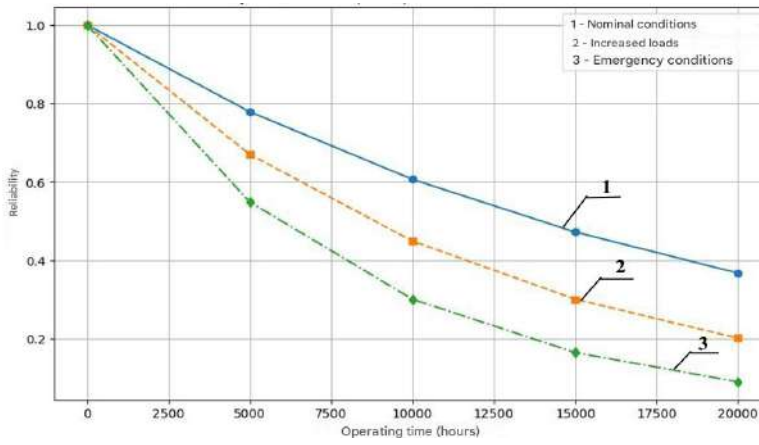


Figure 3. Influence of operational factors on the reliability of the SPP

This measure can extend the service life by 2,000 – 3,000 hours; justification of optimal maintenance intervals: Simulation of various scenarios shows that reducing the interval from 10,000 to 5,000 – 7,000 hours between scheduled maintenance procedures decreases the total failure probability by 18–22% and lowers expected

maintenance costs by more than four times; predictive replacement of critical components - for components whose residual reliability drops to R = 0.50, a condition-based replacement strategy is economically justified before actual failure occurs. This particularly applies to the main engine bearings and cooling system heat exchangers.

Thus, the chart in Figure 3 not only confirms the quantitative influence of operational factors on system reliability but also provides a foundation for justifying adaptive maintenance strategies. These strategies make it possible to maintain the reliability of the system at a safe level while minimizing total costs. For a quantitative analysis of the long-term impact of operational conditions on the reliability of SPP components, simulation scenarios were used to vary key technical environment parameters. Table 5 presents generalized calculated data for three main operational factors: vibration level (average vibration, in g); operating temperature (characteristic for each component, such as oil, windings, coolant, etc.); and relative load (percentage of rated capacity). For each component, the probability of failure-free operation R(t) after 20,000 hours of service is also provided, based on a degradation-simulation model.

These results enable a comparative analysis of the sensitivity of different SPP subsystems to operational stressors. A visual representation of how maintenance frequency affects reliability is additionally shown in Figure 3, which illustrates the role of the maintenance interval as a distinct risk factor.

Table 5

Impact of operational factors on the reliability of SPP components

Component	Vibration (average level), g	Temperature (°C)	Load (% of nominal)	Reliability after 20,0
Main engine	4.5	85	95	0.52
Generator	2.8	75	90	0.60
Cooling system	3.2	90	80	0.48
Ship power station	2.5	70	85	0.58

The analysis of the data presented in Table 5 shows that the reliability of SPP equipment after 20,000 hours of operation is determined by the combined effect of three key operational factors: vibration, temperature, and load. All values were obtained using simulation modeling based on standard operating scenarios. Vibration has a significant influence on the service life of equipment.

The main engine (ME) exhibits the highest vibration level of 4.5 g, reflecting high mechanical loading and corresponding with a relatively low reliability value (R = 0.52). The cooling system, exposed to 3.2 g vibration, shows an even lower

reliability (R = 0.48), which can be attributed to its structural susceptibility to cavitation and resonant effects. The generator (2.8 g) and ship power station (2.5 g) experience the lowest vibration levels and demonstrate the best resource preservation (R = 0.60 and R = 0.58, respectively). Temperature is the second most critical factor. The cooling system operates at 90 °C, and the main engine at 85 °C both indicating thermally stressed conditions that accelerate material degradation and aging of working media. More moderate temperatures are observed in the generator (75 °C) and power station (70 °C), correlating with their higher reliability. In this context, temperature refers to oil, coolant, cylinder gas, winding, and ambient temperatures. Their roles are further detailed in the explanation of the thermal factor below.

Load, expressed as a percentage of nominal power, also has a statistically significant impact. At 95 % load on the main engine and 90 % on the generator, reliability is notably lower than in the power station operating at 85 %. Interestingly, the cooling system despite operating at a relatively low load (80 %) exhibits the worst reliability metric. This confirms that thermal and vibrational loads are the dominant degradation drivers in its case. In summary, two component groups can be distinguished: critically vulnerable: main engine and cooling system subject to cumulative influence from all three factors, requiring early intervention and shortened maintenance intervals; relatively stable: generator and ship power station-operating under near-nominal conditions with extended resource longevity. It should be emphasized that in the development of reliability models and maintenance strategies, not only individual factors but their cumulative effects over time must be taken into account. Even if one parameter (e.g., load) remains moderate, elevated temperature or vibration alone can significantly reduce service life. This underscores the importance of multi-parameter modeling tailored to the operational specifics of each component. Temperature is among the most critical degradation factors for SPP systems, with its impact depending not only on absolute values but also on the specific application point: oil, coolant, cylinders, or windings. Table 6 summarizes the critical temperature thresholds for various media and components, indicating the levels at which accelerated reliability decline begins and the predominant types of failures observed under these conditions.

Based on Table 6, it can be concluded that each SPP component has a specific critical temperature threshold, beyond which a qualitative change occurs in failure mechanisms. For the main engine, oil overheating above 110 °C leads to reduced viscosity, impaired lubrication, and consequently, accelerated wear of friction pairs, specifically journal bearings, liners, and plain bearings. The cooling system loses reliability at temperatures exceeding 95 °C, disrupting the thermal balance of the entire installation. This promotes thermal aging and increases the likelihood of cavitation.

Table 6

Influence of temperature regimes on the reliability of SPP components

Component	Critical temperature, °C	Impact on reliability
Main engine (oil)	>110 °C	Accelerated wear of friction parts
Coolant	>95 °C	Overheating, reduced cooling efficiency
Gaseous medium in cylinders	>500 °C	Increased wear of the piston group
Generator (windings)	>120 °C	Insulation degradation, risk of breakdown

When temperatures in the engine cylinders exceed 500 °C, carbon deposit formation, thermal expansion, gas blow-by, and abrasive wear of the piston group intensify. For the generator, winding overheating above 120 °C is critical due to insulation aging, increased electrical resistance, thermal cycling, and ultimately, dielectric breakdown. These findings provide a crucial foundation for constructing temperature-dependent degradation models. Such models not only guide the development of maintenance algorithms but also define maximum permissible values in monitoring systems and help configure warning thresholds within digital twin frameworks. Based on the data from Tables 5 and.6, it is evident that vibrational stress and elevated temperature have the greatest impact on component reliability. The cooling system is particularly vulnerable to degradation at temperatures exceeding 85 °C. In contrast, the generator and ship power station demonstrate lower sensitivity to vibration.

To quantitatively assess the effect of maintenance frequency on equipment reliability, a reliability function was derived as a function of the interval between maintenance events. Modeling was carried out using an event-driven simulation approach, with intervals ranging from 2,000 to 20,000 hours. The resulting dependency is visualized in Figure 4, which illustrates the decline in failure-free probability as the interval between scheduled maintenance increases.

The graph in Figure 4 is based on the results of event-driven Monte Carlo simulation ($N = 10{,}000$) under averaged operational conditions derived from Table 3.51. It illustrates the exponential decline of failure-free probability R(t) as the maintenance interval increases: under baseline conditions of 1,000 hours, reliability remains high ($R \approx 0.95$); under the economically optimal interval of 5,000 hours, it decreases slightly ($R \approx 0.90$); but at 20,000 hours, reliability drops below 0.40 (RMSE of the forecast = 0.05). This trend is consistent with the results of Han et al. [9]; however, our study further incorporates the economic impact: total lifecycle costs increase from USD 5,300 (for 5,000-hour maintenance) to USD 26,200 (for 20,000-hour maintenance). Therefore, regular maintenance at intervals no longer than 5,000 hours offers a rational compromise between maintaining high system reliability and minimizing lifecycle costs.

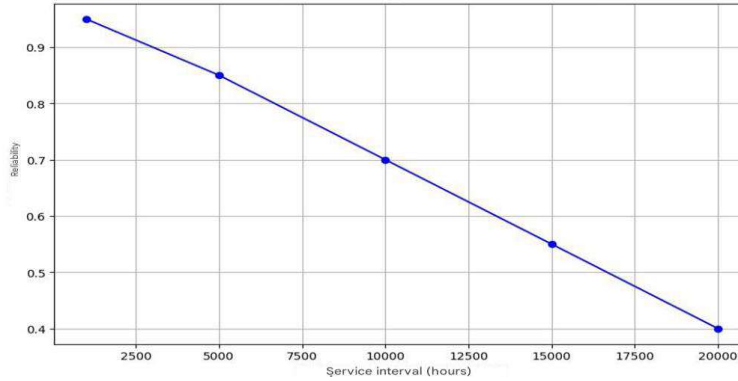


Figure 4. Impact of maintenance frequency on the reliability of SPP

Considering that vibration and thermal loads significantly accelerate degradation (as demonstrated in Table 5), the main engine and cooling system are particularly critical. For these components, it is advisable to adopt a shortened maintenance cycle after 10,000 hours of operation.

Economic assessment of maintenance strategies

Given the identified relationships between reliability and operational factors, the next step is to evaluate how preventive actions impact not only technical performance but also the overall lifecycle costs of SPP equipment. To assess the economic efficiency of different maintenance strategies, a comparative cost analysis was conducted for two operational scenarios: reactive maintenance (repair after failure only); scheduled preventive maintenance (every 5,000 hours). The main engine was selected as the case study component, being the most cost-critical in terms of failure consequences and downtime losses. The economic evaluation model is structured as follows:

$$C = C_{PM}(\Delta) + C_{rep} + C_{down}[R(t)],$$

where $C_{PM}(\Delta)$ - denotes the cost of preventive maintenance at interval Δ;

C_{rep} - direct costs of failure recovery;

C_{down} - losses associated with equipment downtime, which depend on the failure probability *R(t)* and the duration of recovery operations

In the reactive maintenance scenario, the number of failures over 25,000 hours of operation averaged 12, with total costs (repair + downtime) reaching 26.2 thousand USD. In contrast, with regular maintenance performed at 5,000-hour intervals, the number of failures was reduced to 4, and the total expenses amounted to 5.3 thousand USD. Table 7 presents a summary of the economic comparison between the two approaches. To formally compare these scenarios using key

economic metrics, Table 7 below provides a comparison of failure frequency, repair costs, downtime-related losses, and total expenditures.

Table 7

Comparative assessment of main engine maintenance costs under different strategies (25,000-hour cycle)

Operation scenario	Number of failures over 25,000 h	Repair costs C_{rep}, thousand USD	Downtime losses C_{down}, thousand USD	Total costs C, thousand USD
Without regular maintenance	12	5.8	20.4 (≈12 h downtime)	26.2
With regular maintenance (every 5,000 h)	4	1.9	3.4 (≈2 h maintenance downtime)	5.3

As shown in the table, the scheduled maintenance strategy reduces the total number of failures by a factor of three and the overall costs by more than 4.5 times. At the same time, the equipment reliability at the end of the evaluated operational interval remains above 90%, which is confirmed by the results of simulation modeling using the reliability function R(t). The graph (Fig. 4) also demonstrates that extending the maintenance interval to 20,000 hours leads to a decrease in the probability of failure-free operation to 40%.

Thus, the economic evaluation confirms the practical effectiveness of implementing preventive maintenance. The recommended interval of 5,000 hours ensures an optimal balance between operational expenditures and the reduction of failure risks. The proposed approach is scalable to other subsystems of the SPPs and can be integrated into digital twins for dynamic optimization of maintenance strategies in real time.

4 Discussion of results

The conducted study demonstrates the effectiveness of an integrative approach to reliability assessment and the development of maintenance strategies for SPP equipment. Unlike classical methods based on stationary assumptions (e.g., the exponential model with constant failure rates), the proposed methodology combines analytical degradation models, non-stationary Markov processes, and discrete-event simulation modeling. This combination enables accounting for both damage accumulation and the influence of variable operational conditions, including vibration, temperature, and load regime.

Comparison with contemporary research confirms the relevance and scientific soundness of the proposed approach. For instance, several studies (Mauro & Kana [22]; Lv & Lv [23]) consider digital twins as a promising architecture for predictive

control of maritime equipment. However, the authors emphasize the need for model unification and the integration of physically interpretable parameters. The present study meets these requirements: the reliability models are formalized, and the degradation parameters are directly dependent on operational impacts. In the works of Minchev et al. [24], digital twins are used for diagnostics and condition monitoring of marine diesel units, but the aspect of economic feasibility of maintenance is not addressed. In contrast, this study presents an evaluation of total costs under different maintenance strategies, thereby enhancing the practical significance of the results. The economic analysis conducted showed that switching from a reactive to a scheduled strategy (with a 5,000-hour interval) reduces total expenses by more than 4.5 times while increasing equipment reliability by 18–22%. The reliability optimization methods proposed by Zhou et al. [25] are based on particle swarm algorithms and are aimed at individual technological processes (e.g., cylinder block machining), without considering degradation dynamics during operation. In this study, the parametric degradation model with four technical condition states allows adaptation to changing conditions, reflecting the actual behavior of equipment over a long operational interval. The review by Liang et al. [26] highlights the lack of quantitative models capable of accounting for operational impacts such as overheating and vibration. This limitation is overcome in the present work: the developed reliability models incorporate temperature, load, and vibration parameters as arguments of the failure rate function. The introduced threshold values (e.g., 110 °C for oil, 120 °C for windings) are consistent with simulation results and may serve as the basis for automatically generating warning signals in digital twins. The work by D'Agostino et al. (2020) [27] is devoted to multiphysical modeling of ship microgrids in real time, but it lacks a reliability analysis component. The present study can be integrated into such systems, complementing them with residual life assessment and maintenance schedule optimization modules.

One of the key results of the study is the quantitative comparison of four reliability forecasting models. The simulation model demonstrated the highest accuracy (RMSE = 0.05), which is 33% better than that of the Markov model (0.08), and 58% better compared to the exponential model (0.12). The integrated metric Ψ, combining RMSE, AIC, BIC, and χ^2 with expert weights, confirmed the superiority of the hybrid model, demonstrating that accounting for the temporal dynamics of operational factors and degradation processes significantly improves the accuracy of long-term forecasting. Moreover, the study revealed differentiated sensitivity of various SPP components to operational loads. The most vulnerable components were the main engine and cooling system, for which increases in vibration and temperature lead to a sharp decline in reliability. In particular, by 20,000 hours, reliability decreases to 0.52 for the main engine and 0.48 for the cooling system. The generator and ship power station exhibit more stable behavior ($R \approx 0.60$ and 0.58, respectively), confirming the rationale for a differentiated approach to maintenance scheduling.

The practical value of the presented models lies in their ability to support informed managerial decisions regarding equipment maintenance and lifecycle management. Mathematical interpretability, integration capability within digital twins, and consistency with operational parameters make the proposed methodology promising for implementation in ship technical monitoring and decision support systems.

At the same time, the study has certain limitations. First, the input data used are averaged operational profiles that do not include streaming telemetry. Second, the model parameters were identified based on historical data and are not updated in real time. These limitations define directions for future research, including the incorporation of sensor streams, implementation of online calibration, expansion of the component base, and the use of machine learning methods for adaptive model tuning and adjustment of the integrated criterion Ψ.

Thus, the proposed integrative approach to modeling the reliability and maintenance of SPP equipment represents a balanced solution that combines mathematical rigor, engineering applicability, and economic efficiency. It may serve as a foundation for the development of intelligent prognostic systems within the framework of digitalization of marine vessel technical operations.

5 Conclusions

This study has achieved its stated objective: an integrated approach to long-term reliability analysis of SPP components has been proposed and substantiated. The approach combines physically interpretable failure rate dependencies, a non-stationary Markov framework, and simulation modeling of operational scenarios, while also linking the results to the economic efficiency of maintenance strategies.

The developed models enabled a quantitative assessment of the reliability of four key SPP subsystems over a 25,000-hour horizon. According to the simulation results, the probability of failure-free operation by the end of the cycle was approximately 30% for the main engine, 45% for the generator, 35% for the cooling system, and 42% for the ship power station. These figures highlight the need for overhaul or replacement of the most vulnerable components after 20,000 hours of operation. The root mean square error (RMSE) of failure prediction, when compared with field statistics, was 0.05 for the simulation model, 0.08 for the Markov model, and 0.12 for the exponential model—demonstrating a 33% improvement in accuracy over the nearest alternative and a 58% improvement over the baseline constant failure rate model. The integrated criterion Ψ, combining RMSE, AIC, BIC, and χ^2 with weights of 0.4:0.3:0.2:0.1, confirmed the superiority of the hybrid simulation–Markov scheme across all components.

The analysis of operational factors revealed that a 20% increase in relative load accelerates the growth of failure intensity by up to 1.7 times, and cooling water temperatures above 85 °C reduce the remaining life of the cooling system by 30%. The optimal preventive maintenance interval, determined using the residual risk function and economic criterion, was found to be 5,000 hours. With this periodicity, the total costs (maintenance + repairs + downtime) over a 25,000-hour cycle do not

exceed 5.3 thousand USD, whereas foregoing scheduled maintenance increases costs to 26.2 thousand USD more than 4.5 times higher.

The practical value of this work lies in the ability to integrate the developed mathematical module into prognostic systems of SPP digital twins, enabling shipowners to recalculate, in real time, the failure probability, remaining life, and financial consequences of a chosen maintenance strategy. Future research prospects include online calibration of model parameters based on streaming data from ship technical monitoring and control systems, expansion of the component base of the digital twin, and the use of machine learning methods for automatic tuning of weights in the Ψ criterion.

References

1. Вычужанин, В. В. (2009). Повышение эффективности эксплуатации судовой системы комфортного кондиционирования воздуха при переменных нагрузках (Монография). Одесса: ОНМУ.

2. Вычужанин, В. В. (2012). Информационное обеспечение мониторинга и диагностирования технического состояния судовых энергоустановок. Вісник Одеського національного морського університету. Збірник наукових праць, (35), 111–124.

3. Вычужанин, В. В., & Рудниченко, Н. Д. (2019). Методы информационных технологий в диагностике состояния сложных технических систем: Монография. Одесса: Экология.

4. Vychuzhanin, V., & Vychuzhanin, A. (2025). Intelligent diagnostics of ship power plants: Integration of case-based reasoning, probabilistic models, and ChatGPT. A universal approach to fault diagnosis and prognostics in complex technical systems: Monograph. Lviv–Torun: Liha-Pres. https://doi.org/10.36059/978-966-397-516-0

5. Vychuzhanin, V. V., & Rudnichenko, N. D. (2019). *Metody informatsionnykh tekhnologiy v diagnostike sostoyaniya slozhnykh tekhnicheskikh sistem: Monografiya*. Odessa: Ekologiya.

6. Zocco, F., Wang, H.-C., & Van, M. (2023). Digital twins for marine operations: A brief review on their implementation. *arXiv:2301.09574*. https://doi.org/10.48550/arXiv.2301.09574

7. Stadtmann, F., Wassertheurer, H. G., & Rasheed, A. (2023). Demonstration of a standalone, descriptive, and predictive digital twin of a floating offshore wind turbine. *Ocean Renewable Energy*. https://doi.org/10.1115/OMAE2023-103112

8. Polverino, L., Abbate, R., Manco, P., Perfetto, D., Caputo, F., Macchiaroli, R., & Caterino, M. (2023). Machine learning for prognostics and health management of industrial mechanical systems and equipment: A systematic literature review. *International Journal of Engineering Business Management, 15.* https://doi.org/10.1177/18479790231186848

9. Kaklis, D., Varlamis, I., Giannakopoulos, G., Varelas, T. J., & Spyropoulos, C. D. (2023). Enabling digital twins in the maritime sector through the

lens of AI and Industry 4.0. *International Journal of Information Management Data Insights, 3*(2). https://doi.org/10.1016/j.jjimei.2023.100178

10. Nezhada, M. M., Neshat, M., Sylaios, G., & Garcia, D. A. (2024). Marine energy digitalization: Digital twin's approaches. *Renewable and Sustainable Energy Reviews, 191,* 114065. https://doi.org/10.1016/j.rser.2023.114065

11. Mavrakos, A. S., Tsaousis, T., Faggioni, N., Caviglia, A., & Manzo, E. (2024). A digital twin approach for selection and deployment of decarbonization solutions for the maritime sector. In *State-of-the-Art Digital Twin Applications for Shipping Sector Decarbonization*. https://doi.org/10.4018/978-1-6684-9848-4.ch002

12. Liang, Q., Knutsen, K. E., Vanem, E., Æsøy, V., & Zhang, H. (2024). A review of maritime equipment prognostics health management from a classification society perspective. *Ocean Engineering, 301,* 117619. https://doi.org/10.1016/j.oceaneng.2024.117619

13. Han, P., Ellefsen, A. L., Li, G., & Holmeset, F. T. (2021). Fault detection with LSTM-based variational autoencoder for maritime components. *IEEE Sensors Journal, 21*(19). https://doi.org/10.1109/JSEN.2021.3105226

14. Xiao, B., Zhong, J., Bao, X., Chen, L., Bao, J., & Zheng, Y. (2024). Digital twin-driven prognostics and health management for industrial assets. *Scientific Reports, 14,* 13443. https://doi.org/10.1038/s41598-024-63990-0

15. Cui, Y., Sun, Y., Cui, J., Zhao, F., & Yuan, M. (2025). Digital twin for marine diesel engine: Enhancing predictive maintenance and operational efficiency. In *CSAA/IET International Conference on Aircraft Utility Systems (AUS 2024).* ResearchGate. https://www.researchgate.net/publication/388039483

16. Birolini, A. (2017). *Reliability engineering* (8th ed.). Springer. https://doi.org/10.1017/aer.2019.87

17. Tobias, P. A., & Trindade, D. C. (2011). *Applied reliability* (3rd ed.). Chapman and Hall/CRC. https://doi.org/10.1201/b11787

18. International Organization for Standardization. (2013). *ISO 12110-2:2013 Metallic materials — Fatigue testing — Variable amplitude fatigue testing — Part 2: Cycle counting and related data reduction methods.* https://cdn.standards.iteh.ai/samples/54713/496f47ac9d9e40b7af2c43f0bad42aff/ISO-12110-2-2013.pdf

19. Rausand, M., & Hoyland, A. (2004). *System reliability theory: Models, statistical methods, and applications* (2nd ed.). John Wiley & Sons. https://content.e-bookshelf.de/media/reading/L-571449-baa4bae2ce.pdf

20. Rubinstein, R. Y., & Kroese, D. P. (2017). *Simulation and the Monte Carlo method* (3rd ed.). Wiley. https://doi.org/10.1002/9781118631980

21. Burnham, K. P., & Anderson, D. R. (2002). *Model selection and inference: A practical information-theoretic approach* (2nd ed.). Springer-Verlag. http://dx.doi.org/10.1007/b97636

22. Mauro, F., & Kana, A. A. (2023). Digital twin for ship life cycle: A critical systematic review. *Ocean Engineering, 269,* 113479. https://doi.org/10.1016/j.oceaneng.2022.113479

23. Lv, Z., Lv, H., & Fridenfalk, M. (2023). Digital twins in the marine industry. *Electronics, 12,* 2025. https://doi.org/10.3390/electronics12092025

24. Minchev, D., Vasilev, K., & Petkov, P. (2023). Digital twin test bench performance for marine diesel engine applications. *Polish Maritime Research, 30*(4), 81–91. https://doi.org/10.2478/pomr-2023-0061

25. Zhou, H., Yang, W., Sun, L., Jing, X., & Li, G. (2021). Reliability optimization of marine diesel engine block machining using improved PSO method. *Scientific Reports, 11,* 21983. https://doi.org/10.1038/s41598-021-01567-x

26. Liang, Q., Knutsen, K. E., Vanem, E., Æsøy, V., & Zhang, H. (2024). A review of maritime equipment health prognostics from a classification society perspective. *Ocean Engineering, 301,* 117619. https://doi.org/10.1016/j.oceaneng.2024.117619

27. D'Agostino, F., Kaza, D., Martelli, M., Schiapparelli, G. P., Silvestro, F., & Soldano, C. (2020). Development of a multiphysics real-time simulator for model-based design of a DC shipboard microgrid. *Energies, 13*(10), 3580. https://doi.org/10.3390/en13143580

ІНТЕГРОВАНЕ МОДЕЛЮВАННЯ НАДІЙНОСТІ ТА ТЕХНІЧНОГО ОБСЛУГОВУВАННЯ ОБЛАДНАННЯ СУДНОВОЇ ЕНЕРГЕТИЧНОЇ УСТАНОВКИ З УРАХУВАННЯМ ЗНОШЕННЯ ТА ЕКСПЛУАТА

Dr.Sci. В. Вичужанін[1][0000-0002-6302-1832], **Ph.D. А. Вичужанін**[2][0000-0001-8779-2503]
Національний університет «Одеська політехніка» Україна
EMAIL: [1]v.v.vychuzhanin@op.edu.ua, [2]vychuzhanin.o.v@op.edu.ua

Анотація. *Підвищення надійності та економічної ефективності морських суден вимагає розробки інструментів прогнозування відмов, здатних враховувати реальні умови експлуатації та процеси деградації обладнання. У статті представлено інтегрований підхід до кількісної оцінки надійності ключових компонентів енергетичної установки судна (ЕУС) на інтервалі експлуатації 25 000 годин. Методологія поєднує ймовірнісні, деградаційні та імітаційні моделі з урахуванням експлуатаційних параметрів, таких як температура, відносне навантаження та інтервали технічного обслуговування. Розроблено та порівняно чотири класи моделей відмов: експоненціальну модель, модель зі змінною інтенсивністю (нестаціонарний процес Вейбулла), чотиристанову марковську схему та імітаційну модель на основі подійного Монте-Карло. Розрахунки виконано для головного двигуна, генератора, системи охолодження та суднової електростанції. Середньоквадратична похибка (RMSE) прогнозу відмов становила 0,05 для імітаційної моделі, 0,08 для марковської моделі та 0,12 для експоненціальної моделі. Інтегрований критерій якості моделі, що враховує RMSE, AIC, BIC та*

χ^2, підтвердив перевагу гібридного імітаційно-марковського підходу. Порівняльний економічний аналіз показав, що регулярне технічне обслуговування з інтервалом 5000 годин знижує загальні витрати більш ніж у 4,5 раза порівняно з реактивними стратегіями ремонту. Практична цінність методу полягає в можливості його застосування у цифрових двійниках та інтелектуальних системах підтримки прийняття рішень. Подальші розробки передбачають розширення компонентної бази моделі, інтеграцію з потоками даних у реальному часі із суднових систем моніторингу та застосування методів машинного навчання для автоматичного налаштування параметрів.

Ключові слова: *предиктивна діагностика; цифровий двійник; залишковий ресурс; інтервально-орієнтоване обслуговування; марковська модель; імітаційне прогнозування; економічна оцінка відмов.*

УДК 004.896:519.8:614.8
DOI https://doi.org/10.36059/978-966-397-538-2-19

РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ДЛЯ БАГАТОКРИТЕРІАЛЬНОГО АНАЛІЗУ НЕСТІЙКОСТІ АВТОЗАПРАВНИХ СТАНЦІЙ ДО ТИПОВИХ АВАРІЙНИХ ПОДІЙ

Ph.D. О. Іванов[1][0000-0002-8620-974X], **Ph.D. D. Jones**[2][0000-0002-9101-746X], **Dr.Sci. О. Арсірій**[1][0000-0001-8130-9613], **Ph.D. A. Labib**[3][0000-0002-5481-5833], **Ph.D. С. Смик**[1][0000-0001-7020-1826]

[1]Національний університет «Одеська політехніка», Україна,
[2]Lion Gate Building, University of Portsmouth, Portsmouth, United Kingdom,
[3]Faculty of Business and Law, University of Portsmouth, Portsmouth, United Kingdom
EMAIL: lesha.ivanoff@gmail.com, dylan.jones@port.ac.uk, e.arsiriy@gmail.com, ashraf.labib@port.ac.uk, smyk@op.edu.ua

Анотація. *Це дослідження присвячене проєктуванню та впровадженню інтелектуальної інформаційної технології (ІТ), спрямованої на багатофакторне оцінювання нестійкості автозаправних станцій (АЗС) до типових небезпечних подій. Пропонована інтелектуальна ІТ структурована у 10 дискретних послідовних етапів. Початкові етапи процесу розробки включали точне визначення домінантних типів загроз, релевантних для АЗС, та ґрунтовний аналіз нестійкості АЗС, розглянутий крізь призму потенційних негативних наслідків. З цією метою експертами в предметній області було ретельно встановлено 41 метрику (критерій). Для уточнення цього набору даних застосовується метод аналізу ієрархій для аналізу експертних думок, що дозволяє вивести редукований критеріальний простір нестійкості з виключенням метрик, пов'язаних із летальними випадками. На основі цього*

уточненого набору критеріїв було систематично зібрано відповідні експлуатаційні дані, що стосуються мережі АЗС, та надалі запропоновано деталізовану процедуру їхньої попередньої обробки й кодування. Фактичне оцінювання нестійкості кожної АЗС проводиться з використанням розробленої узагальненої моделі. Для полегшення розуміння та інтерпретації результатів оцінювання запропоновано застосування теорії нечітких множин, зокрема використання трапецієподібних функцій належності. Ця інтелектуальна ІТ для оцінювання нестійкості АЗС зрештою реалізована у формі системи підтримки прийняття рішень (СППР). Кінцевий користувач (особа, що приймає рішення), після введення об'єктно-специфічних даних відповідно до критеріїв нестійкості конкретної АЗС, отримує попередній показник нестійкості разом із чітким поясненням методології його виведення. Ключовим елементом СППР є ретельно сформована база знань, що містить попередньо оцінені дані АЗС, яка має функціонал для динамічного редагування та розширення. Впровадження запропонованої інтелектуальної ІТ багатофакторного оцінювання нестійкості АЗС та побудованої на її основі СППР забезпечує надійну можливість для прийняття науково обґрунтованих рішень щодо оцінки потенційно вразливих об'єктів та ефективного запобігання ризикам (або управління ризиками).

Ключові слова. *Автозаправна станція, інтелектуальна інформаційна технологія, система підтримки прийняття рішень, багатофакторний аналіз рішень, небезпечні події, нечітка логіка, метод аналізу ієрархій.*

1. Вступ і постановка проблеми

Ця наукова праця розглядає забезпечення екологічної безпеки територій від потенційних аварій на потенційно небезпечних об'єктах (таких як великі промислові підприємства чи склади зберігання небезпечних речовин, магістральні трубопроводи, посудини під тиском тощо) як одну із суспільно важливих функцій держави. Цей процес набуває особливого значення в умовах військових дій у країні, коли такі об'єкти можуть стати мішенню, спричиняючи техногенну катастрофу, порівнянну за масштабом із невеликим землетрусом чи цунамі (наприклад, руйнування Каховської ГЕС в Україні у 2023 році).

Під час першого етапу спільного українсько-британського дослідницького проєкту *UUT14* «Багатокритеріальна методологія на основі нечіткої логіки для картографування ризиків автозаправних станцій та подальшої оптимізації рішень» між Національним університетом «Одеська політехніка» та Університетом Портсмута (Велика Британія) у рамках ініціативи *UK-Ukraine Twinning Initiative* [1], автозаправні станції (АЗС) були обрані як приклад складної технологічної системи, що може зазнати негативного впливу. Було сформовано чотири групи критеріїв для оцінки нестійкості АЗС до основних типів аварій, загалом 41 критерій. З метою редукції первинного простору критеріїв нестійкості АЗС було застосовано метод багатокритеріального аналізу рішень, а саме метод аналізу ієрархій (MAI), із використанням веб-

опитування експертів. Також була залучена інформаційна технологія (ІТ), що дозволила отримати оцінки важливості критеріїв у кожній групі [2].

На другому етапі проєкту, після обґрунтованого скорочення первинного критеріального простору, запропоновано розробити ІТ, яка дасть змогу, спираючись на зібрані дані щодо мережі АЗС у місті за всіма критеріями, здійснити попереднє оцінювання нестійкості АЗС, надаючи рекомендації для оцінюваних об'єктів із застосуванням методів нечіткої логіки (НЛ), що й розглядається в цій публікації. Початкова оцінка нестійкості АЗС за допомогою ІТ є невід'ємною складовою оцінювання та управління ризиками, що дозволяє державним установам приймати зважені рішення та вживати запобіжних заходів для мінімізації масштабів негативного впливу можливої аварії, сприяючи таким чином покращенню соціально-економічного благополуччя країни та екологічної безпеки територій.

2. Огляд літературних джерел

МАІ є одним із найбільш популярних методів для структурування та аналізу складних процесів прийняття рішень. Його розробив у 1970-х роках Томас Сааті [3]. Розглянемо приклади застосування *МАІ стосовно автозаправних станцій* у нещодавніх наукових публікаціях. У дослідженні [4] МАІ було використано для оцінки преференцій зацікавлених сторін щодо критеріїв вибору місцеположення АЗС. Метою проєкту є встановлення оптимального місця розташування АЗС шляхом інтеграції уподобань різних суб'єктів та аналізу сейсмічної зони. П'ятірка найважливіших критеріїв включає: належне землекористування (вплив 31,2 %), наявність служб оперативного реагування (23,6 %), захист навколишньої території від пожежі та вибуху (18,3 %), захист системи водопостачання від витоків із підземних резервуарів (14,2 %) та легкість доступу до об'єкта для проведення будь-яких супутніх робіт (12,7 %). У роботі [5] автори запропонували комплексний підхід, що поєднує МАІ та багатокритеріальне цільове програмування для ефективного вибору локацій АЗС. На першому рівні структури МАІ метою є вибір місця розташування АЗС. Другий рівень охоплює критерії інтенсивності транспортного потоку, навколишнього середовища та будівельних характеристик. Третій рівень містить підкритерії, як-от: середня кількість автомобілів, мотоциклів та час очікування для критерію трафіку; середнє споживання палива, кількість конкурентів, середня швидкість оплати та сприйняття місцевими мешканцями для критерію навколишнього середовища; а також кількість заправних острівців, кількість колонок та розмір ділянки для критерію будівельних характеристик. Для подальшого вдосконалення методу рекомендовано спростити процес та забезпечити зручні інструменти для осіб, що приймають рішення. Для аналізу ризиків, з якими стикаються АЗС у Пакистані, та визначення їхніх пріоритетів у дослідженні [6] використовувалися МАІ та інтервальний попарний аналіз. Результати засвідчили, що п'ять факторів ризику, які мають найбільший вплив на діяльність АЗС, є такими: транспортування та розвантаження цистерн; розподіл палива; зберігання

палива на об'єкті; ремонт, технічне обслуговування та модифікація; а також інші фактори ризику. Попри комплексний характер дослідження, воно має два основні обмеження. По-перше, зібрані дані походять лише з трьох округів провінції Пенджаб у Пакистані, що може обмежувати узагальнення результатів на ширшу територію. По-друге, використані у цьому дослідженні статистичні методи, як-от нечіткий ієрархічний аналіз та інтервальний попарний порівняльний аналіз, також мають власні ліміти та недоліки. Крім того, багатокритеріальний MAI застосовується для оцінки ризиків будівництва газокомпресорних станцій та пріоритизації небезпечних чинників у роботі [7]. Зважаючи на те, що будівельні майданчики є одним із найпоширеніших місць виникнення інцидентів, проведення оцінки ризику безпеки проєкту є важливою складовою ефективного управління будівельними проєктами. Цей проєкт більшою мірою пов'язаний із безпекою під час спорудження АЗС, ніж з оцінкою їхньої нестійкості, та включає 23 види діяльності з відповідними ризиками, що з них випливають.

Методи НЛ для оцінювання АЗС із різних аспектів активно застосовуються у сучасних дослідженнях. У [8] автори розглядають нову методологію на основі НЛ для визначення оптимального вибору АЗС. Ця праця спрямована на вибір найбільш прийнятної АЗС під час пандемії *COVID-19* відповідно до визначених критеріїв. Слід зазначити, що дослідження, проведене авторами, виходить за межі традиційних підходів завдяки використанню нового методу *AHP-VIKOR*, що ґрунтується на НЛ. Важливим обмеженням була складність збору експертних висновків під час пандемії *COVID-19*, що могло вплинути на ефективність отримання даних та досягнення консенсусу. Наприклад, пандемія ускладнила отримання експертних оцінок щодо деяких аспектів вибору місця розташування АЗС. Крім того, не було враховано важливість місця проживання споживачів, що може бути суттєвим фактором при визначенні найбільш відповідної АЗС. У статті [9] розглядається просторове розміщення АЗС з використанням нечіткої моделі в рамках географічної інформаційної системи (ГІС) на конкретному прикладі. Це дослідження представляє модель для осіб, що приймають рішення, для встановлення оптимального місця розташування АЗС за допомогою комбінації нечіткої логіки та ГІС. У роботі [10] була розроблена методика, що базується на контролері нечіткої логіки з використанням *Google Maps*, для знаходження найближчої АЗС. Для нечіткого моделювання було взято п'ять лінгвістичних вхідних змінних, включно з відстанню, наявністю палива, показником завантаженості доріг, кількістю палива та кількістю світлофорів, щоб отримати один вихідний показник – час, необхідний для досягнення АЗС. Система дозволяє водію визначати місцезнаходження АЗС із більшою точністю, що вимагає менших часових витрат.

Комбіноване застосування багатокритеріального аналізу рішень та ГІС також використовується *для комплексного оцінювання АЗС*, зокрема у таких дослідженнях. Інтегроване використання оцінки впливу на навколишнє середовище та MAI з подальшою візуалізацією даних у ГІС для визначення

придатності земельної ділянки під АЗС обговорюється в [11]. Дослідження зосереджується виключно на аналізі місцеположення АЗС заради максимального збереження природного довкілля. Іншими словами, дослідження не розглядає питання з різних точок зору і не визначає «стійкість» АЗС як таку.

У дослідженні [12] вивчається застосування МАІ та ГІС (*ArcGIS)* для оцінки відповідності місць розташування АЗС. У цій роботі для збору первинних даних використовується ручний *GPS*-навігатор. Вторинні дані включають топографічні та ґрунтові карти, з яких були вилучені типи ґрунтів, дорожня мережа, водні об'єкти, особливості схилу рельєфу та землекористування на території.

Стаття [13] проводить оцінку та покращення просторового розподілу за допомогою ГІС для запобігання екологічним ризикам та забезпечення безпеки АЗС. Дослідження має на меті підвищення безпеки шляхом оптимізації розташування АЗС та зниження потенційних ризиків. Використовуючи методи багатокритеріального аналізу рішень та ГІС, були досліджені ризики АЗС і запропоновано краще місце розташування для них. У роботі [14] аналізується локалізація роздрібних АЗС на основі ГІС. Атрибутивна інформація, така як вік АЗС, кількість колонок, реалізовані нафтопродукти, функціональний стан кожної АЗС та відстань АЗС до інших об'єктів інфраструктури, була оцінена шляхом польового обстеження та розповсюдження анкет серед власників і співробітників кожної станції.

3. Постановка мети і завдань дослідження

Мета дослідження, що розглядається у цій науковій публікації, полягає у розробці інтелектуальної ІТ для багатокритеріального оцінювання нестійкості автозаправних станцій до виникнення основних типів аварій, впровадження якої забезпечить можливість більш оперативного попереднього обстеження техногенної безпеки АЗС.

Для досягнення поставленої мети необхідно виконати такі основні завдання:

- проєктування ключових етапів інтелектуальної ІТ для багатокритеріального оцінювання нестійкості АЗС щодо основних типів аварій, спираючись на попередні результати досліджень [2];
- детальна розробка основних кроків запропонованої інтелектуальної ІТ, зокрема: побудова схеми кодування даних про АЗС; визначення вагових коефіцієнтів критеріїв нестійкості АЗС; створення узагальненої моделі для отримання числової оцінки нестійкості АЗС, а також її інтерпретація на основі методів НЛ;
- впровадження запропонованої інтелектуальної ІТ у вигляді системи підтримки прийняття рішень (СППР) та її апробація на базі даних мережі АЗС певного населеного пункту.

У дослідженні застосовуються такі методичні підходи: метод багатокритеріального аналізу рішень для оцінки нестійкості АЗС (МАІ),

методика інтелектуальної обробки та кодування вхідних даних за критеріями нестійкості АЗС, метод нечіткої логіки для категоріального оцінювання нестійкості АЗС, а також метод градієнтної шкали та координатної нормалізації для візуалізації результатів.

4. Інтелектуальна інформаційна технологія багатокритеріального оцінювання нестійкості АЗС до основних типів аварій

Розробка інтелектуальної ІТ, що дозволить визначати нестійкість АЗС до основних типів аварій [2], вимагає залучення фахівців у галузі екологічної безпеки та розробників у сфері *Data Science*. Для створення інтелектуальної ІТ, що буде реалізована як СППР для оцінки нестійкості АЗС до основних типів аварій, може бути запропонований наступний перелік із 10 послідовних етапів із зазначенням суб'єктів, що виконують відповідний етап (у дужках):

1) ідентифікація головних видів аварій (Експерт (Е));
2) розробка первинного простору економічних, екологічних та соціальних критеріїв (ЕЕСК) для оцінювання нестійкості АЗС до основних типів аварій (Е, фахівець із *Data Science* (ФД));
3) отримання вектора вагових коефіцієнтів ЕЕСК на основі МАІ (ФД);
4) формування вторинного простору ЕЕСК для оцінювання нестійкості АЗС (ФД);
5) отримання оцінок значень ЕЕСК вторинного простору для кожної АЗС на основі публічних даних (Е, ФД);
6) попередня обробка та кодування отриманих публічних даних у вторинному просторі (ФД);
7) отримання вектора вагових коефіцієнтів ЕЕСК на основі МАІ у вторинному просторі (ФД);
8) удосконалення узагальненої моделі для оцінювання нестійкості АЗС із урахуванням етапів 1–7 (ФД);
9) визначення та інтерпретація оцінки нестійкості АЗС із застосуванням НЛ (ФД);
10) побудова бази знань для оцінювання нестійкості АЗС до основних типів аварій (ФД).

Зважаючи на вищезазначені етапи, на рис. 1 представлено схему розробленої інтелектуальної ІТ оцінки нестійкості АЗС до основних типів аварій у вигляді СППР.

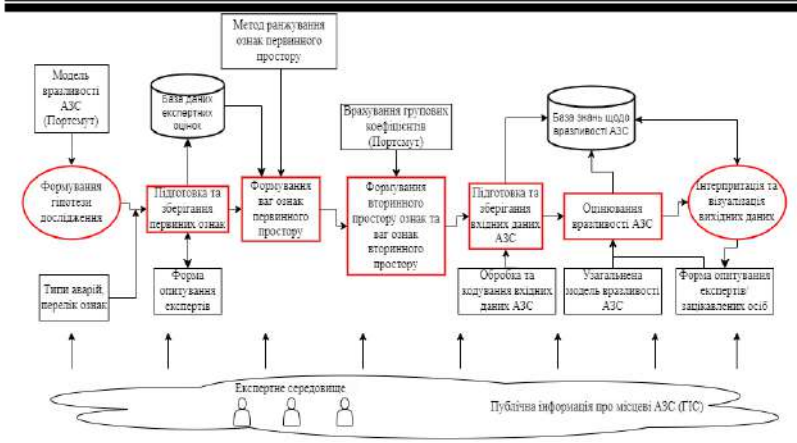


Рисунок 1. Концептуальна схема інтелектуальної ІТ оцінки нестійкості АЗС до основних типів аварій у форматі СППР [17]

5. Розширений огляд головних етапів запропонованої ІТ

На *першому етапі* вищеописаного переліку, під час розробки інтелектуальної ІТ для оцінки нестійкості АЗС, у [2] були встановлені основні типи аварій: вибух пароповітряної суміші нафтопродуктів із утворенням ударної хвилі, пожежа розливу нафтопродуктів та «вогняна куля».

На *другому етапі*, для розробки первинного простору критеріїв оцінювання нестійкості АЗС, було запропоновано обрати 3 групи критеріїв стосовно можливих наслідків аварії: економічні, соціальні та екологічні, а також окрему групу «втрачені життя», яку не слід безпосередньо порівнювати з іншими критеріями. Ці групи критеріїв формують первинний простір критеріїв Sp_{pf}, детально описаний у [2]:

$$Sp_{pf} = \langle LL, E, S, A \rangle, \quad (1)$$

де втрачені життя $LL = \{LL_1, LL_2, LL_3, LL_4, LL_5, LL_6, LL_7, LL_8, LL_9\}$;

економічні $E = \{E_1, E_2, E_3, E_4, E_5, E_6, E_7, E_8, E_9, E_{10}\}$;

соціальні $S = \{S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}\}$;

екологічні $A = \{A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10}, A_{11}\}$.

На *третьому етапі* було здійснено оцінювання критеріїв за допомогою MAI, реалізованого через веб-опитування експертів, а також розробленого алгоритму, що трансформував експертні оцінки у матрицю попарних порівнянь для подальшого виконання оцінки критеріїв у кожній групі відповідно до процедури MAI [2].

На *четвертому етапі* сформовано вторинний простір критеріїв шляхом редукції первинного простору. Відбір критеріїв, які переходять із первинного простору до вторинного для кожного типу аварії *j*, здійснювався з використанням порогової обробки відповідних значень вагових коефіцієнтів критеріїв w_{ji} у відповідній групі:

$$w_{eji}^{'} = \left\{ \forall w_{eij} \geq \lambda_E \right\}; i = \overline{1, N_E}; \; j = \overline{1,3}$$

$$w_{sji}^{'} = \left\{ \forall w_{sij} \geq \lambda_S \right\}; i = \overline{1, N_S}; \; j = \overline{1,3} \qquad (2)$$

$$w_{aji}^{'} = \left\{ \forall w_{aij} \geq \lambda_A \right\}; i = \overline{1, N_A}; \; j = \overline{1,3}$$

Як порогові значення автори прийняли $\lambda_E = \lambda_S = \lambda_A = 0{,}1$ для кожної групи критеріїв *k*. При цьому група критеріїв, що визначає можливі втрачені життя внаслідок аварії, не підлягає редукції згідно з принципами проєкту [2].

Тоді вторинний простір критеріїв оцінювання нестійкості АЗС матиме вигляд:

$$Sp_{sf} = \left\langle \left\{ \left\langle LL_{ji} \right\rangle \right\}, \left\{ \left\langle w_{eji}^{'}; E_{ji}^{*} \right\rangle \right\}, \left\{ \left\langle w_{sji}^{'}; S_{ji}^{*} \right\rangle \right\}, \left\{ \left\langle w_{aji}^{'}; A_{ji}^{*} \right\rangle \right\} \right\rangle, \qquad (3)$$

де $w_{eji}^{'}$, $w_{sji}^{'}$, $w_{aji}^{'}$ – це вагові коефіцієнти відповідних критеріїв в економічній E_{ji}^{*}, соціальній S_{ji}^{*} та екологічній групі A_{ji}^{*} відповідно.

Відповідно, групи критеріїв у вторинному просторі після завершення четвертого етапу методу набувають такої форми:

- втрачені життя $LL = \left\{ LL_1, LL_2, LL_3, LL_4, LL_5, LL_6, LL_7, LL_8, LL_9 \right\}$;
- економічні $E^{*} = \left\{ E_1^{*}, E_2^{*}, E_3^{*}, E_4^{*}, E_5^{*}, E_6^{*} \right\}$;
- соціальні $S^{*} = \left\{ S_1^{*}, S_2^{*}, S_3^{*}, S_4^{*}, S_5^{*} \right\}$;
- екологічні $A^{*} = \left\{ A_1^{*}, A_2^{*}, A_3^{*}, A_4^{*}, A_5^{*}, A_6^{*} \right\}$.

На *п'ятому етапі* автори здійснили збір даних на основі сформованого вторинного простору критеріїв для мережі АЗС певного населеного пункту з публічних джерел. Фрагменти отриманих «сірих» даних частково представлені в табл. 1, і з міркувань безпеки координатне прив'язування до конкретної АЗС не здійснюється. На цьому етапі дані щодо критеріїв втрачених життів не збиралися.

Таблиця 1

Фрагмент значень «сірих» даних про АЗС, отриманих із публічних джерел [17]

АЗС номер		4	5	6	7
E_2^*	число паливо-роздавальних колонок	1	6	4	2
E_3^*	додаткові підприємства	немає	магазин	немає	немає
S_2^*	тип зони	промислова	промислова, рекреаційна	промислова, рекреаційна	промислова
A_1^*	тип палива, стандарт	газ	5 + газ	3	4
A_3^*	відстань до лісової зони	300 м	25 м	10 м	немає

При отриманні таких «сірих» даних виникає низка проблем, що ускладнюють їх збір, обробку та аналіз для подальшого оцінювання нестійкості АЗС:

- різноманітність необхідних даних: дані, потрібні для оцінки значень ЕЕСК, є дуже різнорідними, що унеможливлює знаходження всіх відомостей в одному джерелі; деякі дані вимагають знання особливостей АЗС, кваліфікації персоналу тощо;
- необхідність ручного збору даних: переважно потрібен пошук та аналіз даних зацікавленою особою, що може призвести до пошуку великого обсягу, іноді надлишкової інформації, та спотворення даних;
- неструктурованість даних: відомості, отримані під час пошуку та аналізу, здебільшого є дуже неструктурованими та деталізованими, що ускладнює їх аналіз та обробку.

На *шостому етапі* отримані «сірі» дані про АЗС, фрагменти яких були показані в табл. 1, обробляються та кодуються відповідно до потреб побудови подальшої СППР для оцінки нестійкості АЗС до основних типів аварій. Згідно з визначеними значеннями критерію, даним надається кодове значення в діапазоні від 0 до 1, де 0 відповідає найменш несприятливому значенню критерію, а 1 – найбільш небезпечному. Фрагмент таблиці кодування даних для деяких критеріїв представлений у табл. 2, повну таблицю кодування для всіх критеріїв вторинного простору, окрім втрачених життів, можна знайти в [15].

Таблиця 2

Фрагмент схеми кодування «сірої» бази даних про АЗС [15]

Критерій	E_1^* – тип конструкції АЗС			
Значення	з підземними резервуарами		з наземними резервуарами	
Код	0.5		1.0	
Критерій	S_1^* – густина населення в районі поблизу АЗС			
Значення	висока	середня	низька	дуже низька
Код	1.0	0.6	0.2	0
Критерій	A_1^* – тип палива, що використовується, стандарт			
Значення	кількість типів палива $GS_{fueltypes}$ + газ ($GS_{gasfuel}$ наявний чи ні – 0 або 1)			
Код	$a_{f1} = \begin{cases} \frac{GS_{fueltypes}}{6}, GS_{fueltypes} \le 6 \\ 0.7, GS_{fueltypes} > 6 \end{cases} + 0.3 \cdot GS_{gasfuel}$			

На *сьомому етапі*, для отримання вектора вагових коефіцієнтів відповідних груп ЕЕСК за допомогою MAI, автори виходили з таких міркувань. Оскільки вторинний критеріальний простір містив 17 компонентів, для спрощення реалізації MAI були використані дані, отримані британськими колегами в рамках проєкту, а саме: за допомогою класичного MAI вони порівняли відповідні групи критеріїв між собою, таким чином отримавши вагові коефіцієнти для групи [16]. Результати цього порівняння та відповідні вагові коефіцієнти наведено в табл. 3. Перерахунок вагових коефіцієнтів критеріїв, отриманих на четвертому етапі, відбувається шляхом множення ваги відповідного критерію на значення відповідного вагового коефіцієнта групи (табл. 3), що відображено як *InitialWeight* у наступних формулах:

$$w_{eji}^{"} = w_{eij}^{'} \cdot EconomicInitialWeight \; ; \; w_{sji}^{"} = w_{sij}^{'} \cdot SocialInitialWeight \; ;$$
$$w_{aji}^{"} = w_{aij}^{'} \cdot EnvironmentalInitialWeight \quad (4)$$

.

Таблиця 3

Результати попарного порівняння груп критеріїв нестійкості АЗС згідно з MAI[16]

Матриця попарного порівняння груп критеріїв			
	Економічні	Екологічні	Соціальні
Економічні	–	1,5	3
Екологічні	0,667	–	1,667
Соціальні	0,333	0,6	–
Значення вагових коефіцієнтів груп критеріїв згідно з MAI, що були отримані з матриці попарних порівнянь			
Вимір нестійкості		Початкова вага	
Економічні		0,505	
Екологічні		0,317	
Соціальні		0,179	

Далі ми підсумовуємо отримані вагові коефіцієнти для кожного типу аварії *j* щоб отримати значення W_j для подальшої нормалізації ваг критеріїв, яка дозволить отримати вектор вагових коефіцієнтів для кожного типу аварії із загальною сумою 1:

$$W_j = \sum_{i=1}^{N_E^*} w_{eji}^{"} + \sum_{i=1}^{N_S^*} w_{sji}^{"} + \sum_{i=1}^{N_A^*} w_{aji}^{"}, j = \overline{1,3} \quad ; \tag{5}$$

$$w_{eij}^{*} = \frac{w_{eji}^{"}}{W_j}, w_{sij}^{*} = \frac{w_{sji}^{"}}{W_j}, w_{aij}^{*} = \frac{w_{aji}^{"}}{W_j} \quad . \tag{6}$$

Значення векторів вагових коефіцієнтів критеріїв, отримані на сьомому етапі для кожного типу аварії, наведені в табл. 4.

На *восьмому етапі*, з урахуванням попередніх етапів 1–7, ми пропонуємо наступну узагальнену модель для оцінювання нестійкості АЗС:

$$V_f = \frac{\sum_{j=1}^{N_{ac}} \left(\sum_{i=1}^{N_E^*} w_{eji}^{*} \cdot e_{fi} + \sum_{i=1}^{N_S^*} w_{sji}^{*} \cdot s_{fi} + \sum_{i=1}^{N_A^*} w_{aji}^{*} \cdot a_{fi} \right)}{N_{ac}} \quad , \tag{7}$$

де V_f – оцінка нестійкості АЗС з ідентифікатором *f*;

N_{ac} – загальна кількість основних типів аварій (у дослідженні *j* = 3);

N_E^*, N_S^*, N_A^* – загальна кількість ЕЕСК відповідно;

w_{eji}^{*}, w_{sji}^{*}, w_{aji}^{*} – вага ЕЕСК i для j -го сценарію аварії відповідно (табл. 4);

e_{fi}, s_{fi}, a_{fi} – оцінка *i-го* ЕЕСК для *f-тої* АЗС за шкалою від 0 до 1 відповідно (табл. 2 [15]).

Таблиця 4

Вектори вагових коефіцієнтів вторинного простору ознак [17]

Економічні критерії	w^*_{eij}	Соціальні критерії	w^*_{sij}	Екологічні критерії	w^*_{aij}
Вибух з утворенням ударної хвилі					
E_1^*	0,073	S_1^*	0,056	A_1^*	0,061
E_2^*	0,122	S_2^*	0,024	A_2^*	0
E_3^*	0,069	S_3^*	0,048	A_3^*	0,058
E_4^*	0,069	S_4^*	0,029	A_4^*	0
E_5^*	0,116	S_5^*	0,027	A_5^*	0,07
E_6^*	0,106	–	–	A_6^*	0,073
Пожежа розливу нафтопродуктів					
E_1^*	0,098	S_1^*	0,042	A_1^*	0,058
E_2^*	0,125	S_2^*	0,038	A_2^*	0,058
E_3^*	0	S_3^*	0,052	A_3^*	0,094
E_4^*	0	S_4^*	0,038	A_4^*	0,055
E_5^*	0,138	S_5^*	0,042	A_5^*	0,058
E_6^*	0,103	–	–	A_6^*	0
«Вогняна куля»					
E_1^*	0,082	S_1^*	0,053	A_1^*	0,062
E_2^*	0,153	S_2^*	0,03	A_2^*	0,056
E_3^*	0	S_3^*	0,055	A_3^*	0,092
E_4^*	0,074	S_4^*	0,032	A_4^*	0
E_5^*	0,109	S_5^*	0,032	A_5^*	0
E_6^*	0,105	–	–	A_6^*	0,065

Оскільки значення оцінки нестійкості АЗС, отримане за формулою (7), виражається десятковим числом у діапазоні [0; 1], необхідно інтерпретувати отримане значення V_f у формі, зрозумілій користувачеві (особі, що приймає рішення). Враховуючи, що чітке визначення приналежності АЗС до однієї з категорій оцінки нестійкості є нетривіальним завданням, ми можемо використати НЛ та визначити цю приналежність за допомогою лінгвістичної змінної.

На *дев'ятому етапі* оцінка нестійкості АЗС визначається як лінгвістична змінна GV, що являє собою кортеж:

$$\langle GV, T, V \rangle, \tag{8}$$

де: GV – назва лінгвістичної змінної;

T = {«низька нестійкість», «середня нестійкість», «висока нестійкість»} – базова терм-множина лінгвістичної змінної, кожне значення якої представляє

нечітку змінну α_i . У якості першого наближення при виборі базової терм-множини пропонується використовувати три терми, хоча надалі ця кількість нечітких змінних може бути збільшена;

V = [0; 1] – область визначення нечітких змінних, що входять до визначення лінгвістичної змінної.

Нечіткі змінні з терм-множини T також визначаються як кортежі вигляду:

$$T = \langle \alpha_i, V_i, A_i \rangle, i = \overline{1,3}, \quad (9)$$

де: α_i – назва нечіткої змінної (α_1 – «низька нестійкість», α_2 – «середня нестійкість»; α_3 – «висока нестійкість» відповідно);

V_i – область визначення, для всіх 3 змінних V = [0; 1];

$A_i = \{V_f, \mu_{A_i}(V_f)\}$ – нечітка множина на V, що описує можливі значення, які може нечітка змінна α_i набувати ($\mu_{A_i}(V_f)$ – ступінь приналежності значення V_f до заданої нечіткої змінної α_i у діапазоні [0; 1]).

Нечіткі множини A_1, A_2 і A_3 запропоновано подати у вигляді трапецієподібних функцій належності f_{T_i} як перше наближення через їхню простоту, частоту використання в наукових дослідженнях, а також через відсутність досліджень щодо розподілу (гаусового та інших) оцінок АЗС через малий обсяг досліджуваної сукупності об'єктів. Визначимо трапецієподібні функції належності f_{T_i} для нечітких змінних α_i із такими параметрами:

- $f_{T_1}(V_f; 0; 0; 0,3; 0,4)$ – для α_1 «низька нестійкість»;
- $f_{T_2}(V_f; 0,3; 0,4; 0,6; 0,7)$ – для α_2 «середня нестійкість»;
- $f_{T_3}(V_f; 0,6; 0,7; 1; 1)$ – для α_3 «висока нестійкість».

Відповідні функції належності f_{T_i} можуть бути графічно відображені на рис. 2. Нечітка змінна «низька нестійкість» показана чорним кольором, «середня нестійкість» – синім, а «висока нестійкість» – зеленим.

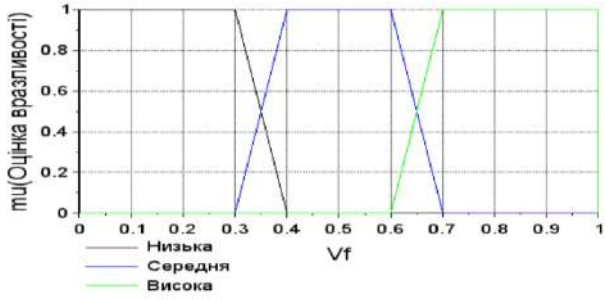


Рисунок 2. Графік трапецієподібних функцій належності нечітких змінних оцінки нестійкості АЗС [17]

Після підстановки значення V_f можна обчислити ступінь приналежності оцінки нестійкості до однієї з трьох нечітких змінних (відповідно $\mu_{A_i}(V_f)$). Перехідні значення від «низької» до «середньої» нестійкості в діапазоні [0,3; 0,4] та від «середньої» до «високої» нестійкості в діапазоні [0,6; 0,7] встановлюють ступінь нечіткості у визначенні приналежності оцінки АЗС до відповідних змінних.

Відповідно до вищезазначених міркувань, на *десятому етапі* формується база знань для оцінювання нестійкості АЗС до виникнення аварій основних типів, яка, своєю чергою, складається із заповненої бази даних про мережу АЗС, що надалі поповнюватиметься відповідно до етапів 5–6, а також міститиме базу правил для виведення інтерпретованого результату оцінки нестійкості АЗС. Згідно з вищезазначеним, можемо запропонувати таку базу правил для виведення інтерпретованого результату (з урахуванням кількості запропонованих термів, вона може бути розширена в майбутньому):

- П$_1$: якщо $f_{T_i} > 0$ – вивести повідомлення: «Оцінювана АЗС зі ступенем f_{T_i} % належить до категорії «назва *i-ї* нечіткої змінної»;
- П$_2$: if $f_{T_i} = 0$ – не виводити повідомлення;
- П$_3$: if $N(f_{T_i} > 0) > 1$ – вивести повідомлення: «Оцінювана АЗС зі ступенем $f_{T_{i1}}$ % належить до категорії «назва першої *i-ї* нечіткої змінної» і зі ступенем $f_{T_{i2}}$ % належить до категорії «назва другої *i-ї* нечіткої змінної».

Після виведення інтерпретованого результату оцінки нестійкості АЗС до основних типів аварій можуть бути сформовані такі повідомлення, що характеризують нечіткі змінні нестійкості АЗС і можуть слугувати підґрунтям для прийняття рішень зацікавленими сторонами:

- «низька нестійкість» означає, що у випадку ймовірної аварії на оцінюваній АЗС масштаб несприятливих наслідків (економічних, соціальних та екологічних) може бути низьким і відповідати певному прийнятному рівню;
- «середня нестійкість» означає, що у випадку ймовірної аварії на оцінюваній АЗС масштаб несприятливих наслідків (економічних, соціальних та екологічних) може бути вищим за середній рівень і може слугувати підставою для проведення обстеження оцінюваної АЗС контролюючими органами на предмет дотримання всіх необхідних нормативів та вжиття необхідних заходів;
- «висока нестійкість» означає, що у випадку ймовірної аварії на оцінюваній АЗС масштаб несприятливих наслідків (економічних, соціальних та екологічних) може бути значним, тобто таким, що може спричинити істотний вплив, і в цьому випадку рекомендується інспектування оцінюваної АЗС контролюючими органами на предмет дотримання всіх необхідних нормативів для вжиття заходів щодо зменшення загроз або закриття АЗС.

Приклад екранної форми виведення інтерпретації нестійкості АЗС, де оцінка нестійкості АЗС належить одразу до двох нечітких змінних і, отже, підпадає під правило $П_3$ виведення інтерпретованого результату, показано на рис. 3.

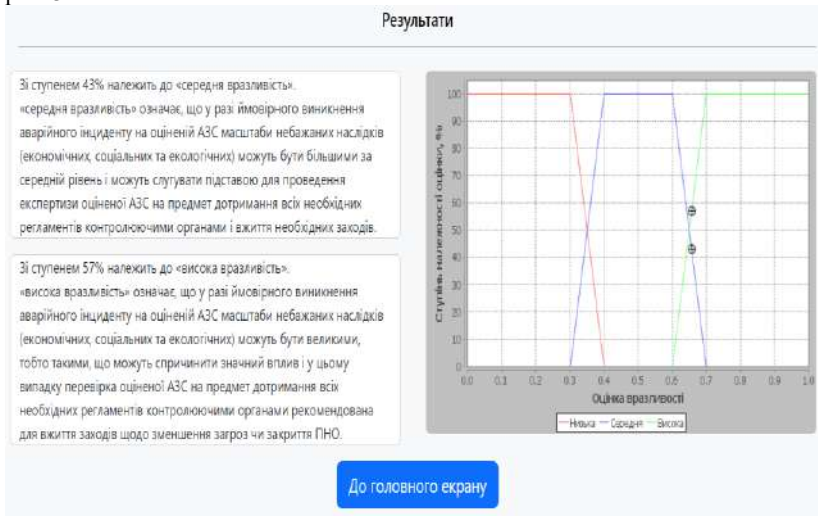


Рисунок 3. Екранна форма інтерпретації оцінки нестійкості АЗС [17]

6. Реалізація пропонованої інтелектуальної ІТ у формі СППР та її апробація на даних мережі АЗС

У межах впровадження проєкту було зібрано дані з відкритих джерел про 17 АЗС у певному населеному пункті. Надалі отримані відомості були закодовані відповідно до таблиці кодування, представленої в [15]. Результат кодованих значень критеріїв нестійкості АЗС, отриманий за допомогою інтелектуальної ІТ, наведено в табл. 5 (повний варіант – у [15]). Після кодування даних АЗС, представлених у табл. 5, за допомогою моделі (7) були розраховані оцінки нестійкості АЗС до основних типів аварій, значення яких відображено в останньому стовпці табл. 5 (*V*).

Таблиця 5

Вигляд значень атрибутів кодованих даних про АЗС [15]

№АЗС	E1	E2	E6	S1	S2	S5	A1	A6	*V*
1	0,5	0,6	0	0,3	0	0,3	0,4	0	*0,403*
2	0,5	0,4	0,7	0,7	1	0,3	0,4	0,3	*0,551*
3	0,5	0,8	1	0,3	1	0,3	0,5	0,3	*0,630*
4	0,5	0,1	0	0,3	0,7	0,3	0,3	0	*0,196*
5	0,5	0,6	0,3	1	1	0,3	0,8	1	*0,716*
6	0,5	0,4	1	0,7	1	0,3	0,3	1	*0,654*
7	1	0,2	1	0,3	0,7	0,3	0,4	0	*0,531*
…	…	…	…	…	…	…	…	…	…
17	1	0,4	0,3	0	0,7	0,3	0,4	0	*0,496*

Базуючись на результатах оцінок нестійкості АЗС, на рис. 4 побудовано графік розподілу оціночних параметрів залежно від їхнього просторового розташування. Параметр *x* є нормалізованим значенням широти (відповідно *y* – довготи), на якій розташована АЗС. АЗС розміщені на графіку відповідно до координат і мають колір, що відповідає параметру оцінки їхньої нестійкості до основних типів аварій.

З рис. 4 можна виділити такі класи АЗС при їхньому порівнянні між собою:

- темно-синій і світло-синій (4 об’єкти) – мають найменшу нестійкість;
- зелений (3 об’єкти) – мають нестійкість нижче середнього рівня;
- жовтий (3 об’єкти) – мають середню нестійкість;
- помаранчевий (4 об’єкти) – мають нестійкість вище середнього рівня;
- червоний і коричневий (3 об’єкти) – мають високу нестійкість.

Розглянемо інтерпретацію оцінки АЗС, поділивши її на категорії за допомогою нечітких змінних, описаних на рис. 2. Отримані значення для 17 досліджуваних АЗС відобразимо на рис. 5 у вигляді точкових значень на відповідних графіках функцій належності.

З рис. 5 ми бачимо, що досліджені АЗС мають наступний розподіл за нечіткими змінними оцінки нестійкості АЗС: 1 АЗС має низьку нестійкість, 13 АЗС мають середню нестійкість, 2 АЗС мають середню та високу нестійкість із різним ступенем, 1 АЗС має високу нестійкість.

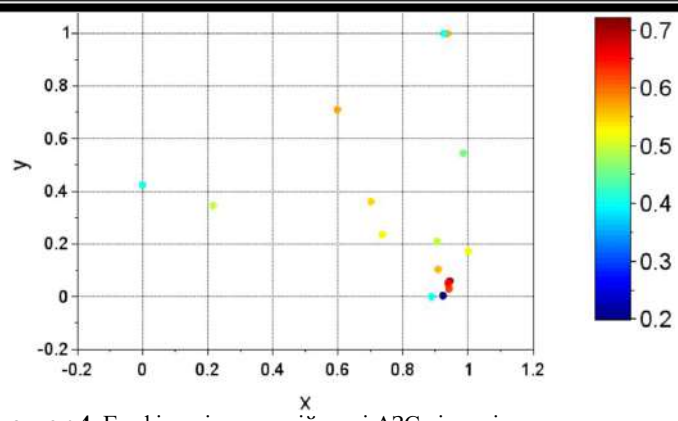


Рисунок 4. Графік оцінок нестійкості АЗС відповідно до просторових координат [17]

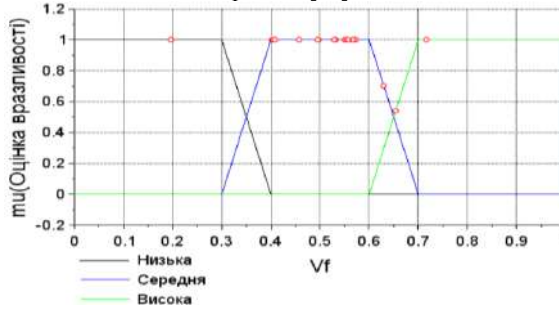


Рисунок 5. Графік оцінки нестійкості досліджуваної мережі АЗС за нечіткими змінними [17]

7 Висновки і перспективи дослідження

У результаті проведеного дослідження була розроблена інтелектуальна ІТ для багатокритеріального оцінювання нестійкості АЗС до основних типів аварій. Для її побудови було запропоновано та описано 10 послідовних етапів, що дозволяє створити СППР. Ця система здатна трансформувати числові оцінки нестійкості АЗС, отримані на основі зібраних даних, у категоріальні за допомогою нечіткої логіки для подальшої інтерпретації особами, що приймають рішення.

Серед переваг використання запропонованої інтелектуальної ІТ слід відзначити підвищення ефективності попереднього аналізу нестійкості АЗС, оскільки розроблена ІТ не лише збільшує швидкість аналізу та збору даних, але й надає обґрунтовані оцінки нестійкості АЗС, що містять рекомендації для стейкхолдерів. Варто також підкреслити, що запропонований підхід, за умови

подальшого розвитку та практичного впровадження, може слугувати методологічною основою для оцінювання нестійкості більш складних потенційно небезпечних об'єктів, особливо при їхній комбінації.

Як недолік зазначеного підходу слід відзначити його часткову неуніверсальність, оскільки аварії на потенційно небезпечних об'єктах та їхні комбінації можуть мати відмінну природу; потенційну складність формування початкової моделі нестійкості технологічного об'єкта, що може вимагати залучення досвідчених вузькоспеціалізованих фахівців, що є часо- та матеріаломістким завданням; а також суб'єктивність у прийнятті рішень, особливо коли вони ґрунтуються на експертних оцінках, що може спричинити упередженість та вплинути на об'єктивність результатів. Крім того, через обмеженість термінів реалізації та фінансові можливості проєкту, із розгляду моделі нестійкості АЗС була виключена група критеріїв «втрачені життя» у можливій аварії, проте це може стати фундаментом для подальших досліджень та вдосконалення розглянутої моделі.

Загалом, розроблена інтелектуальна ІТ може слугувати базою для попереднього огляду існуючих АЗС та початкової оцінки їхньої нестійкості до основних типів аварій, що може стати підставою для прийняття рішень щодо проведення розширеного оцінювання та обмеження їхньої діяльності або закриття АЗС. Це дослідження може становити інтерес для широкого кола зацікавлених сторін, включаючи регуляторні органи, населення, громадські організації тощо.

8. Подяки

Цей проєкт став можливим завдяки схемі грантів *UK-Ukraine twinning grants scheme*, що фінансується *Research England* за підтримки *Universities UK International* та *UK Research and Innovation*.

Література

1. Odesa Polytechnic, University of Portsmouth. (2023). *Multiple criteria fuzzy logic based methodology for risk mapping of gas filling stations and consequent decision optimization*. Odesa Polytechnic National University URL: https://op.edu.ua/en/international/projects/uk-ukraine-twinning-initiative-14.

2. Labib, A., Jones, D., Arsirii, O., Smyk, S., & Ivanov, O. (2023). Analysis of petrol station vulnerability factors regarding accidents using analytic hierarchy process and ranking. In A. Pakštas, Y. Kondratenko, V. Vychuzhanin, H. Yin, & N. Rudnichenko (Eds.), *Proceedings of the 11th International Conference Information Control Systems & Technologies (ICST 2023)* (Vol. 3513, pp. 330–341). CEUR. https://ceur-ws.org/Vol-3513/paper27.pdf.

3. Saaty, R. W. (1987). The analytic hierarchy process – What it is and how it is used. *Mathematical Modelling, 9*(3–5), 161–176. https://doi.org/10.1016/0270-0255(87)90473-8.

4. Aulia, B. U., Utama, W., & Ariastita, P. G. (2016). Location analysis for petrol filling station based on stakeholders' preference and seismic microzonation. *Procedia – Social and Behavioral Sciences, 227*, 115–123. https://doi.org/10.1016/j.sbspro.2016.06.051.

5. Wang, S.-P., Lee, H.-C., & Hsieh, Y.-K. (2016). A multicriteria approach for the optimal location of gasoline stations being transformed as self-service in Taiwan. *Mathematical Problems in Engineering, 2016*, 1–10. https://doi.org/10.1155/2016/8341617.

6. Mohsin, M., Zhan-ao, W., Shijun, Z., Hengbin, Y., & Weilun, H. (2022). Risk prioritization and management in gas stations by using fuzzy AHP and IPA analysis. *Journal of Scientific & Industrial Research, 80*(12), 1107–1116. https://doi.org/10.56042/jsir.v80i12.49210.

7. Koulinas, G. K., Demesouka, O. E., Bougelis, G. G., & Koulouriotis, D. E. (2022). Risk prioritization in a natural gas compressor station construction project using the analytical hierarchy process. *Sustainability, 14*(20), 13172. https://doi.org/10.3390/su142013172.

8. Ayyildiz, E., & Taskin, A. (2022). A novel spherical fuzzy AHP-VIKOR methodology to determine serving petrol station selection during COVID-19 lockdown: A pilot study for İstanbul. *Socio-Economic Planning Sciences, 83*, 101345. https://doi.org/10.1016/j.seps.2022.101345.

9. Akbari, D., & Saati, M. (2023). Location of fuel stations using fuzzy model in geospatial information system (Case study: 7th district, Tehran city). *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, X-4/W1-2022*, 31–36. https://doi.org/10.5194/isprs-annals-X-4-W1-2022-31-2023.

10. Jafar, M. N., Saqlain, M., Mansoob, A., & Riffat, A. (2020). The best way to access gas stations using fuzzy logic controller in a neutrosophic environment. *Scientific Inquiry and Review, 4*(1), 30–45. https://doi.org/10.32350/sir.41.03.

11. Ajman, N. N., Zainun, N. Y., Sulaiman, N., Khahro, S. H., Ghazali, F. E. M., & Ahmad, M. H. (2021). Environmental impact assessment (EIA) using geographical information system (GIS): An integrated land suitability analysis of filling stations. *Sustainability, 13*(17), 9859. https://doi.org/10.3390/su13179859.

12. Peprah, M. S., Boye, C. B., Larbi, E. K., & Opoku Appau, P. (2018). Suitability analysis for siting oil and gas filling stations using multi-criteria decision analysis and GIS approach – A case study in Tarkwa and its environs, *Journal of Geomatics*, *12*(2), 158–166. https://isgindia.org/wp-content/uploads/2018/11/Pap_9_Volume_2_Oct_2018.pdf.

13. Bedewy, B. A. H., & Abdulameer, M. H. (2023). Evaluating and improving the spatial distribution using GIS to avoid environmental risks and achieve safety for petrol stations in the Nile district center in Mahaweel / Iraq. *International Journal of Safety and Security Engineering, 13*(3), 433–444. https://doi.org/10.18280/ijsse.130306.

14. Odipe, O. E., Lawal, A., Adio, Z., Karani, G., & Sawyerr, O. H. (2018). GIS-based location analyses of retail petrol stations in Ilorin, Kwara State, Nigeria.

International Journal of Scientific & Engineering Research, 9(12), 790–794. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3312528.

15. Arsirii, O., Ivanov, O., Smyk., S., Oliinyk, V., & Bieliaiev, K. (2024). *Data for IT of Gas Stations Vulnerability Assessment*. https://drive.google.com/drive/folders/ /1MVzBfIx6uB4NvKm6ocLE-eHmeJ6_8K6S?usp=sharing.

16. Jones, D., Ivanov, O., Arsirii, O., Crook, P., Kanada, L., Labib, A., Teeuw, R., & Smyk, S. (2025). Multiple sustainability criteria mapping of gas station incident consequences and subsequent decision optimisation. *European Journal of Operational Research, 326*(2), 299–310. https://doi.org/10.1016/j.ejor.2025.04.026.

17. Arsirii, O., Ivanov, O., Smyk, S., Oliinyk, V., & Bieliaiev, K. (2024). Intelligent information technology for multi-criteria vulnerability assessment of gas stations to the main types of accidents. In A. Pakštas, Y. Kondratenko, V. Vychuzhanin, H. Yin, & N. Rudnichenko (Eds.), *Proceedings of the 12th International Conference Information Control Systems & Technologies (ICST 2024)* (Vol. 3790, pp. 458–470). CEUR. https://ceur-ws.org/Vol-3790/paper40.pdf.

DESIGN OF AN INFORMATION TECHNOLOGY FOR MULTI-FACTOR ASSESSMENT OF PETROL STATION SUSCEPTIBILITY TO COMMON HAZARD EVENTS

Ph.D. O. Ivanov[1][0000-0002-8620-974X], **Ph.D. D. Jones**[2] [0000-0002-9101-746X], **Dr.Sci. O. Arsiriy**[1][0000-0001-8130-9613], **Ph.D. A. Labib**[3][0000-0002-5481-5833], **Ph.D. S. Smyk**[1][0000-0001-7020-1826]

[1] *National University "Odesa Polytechnic", Ukraine,*
[2] *Lion Gate Building, University of Portsmouth, Portsmouth, United Kingdom,*
[3] *Faculty of Business and Law, University of Portsmouth, Portsmouth, United Kingdom*
EMAIL: lesha.ivanoff@gmail.com, dylan.jones@port.ac.uk, e.arsiriy@gmail.com,ashraf.labib@port.ac.uk, smyk@op.edu.ua

Abstract. *This work addresses the design and implementation of an intelligent information technology (IT) aimed at the multi-factor susceptibility assessment of petrol stations (PSs) to typical hazardous occurrences. The proposed IS is structured into 10 discrete sequential phases. The initial phases of the development process involved the precise identification of the dominant hazard types relevant to PSs and a thorough examination of PS susceptibility viewed through the lens of potential detrimental repercussions. For this purpose, 41 metrics (criteria) were meticulously established by domain experts. To refine this dataset, the analytical hierarchy process (AHP) is employed for the expert data analysis to derive a reduced criterion space of susceptibility, deliberately excluding the metrics related to fatalities. Based on this refined set of criteria, relevant operational data*

pertaining to the PS network was systematically gathered, and a detailed procedure for data pre-processing and encoding was subsequently put forward. The actual susceptibility rating for each PS is then conducted utilising a developed generalised model. To facilitate the comprehension and interpretation of the assessment outcomes, the application of fuzzy set theory, specifically using trapezoidal membership functions, is proposed. This intelligent IT for PS susceptibility rating is ultimately realised as a decision support system (DSS). The end-user (decision-maker), upon entering site-specific data corresponding to the susceptibility criteria of a given PS, obtains a preliminary susceptibility score along with a clear explanation of the derivation methodology. A core element of the DSS is a meticulously curated knowledge base containing pre-evaluated PS data, which possesses the functionality for dynamic editing and expansion. The deployment of the suggested intelligent IT for multi-factor PS susceptibility assessment, and the resultant DSS, provides a robust capability for evidence-based decision-making concerning the evaluation of potentially vulnerable facilities and effective risk mitigation.

Keywords. *Petrol station, intelligent information technology, decision support system, multi-factor decision-making analysis, hazard events, fuzzy logic, analytic hierarchy process.*

UDC 004.89:519.8:681.518
DOI https://doi.org/10.36059/978-966-397-538-2-20

УДОСКОНАЛЕННЯ ПОШУКУ ПАРАМЕТРІВ ЕЛЕКТРОКАРДІОГРАМ ЗА ДОПОМОГОЮ ОПТИМІЗАЦІЇ НА ОСНОВІ ВЕЙВЛЕТ-ФУНКЦІЇ

Д. Кошутіна[1][0009-0004-1326-8775], **Dr.Sci. С.Антощук**[1][0000-0002-9346-145X],
Dr.Sci. Г.Щербакова[3][0000-0003-0475-3854]
Національний університет «Одеська політехніка», Ukraine
EMAIL:[1]d.v.koshutina@op.edu.ua, [2]asg@op.edu.ua,
[3]galina.sherbakova@op.edu.ua

Анотація. *Досліджено та розроблено методи оптимізації для мультимодальної зашумленої цільової функції в задачах автоматизації технічної та медичної діагностики. Використовуючи можливості вейвлет-перетворення, цей метод дозволяє визначити координату екстремуму зашумленої або мультиекстремальної цільової функції. Така властивість може бути необхідною в широкому спектрі застосувань, наприклад, при моніторингу якості обладнання критичних застосувань, при моніторингу електрокардіограми. При розробці автоматизованих систем (АС) та їх процедур на основі обробки зображень часто виникає необхідність*

потрапляти в заданий діапазон ефективності, що відповідає цілям обробки (область прагматичної достатності). Це може відбуватися з удосконаленням ліків у медицині, зі зміною умов спостереження в системах відстеження рухомих об'єктів. Тому важливо створювати процедури для таких систем, що дозволяють коригувати параметри. Оскільки значна частина процедур таких АС (ідентифікація, кластеризація та сегментація) реалізується на основі методів оптимізації, необхідні методи оптимізації, що дозволяють визначати такі діапазони параметрів. Апарат вейвлет-аналізу дозволяє розширити можливості методу оптимізації. Під час визначення координати екстремуму можна використовувати відому властивість вейвлет-перетворення для зміни знаку при проходженні через екстремум. Запропоновано та досліджено метод оптимізації на основі вейвлет-функції Хаара. Описано реалізацію цього методу. Запропонована методика може бути використана в широкому спектрі важливих застосувань, що характеризуються високим рівнем шуму.

***Ключові слова*:** *оптимізація, вейвлет-перетворення, вейвлет-функція Хаара, вейвлет-домен, субградієнт.*

1. Вступ та постановка проблеми

Важливою тенденцією сучасних інформаційних технологій є орієнтація на розробку ефективних систем діагностики на основі розпізнавання образів [1-9]. Аналіз існуючих високовартісних систем технічної та медичної діагностики показав, що вони орієнтовані на обробку великих обсягів даних. Крім того, методи обробки діагностичних параметрів у таких системах вимагають багато досліджень для зміни асортименту продукції в технічній діагностиці та у випадку досліджень (наприклад) нових лікарських засобів у медичній діагностиці. Зрозуміло, що зростання обсягів досліджень може призвести до підвищення ефективності, але в таких умовах також зростає споживання ресурсів.

Відомо, що оптимізація є основною процедурою в класифікації, кластеризації, сегментації, фільтрації [1, 4, 7, 8, 9]. Попередній аналіз показав, що цільова функція таких процедур може бути з шумом, оскільки дослідження часто проводяться з використанням малих наборів даних. Більше того, ця цільова функція може бути мультимодальною. У таких умовах існуючі методи оптимізації не задовольняють вимоги практики. Градієнтні методи оптимізації мають низьку стійкість до шуму та високу чутливість як до локального екстремуму, так і до початкової точки пошуку. Субградієнтні методи оптимізації мають низьку точність, що суттєво знижує ефективність діагностики. Для вирішення суперечностей між вищезазначеними методами оптимізації авторами було запропоновано методи оптимізації, що базуються на оцінці напряму пошуку екстремуму з допомогою вейвлет-перетворення (ВП) [1, 4, 8, 9]. Результатом його роботи є точкова оцінка координати екстремуму, яка може бути використана в ряді інформаційних технологій та автоматизованих систем (АС) діагностування на їх основі [4, 8, 9].

Однак, при розробці таких автоматизованих систем та їх процедур на основі обробки зображень [1-9] часто виникає необхідність потрапляти в заданий діапазон ефективності, що відповідає цілям обробки (область прагматичної достатності) [10]. Це може відбуватися з удосконаленням лікарських засобів у медицині, зі зміною умов спостереження в системах відстеження рухомих об'єктів [4, 7, 8, 10]. Тому важливо створювати процедури для таких систем, що дозволяють коригувати параметри. Оскільки значна частина процедур таких АС (класифікація, кластеризація та сегментація) реалізується на основі методів оптимізації, необхідні нові методи оптимізації, що дозволяють визначати такі діапазони параметрів.

Апарат вейвлет-аналізу дозволяє розширити можливості методу оптимізації, запропонованого в роботі. Він дозволяє отримати результат у вигляді кількох діапазонів (звужувальних областей), в яких знаходиться екстремум. Ці області визначаються обмеженнями у вигляді нерівностей. При визначенні цих обмежень можна використовувати відому властивість вейвлет-аналізу змінювати знак при проході через екстремум [8, 9]. Такий підхід також може бути використаний у вищезгаданих застосуваннях [10, 12].

2. Аналіз методів оптимізації для систем діагностики

Для аналізу існуючих методів оптимізації розглянемо розв'язок оптимізаційної задачі, за допомогою якого можна знайти екстремум критерію оптимальності [13]

$$J(c) = E\{Q(x,c)\}, \quad (1)$$

де E - оператор математичного очікування; $Q(x,c)$ - функціонал від вектору параметрів $c = (c_1, \ldots, c_N)$, який залежить від $x = (x_1, \ldots, x_M)$ - вектор шуму.

Для більшості задач оптимізації в системах технічної та медичної діагностики явна форма для $J(c)$ невідома. Поверхня $J(c)$ в таких системах може бути мультимодальною і мати характер поверхні типу «яр». Це може бути пов'язано з особливостями задачі, технічними можливостями та вартістю вимірювань. Крім того, це може бути викликано малою кількістю вибірок параметрів, які було можливо отримати під час вимірювань.

Існуючі методи оптимізації діляться на два класи.

До першого класу відносяться методи глобальної (мультиекстремальної) оптимізації. Перша група таких методів глобальної оптимізації основана на послідовних багатократних запусках традиційних методів локального (детермінованого чи стохастичного) пошуку з різних стартових точок з області визначення. Важливо пам'ятати, що методи детермінованого локального пошуку на основі оцінок першої та/або другої похідної відрізняються низькою завадостійкістю [11, 13].

Один з можливих підходів до вирішення задач оптимізації з багатьма екстремумами полягає в сполученні методів локальної оптимізації з певною

процедурою перебору початкових точок. Наприклад, можна проводити спуск методом спряжених градієнтів з вершин рівномірної сітки, яка покриває область апріорної локалізації мінімуму. Вихідні «пробні точки» можна розташовувати і іншим чином. Наприклад, можна використовувати підхід на базі «більш рівномірного» розташування точок старту в багатовимірному паралелепіпеді, ніж у вершинах прямокутної сітки [14, 15] Таким чином можна знизити кількість «пробних точок» до кількох десятків. Процес чергової локальної мінімізації при цьому пропонують завершувати, якщо вже дісталися до дослідженої раніше зони локального мінімуму, або значення функції, що піддається мінімізації, в грубо знайденому локальному мінімумі значно більше вже досягнутого екстремуму.

До окремої групи можна віднести методи, в котрих глобальний пошук реалізований як єдиний ітеративний процес. Для цього алгоритм має володіти можливістю «виходити» з локальних мінімумів.

Одним з прикладів такого підходу є метод «важкої кульки» [11]. Ітеративна схема пошуку координати екстремуму цього відомого методу наступна

$$x(k+1) = x(k) - \alpha\, \nabla f\big(x(k)\big) + \beta\big(x(k) - x(k-1)\big).$$

Зрозуміло, що у випадку, коли $\nabla f\big(x(k)\big) = 0$, але $x(k) \neq x(k-1)$, то можна отримати $x(k+1) \neq x(k)$, тобто метод не буде «застрягати» в стаціонарній точці. Тут, як правило, приводять пояснення, а саме про переміщення важкої кульки по нерівній поверхні. У випадку, коли швидкість кульки достатньо велика, то вона буде оминати неглибокі западини. Продовжуючи цю аналогію, можна зауважити те, що кулька може лишитися в більш глибокому мінімумі та вже не вийти з нього. Це і є важливим недоліком цього методу.

Відомим також є так званий «метод яру». Автори пропонують використовувати такий метод у випадку, коли є апріорна інформація про цільову функцію, тобто відомо, що вона слабко змінюється по певним напрямкам (тим, що створюють дно яру), і більш змінюється по іншим напрямкам (повз схили яру).

Прикладом функції типу «яр» з одним екстремумом може бути квадратична функція з погано обумовленою матрицею. Метод оснований на чергуванні кроків «спуску» (вони проводяться обраним локальним методом (звичайно на основі градієнту)) – ці кроки дозволяють досягти область дна яру, і кроків по дну «яру». Винайдені в результаті точки з малими значеннями *f(x)* далі уточнюються з допомогою більш потужних локальних методів. Метод не вільний від певних недоліків. По-перше, досить складним є вибір довжини кроку по дну «яру» (як вона велика – метод оминає багато мінімумів, якщо мала – метод не відстежує напрям дна яру і переміщення стає хаотичним). По-друге, напрям кроку по дну яру залежить від значної кількості обставин (прийнятої точності локальних спусків, розташування попередньої точки та

інш.). Тобто потрібна апріорна інформація не тільки про тип цільової функції, а й про особливості пошуку на певній її поверхні.

Ідеї «методу яру» використовуються у відомому підході «спуск-підйом-перевал» [11]. В цьому методі процес пошуку глобального мінімуму розділений на три етапи, які повторюються циклічно.

На етапі спуску (наприклад, з допомогою методу спряжених градієнтів) здійснюється пошук локального мінімуму. На етапі підйому здійснюється вихід з зони мінімуму. Зсув при цьому відбувається по напряму «найповільнішого підйому». Цей напрям визначається наступним чином.

В точці *x(k)* здійснюється побудова функції $f(x(k)) = f(x) - (\nabla f(x(k)), x)$. При цьому перевіряються відомі умови: як перша похідна дорівнює нулю, а друга похідна більша за нуль, тоді $x(k)$ – точка локального мінімуму, а якщо знак другої похідної визначити не можна, то $x(k)$ - сідлова точка для $f(x(k))$ [11].

Далі з точки $z(0) = x(k) + \varepsilon d(k-1)$ (де $d(k-1)$ – напрям попереднього кроку, ε – параметр) здійснюють кілька кроків градієнтного методу для $f(x(k))$, а саме

$$z(i+1) = z(i) - \gamma \nabla f(z(i)).$$

Далі, якщо точки $z(i)$ прагнуть до $x(k)$ – напрям $d(k) = (z(i) - x(k))/\|z(i) - x(k)\|$ обирають в якості напряму підйому. Далі робиться крок $\bar{x}(k+1) = x(k) + \lambda(k) d(k)$ та градієнтний крок з $\bar{x}(k+1)$, який приводить до нової точки $x(k+1)$, де повторюється етап підйому. Якщо ж точки $z(i)$ віддаляються від $x(k)$, виконують етап визначення перевалу [11].

При цьому вектор $d(k) = (z(i) - x(k))/\|z(i) - x(k)\|$ задає напрям руху на перевал, котрий виконується в сполученні з градієнтним спуском після кожного кроку. При проходженні перевалу (критерієм є зміна знаку величини $(\nabla f(x(k+1)), d(k))$) починається спуск в новий локальний мінімум. Розділення пошуку на етапи у випадку багато екстремальної функції автор [11] вважає більш доцільним, ніж одноманітна процедура зсуву в «методі яру». Серед підходів до пошуку екстремуму багато екстремальних функцій можна також виділити стохастичні евристичні методи. Серед них превалюють наступні два підходи. В першому випадку це методи випадкового пошуку (при їх обчисленні випадковість вноситься в процес мінімізації), в другому випадку будують ту чи іншу модель функції, що мінімізується.

Методи першої групи для пошуку екстремуму у вказаних умовах використовують методи локальної оптимізації плюс певні «великі кроки», що дозволяють вийти з області локального екстремуму. Однак, як стверджує автор [11] такий підхід близький по характеристикам до методу повного перебору.

Методи другої групи виконують обчислення значень цільової функції в наборі обраних точок та на основі цього роблять певні висновки про її

значення в інших точках. Ці методи мають певні складності як з визначенням апостеріорних ймовірностей, так і з процедурою пошуку «найбільш перспективної» точки [11]. Незначно підвищити завадостійкість локального пошуку дозволяють методи на основі субградієнта [11] та стохастичні методи [11, 13]. Через багаторазовий запуск пошукової процедури підхід вимагає значних витрат часу та/або мультипроцесорної комп'ютерної реалізації [13, 16]. Через те, що для подібних методів доведено збіжність лише до локального екстремуму, їх часто називають методами «approximate minimization» [17]. Друга група таких методів – методи інтервальної глобальної оптимізації [18, 19, 20, 21]. Детерміновані методи цієї групи засновані на адаптивному дробленні області визначення цільової функції на інтервали зі стратегії «гілок та границь» [18, 20, 21].

Вони доказово дозволяють знаходити гарантовані двосторонні межі координат екстремуму. На етапах пошуку в межах інтервалу ці методи можуть використовувати локальний пошук, що знижує їхню перешкодостійкість. У зв'язку з цим і для скорочення часу пошуку їх рекомендують застосовувати за невисокої розмірності цільових функцій [20, 21].

Стохастичні методи цієї групи засновані на інтервальному оцінюванні по підобластях з рандомізацією управління їх виконанням на основі випадкового пошуку [20, 21, 22]. Такий підхід за рахунок відмови від доказовості та/або суто детерміністського характеру інтервальних методів оптимізації дозволяє покращити обчислювальну ефективність (скоротити час пошуку екстремуму) на низці цільових функцій [20, 21].

Третя група методів базується на основі різних перетворень.

До першої підгрупи належать раціональні селективно-усереднювальні методи [23, 24]. Використовують два етапи пошуку. На етапі пошуку області глобального екстремуму вони використовують регулярні та стохастичні методи. При евристичному виборі селективного перетворення на другому етапі використовують інформацію про характер цільової функції та її головного екстремуму в області визначення [23].

До другої підгрупи відносять методи на основі вейвлет-перетворень (ВП). Евристичні методи цієї підгрупи застосовують властивість ВП знаходити спеціальні точки цільових функцій з багатьма екстремумами (наприклад, сигналів електрокардіограм) [25, 26]. Через евристичний вибір ВП в рамках особливостей прикладної задачі, адаптивні процедури сортування та порівняння з порогом при пошуку глобального екстремуму такий підхід може за часовими витратами наближатися до методів першої групи.

Методи оптимізації з ВП, засновані на детермінованому пошуку [27, 28], дозволили підвищити завадостійкість і ймовірність виходу в область глобального екстремуму для різних процедур на основі пошуку зашумлених, мультиекстремальних цільових функцій [4].

З використанням їх для багатовимірних цільових функцій на значній області визначення з поетапним підвищенням точності пошуку з допомогою різних ВП – зростають і часові витрати [27, 28].

Зменшити витрати часу дозволяє, наприклад, ареальний метод з ВП на основі вейвлет-функції Хаара [12]. Використовуючи відому властивість ВП змінювати знак під час переходу через екстремум, цей метод дозволяє визначати набір обмежень у вигляді нерівностей. Ці обмеження визначають набір вкладених областей (інтервалів) з високою ймовірністю, що містять глобальний екстремум при зниженні часових витрат [12]. Вибір критерію зупинки може базуватися на прагматичній достатності [10].

Відомо, що в загальному випадку завдання глобальної оптимізації є важкорозв'язним [20, 21]. Так, доведено, що з поліноміальної цільової функції необхідні щонайменше, ніж експоненціальні (залежно від розмірності) витрати [20, 21]. За межами класу опуклих цільових функцій розв'язання задачі глобальної оптимізації є NP складним [20, 21, 29].

Ці методи відрізняються способом вибору початкових точок пошуку оптимуму, кількістю початкових точок, процедурами спуску до мінімуму та способами вибору значень змінних, що впливають на цільову функцію. Основними недоліками цих методів є обчислювальна складність і висока вартість обладнання при використанні багатопроцесорного методу [16, 23, 30].

До другого класу відносяться методи локальної оптимізації, які орієнтовані на оптимізацію з оцінками значень цільових функцій, або перших (других) похідних. Ці методи називаються методами нульового порядку, методами з оцінкою першої або другої похідних відповідно [11, 31-33].

В прикладних задачах, що розглядаються, вимірювання діагностичних параметрів можуть проводитися з похибками. Результати цих вимірювань можуть мати шуми. Кількість таких вимірювань може бути невеликою. За цих умов поверхні цільової функції, наприклад, у класифікації, можуть бути типу «долини» (яру), можуть бути з шумом та можуть бути мультимодальними.

Через складний характер цільових функцій важко отримати точні оцінки похідних і вибрати напрямок до екстремуму. Це основне джерело похибок існуючих методів локальної оптимізації в таких заданих умовах. Усі ці методи чутливі до шуму цільової функції. Наприклад, метод Ньютона використовує оцінки других похідних, і методи, засновані на цьому підході, можуть розходитися в разі високого рівня абсолютного шуму. За наявності шуму швидкість збіжності узагальненого методу Ньютона зводиться до швидкості збіжності градієнтних методів [11]. Іншим недоліком модифікацій методів Ньютона є великі обчислювальні витрати для знаходження власних векторів симетричних чисел матриці [11, 31].

Методи, засновані на оцінці градієнта (першої похідної) за наявності адитивного шуму, приводять лише до околиці екстремуму [11]. Крім того, для методів першої похідної – званих «методами квазіНьютона» (методи Бройдена, Девідона-Флетчера-Пауелла, Бройдена-Флетчера-Голдфарба-Шанно) – похибка збільшується у випадку оцінки гессіану за допомогою градієнтних вимірювань [31]. Серед методів локальної оптимізації методи нульового порядку орієнтовані на мінімізацію лише недиференційованих

розривних функцій. Однак їх загальним недоліком є низька швидкість збіжності.

Тому використовуються субградієнтні методи оптимізації. Вони засновані на ітераційній схемі заміни градієнта на випадкову субградієнтну функцію [11]

$$c(k+1) = c(k) + \gamma(k)\partial f(x, c(k)), \quad (2)$$

де k – це номер ітерації; $\partial f(x, c(k))$ - субградієнт; $\gamma(k)$ - крок на k-ій ітерації.

Але субградієнтні методи мають невисоку точність, повільну швидкість збіжності і, залежно від прийнятого субградієнта оцінки, можуть не задовольняти вимогам завадостійкості [11]. Збіжність субградієнтних методів оптимізації прискорюється за рахунок скорочення оптимальної області пошуку результатами попередніх ітерацій. Однак реалізація цих методів може включати велику кількість додаткових обчислень за одну ітерацію. Багатоступінчасті методи оптимізації дозволяють підвищити завадостійкість у цьому випадку, але визначення їх параметрів є надто складним [13].

Підводячи підсумок, можна зробити висновок, що методи обох класів характеризуються низкою суперечливих рис. Методи оптимізації з оцінкою градієнта мають низьку похибку, але (через специфіку оцінки градієнта) мають низьку завадостійкість, високу чутливість до локального екстремуму та початкової точки пошуку, слабку збіжність у цільових функціях типу «долина» (яр). Методи субградієнтної оптимізації, у свою чергу, характеризуються низькою точністю.

У підсумку слід зазначити, що для випадку, коли автоматизовані системи технічної та/або медицинської діагностики виконують обробку даних і діагностичні операції над малими масивами даних з можливою наявністю шуму, обробки цільових функцій з багатьма екстремумами, потрібно досліджувати та/або удосконалювати нові завадостійкі методи оптимізації. В цьому аспекті можна навести дослідження удосконаленого авторами методу, присвяченому визначенню параметрів сигналів електрокардіограм (ЕКГ) [28].

3. Обгрунтування методу оцінки координат екстремумів періодичних сигналів для визначення R-R інтервалів ЕКГ

Сучасний рівень обробки даних, наприклад, у телемедицині, часто вимагає аналізу нестаціонарних періодичних сигналів. Електрокардіограми є прикладом таких сигналів. Як правило, при діагностиці серцево-судинних захворювань за допомогою ЕКГ проводяться три етапи обробки даних: попередня обробка, ідентифікація (виділення ознак) та класифікація [7]. У цьому випадку часто вимірювання таких сигналів не дозволяє позбутися шумів. Ці шуми виникають як під час реєстрації сигналів, так і під час передачі даних по лініях зв'язку [35, 36, 37, 38]. Етап попередньої обробки спрямований на усунення таких шумів. Для усунення цих особливостей сигналу ЕКГ часто використовуються низькочастотні та високочастотні фільтри та/або вейвлет-обробка [7]. Сигнали ЕКГ мають складні часово-

частотні характеристики з високочастотною та низькочастотною складовими, часто зі спектром з перекриттям [39] корисного ЕКГ-сигналу та шуму. Це може призвести до зниження достовірності діагностики [38]. За даними [7, 40] відповідно до протоколів медичних досліджень, під час ідентифікації виділяють ознаки, які дозволяють розрізняти особливості досліджуваних захворювань. Через високу варіабельність частоти серцевих скорочень при ряді захворювань буває важко отримати великі набори даних навіть від одного пацієнта [7]. Крім того, існує велика кількість методів, спрямованих на автоматизацію процедури ідентифікації. Серед цих методів ідентифікації можна виділити три групи: методи, що виділяють ознаки змін амплітуди та частоти серцевих скорочень з часом, методи, що виділяють спектральні параметри ЕКГ-сигналів, та методи, що дозволяють визначати як часові, так і частотні характеристики ЕКГ-сигналу – вейвлет-методи обробки. Завдяки різноманітності методів, важливим завданням є зменшення розмірності набору ознак, що дозволяє зменшити обчислювальні витрати та підвищити продуктивність на етапі класифікації [11, 12, 13]. Так, наприклад, на цьому етапі використовуються такі методи вибору ознак, як методи, спрямовані на рекурсивне зменшення набору ознак; генетичні та ройові алгоритми [7, 44], методи дисперсійного аналізу [7, 44]. Зменшення обсягу оброблюваних даних до етапу ідентифікації дозволяє використовувати вейвлет-перетворення (ВП) для аналізу ЕКГ [4, 7, 45]. За допомогою цього аналізу досліджуваний сигнал розкладається на набір вейвлет-функцій (ВФ) [45]. Ці функції враховують часовий зсув та зміни масштабу вихідної ВФ [36]. Отже, обробка за допомогою ВФ дозволяє враховувати часові та частотні особливості сигналу, отримуючи набір коефіцієнтів для подальшої обробки [35].

Через складність завдань методи обробки ЕКГ на сучасному етапі різноманітні [46]. Таким чином, відомі підходи до аналізу ЕКГ, засновані на теорії хаосу, поєднанні статистичних, геометричних та нелінійних ознак варіабельності серцевого ритму, RPCA - рекурсивному аналізі головних компонент, SPSA - методі стохастичної апроксимації одночасного збурення, FDBT - методі на основі першої похідної, SDBT - методі на основі другої похідної, методі на основі перетворення Гілберта та інших [7]. Деякі з цих методів, наприклад, методи, засновані на оцінці першої та другої похідних в умовах оцінки параметрів нестаціонарного сигналу ЕКГ з багатьма екстремумами, можуть мати низьку завадостійкість та/або підвищену похибку. Ряд авторів використовують один з рівнів обробки з WT як вектор класифікації для подальшої класифікації [47].

Інший підхід передбачає оцінку статистичних характеристик сигналу ЕКГ після обробки за допомогою вейвлет-функції [42]. Цей підхід використовує ці статистичні характеристики як вектор класифікації в діагностиці. Однак, наприклад, для діагностики різних типів аритмій потрібен аналіз великих обсягів даних. Тому обидва підходи вимагають значного часу обчислення [42]. При високих вимогах до ефективності [37] зменшують розмірність вектору класифікації, визначаючи характерні ознаки фрагментів QRS ЕКГ (рис.1).

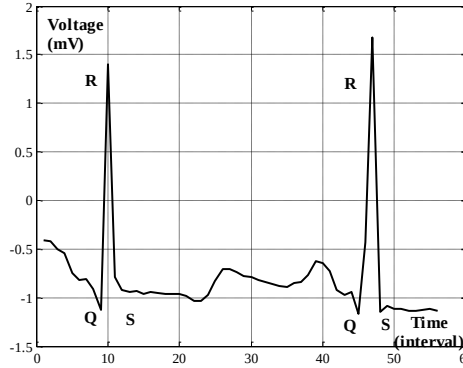
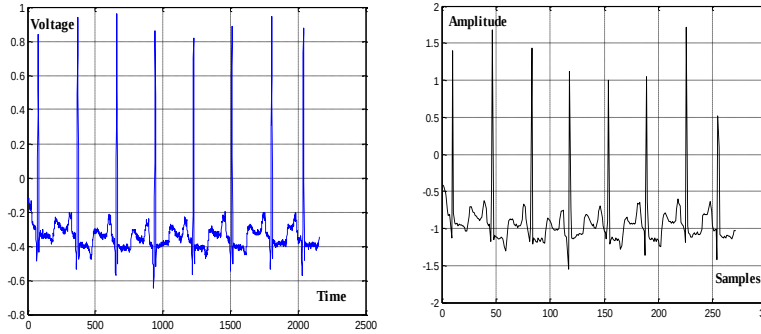


Рисунок 1. Стандартна форма хвилі ЕКГ з комплексом QRS [28]

Слід також зазначити, що розмірність вектору ознак змінюється під час ідентифікації для різних захворювань [40]. Так, наприклад, ряд авторів визначають рівні обробки за допомогою як вибору вейвлет-функції та масштабу вейвлет-функцій [48]. Такий підхід при аналізі довготривалих ЕКГ, наприклад, при використанні холтерівського моніторингу [49], дозволяє зменшити розмір початкового набору даних (рис.2, а) під час обробки (рис.2, б). Для пошуку екстремуму полімодальних функцій було розроблено метод оптимізації з використанням вейвлет-функції. При цьому була досліджена завадостійкість та погрішність пошуку екстремуму.



а *б*

Рисунок 2. Сигнал ЕКГ на різних етапах аналізу: (a) – вхідний сигнал ЕКГ, (b) – коефіцієнт ВФ Добеші [28]

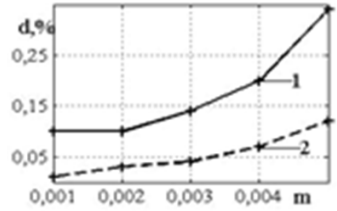
На першому етапі було досліджено погрішність визначення координати екстремуму для ВФ Хаара $\Psi_1(i)$ та гіперболічної ВФ $\Psi_3(i)$

$$\Psi_3(i) = \begin{cases} \frac{1}{\alpha_k(|i|+1)}, & \text{если } i > 0, \\ -\frac{1}{\alpha_k(|i|+1)}, & \text{если } i < 0, \end{cases}$$

$$i \in [-\tfrac{s_k}{2}, +\tfrac{s_k}{2}], \quad i \neq 0.$$

Для цього було синтезовано асиметричну тестову функцію $f_1(x) = x^2 \cdot e^{xm}$, з $m = 0 \ldots 0{,}005$ – коефіцієнтом регулювання асиметрії і $x \in (-500;\ 500)$. Встановлено, що для кроку дискретизації $a = 50$ і довжині носія ВФ $s_1 = 10$ відносна погрішність d визначення координати оптимуму функції $f_1(x)$ з ВФ $\Psi_1(i)$ не залежить від стартової точки пошуку, прямо пропорційна коефіцієнту асиметрії і для максимального досліджуваного значення $m = 0{,}005$ складає $0{,}3$ % (крива 1 на рис. 3). Абсолютна погрішність складає 0,3. Для оцінки з гіперболічною ВФ $\Psi_3(i)$ ці погрішності $0{,}1$ % (крива 2 на рис. 3) і 0,08 відповідно.

Досліджено швидкість збіжності методу пошуку екстремуму в порівнянні з методом градієнтного спуску. Дослідження проводились з допомогою функції «яр Розенброка» $f_2(\boldsymbol{x}) = 100(x_2 - x_1^2) + (1 - x_1)^2$. У цієї функції при $\boldsymbol{x} \in (-2{,}048;\ 2{,}048)$ глобальний мінімум $f_2(\boldsymbol{x}) = 0$ при значеннях $x_1 = 1, \quad x_2 = 1$. При дослідженні було прийнято – крок дискретизації ВФ $a = 0{,}0005$ і довжина носія $s_k = 10$. Метод на основі обробки з ВФ Хаара і гіперболічної ВФ дозволив досягти екстремума в 1,7 разів швидше (за кількістю ітерацій) у порівнянні з методом градієнтного спуску.



1 – $\Psi_1(i)$; 2 – $\Psi_3(i)$

Рисунок 3. Залежність відносної погрішності d від коефіцієнта асиметрії m при виначенні екстремуму з різними вейвлет-функціями

Долсіджено чутливість розробленого методу оптимізації до локальних екстремумів і стартовій точці пошуку з допомогою функції Швефеля $f_3(x) = 418{,}9829 + (-x \cdot sin\sqrt{|x|})$ (з хибним глобальним мінімумом). У цієї функції при $x \in (-500;\ 500)$ глобальний мінімум $f_3(x) = 0$ при $x = 420{,}9829$. При дослідженні точку старту обирали випадковим чином.

В результаті досліджень встановлено, що метод градієнтного спуску дозволяв знайти найближчий до старту мінімум, а метод з обробкою на основі вейвлет-функцій дозволив досягти глобального мінімуму з погрішностю $\delta \le 10^{-2}$ в 122 випадках з 150. Тобто було досягнуто ймовірність визначення координати екстремуму 0,8. Глобальний мінімум не був знайдений, коли значення точки старта обирались поза межами інтервалу $x \in (-420;\ 470)$. Було досліджено також завадостійкість методу з використанням функції Де Йонга 1 $f_4(x) = x^2$ (при $x \in (-205;\ 205)$) з додаванням завади (рис.4). Завада була розподілена за нормальним законом з нульовим середнім значенням і середньоквадратичним відхиленням (СКВ) від 0 до 40000, максимальне значення функції було $f_4(x) = 42000$. Встановлено, що при відношенні сигнал/завада по амплітуді від 1,05 метод дозволив вийти в область глобального мінімуму з погрішністю $\delta \le 10^{-2}$.

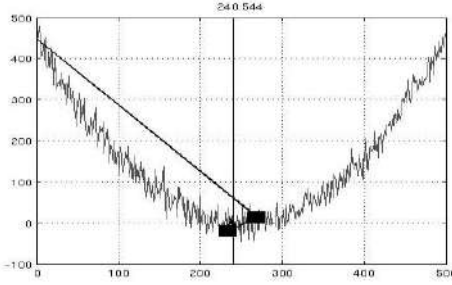


Рисунок 4. Траєкторія пошуку екстремуму на етапі обробки з ВФ Хаара $\Psi_1(i)$ для функції $f_4(x) = x^2$ з додаванням шуму

Автори запропонували метод пошуку екстремумів періодичних сигналів використовуючи ряд етапів цього методу [28] для визначення інтервалів між R-хвилями ЕКГ.

У цій роботі використовувалися лише етапи обробки сигналів за допомогою вейвлет-функції Хаара [14, 15, 48], були підібрані параметри методу та оцінено їх вплив на швидкодію процедури оцінки координат екстремумів.

4. Дослідження методу пошуку координат екстремумів періодичних сигналів ЕКГ

В результаті досліджень завадостійкості, швидкості збіжності та похибки в роботі для оцінки напрямку пошуку координати екстремуму в (3) потрібно використовувати симетричні та нестаціонарні вейвлет-функції – ВФ Хаара [14, 15, 48]. Координата екстремуму при цьому оцінюється по ітераційній схемі

$$\mathbf{c}[n] = \mathbf{c}[n-1] - \gamma[n]\mathrm{WT}_k(\mathbf{Q}(\mathbf{x}[n], \mathbf{c}[n-1])), \quad (3)$$

де $\gamma[n]$ – крок; n – номер ітерації; *k* – номер старту; WT_k – визначає напрям

переміщення пошуку до естремуму і обчислюється згідно:

$$WT_k(\mathbf{Q}(\mathbf{x}[n], \mathbf{c}[n-1])) = \{G_{1k}, G_{2k}, \ldots, G_{Nk}\}, \quad (4)$$

де G_{jk} – результат обробки по *j*-ій змінній:

$$G_{jk} = \frac{1}{s_k} \sum_{\substack{i=-\frac{s_k}{2} \\ i \neq 0}}^{\frac{s_k}{2}} Q(\mathbf{x}[n], \mathbf{c}_j + ia) \cdot \Psi_k(i), \quad (5)$$

де s_k – довжина носія ВФ; *a* – крок дискретизації ВФ; $\Psi_k(i)$ – ВФ Хаара на k - ому старті; $j = 1, \ldots, N$ – розмірність вектору параметрів.

Такий підхід дозволяє визначити діапазон зміни координат экстремума [12, 28]. Тут δ_1 – погрішність пошуку оптимуму старта (визначається на етапі апріорних досліджень); δ_2 – погрішність пошуку оптимуму прикладної задачі. Пошук проводиться з урахуванням наступних параметрів: $c[0]$ – початкове наближення до координати оптимуму в першому періоді; $\gamma[1]$ – крок; a – крок дискретизації ВФ; S_1 – довжина носія ВФ першого старту $\Psi_1(i)$; Δ_s – крок зміни довжини носія ВФ $\Psi_1(i)$ при визначенні діапазону координат екстремуму; номер старту $k=1$; номер ітерації $n=1$; A_1 - значення мінімальної висоти екстремуму, що реєструється; ΔR - початкове наближення до значення довжини інтервалу між екстремумами, що реєструються. При пошуку R- піку оцінюється напрям пошуку по (3) в точці наближення до координати оптимуму для старта *k*. При *k-1* для цього використовується зважена сума з ВФ Хаара з S_1 (в точці $\mathbf{c}[0]$ при $n=1$)). Вплив довжини носія для ВФ $\Psi_1(i)$ на швидкодію при аналізі цільової функції досліджено в роботі. На цьому етапі перевіряється знак оцінки по (3)

$$G_{jk} = \frac{1}{s_k} \sum_{\substack{i=-\frac{s_k}{2} \\ i \neq 0}}^{\frac{s_k}{2}} Q(\mathbf{x}[n], \mathbf{c}_j + ia) \cdot \Psi_k(i).$$

Далі з використанням відомої властивості ВП міняти знак при переході через оптимум, на цьому етапі визначається діапазон зміни координат екстремуму як $c[n-1] \le c^* < c[n]$. Далі проводиться пошук діапазонів координат оптимуму *Q(z,c)* по (3) при *k* до закінчення набору даних, що досліджується.

Для скорочення об'єму даних анализу та зниження рівню шуму сигналу ЕКГ проведено ВП сигналу ЕКГ з допомогою ВФ Добеши [45]. Визначення просторових координат R-піків проводиться шляхом пошуку координат екстремумів періодичних сигналів за допомогою розробленого методу. У роботі досліджено сигнали з MIT/BIH Arrhytmia Database [50]. Для

послідовності (рис.2, б) визначено координати перших 7 екстремумів та інтервали між ними.

5. Висновки та перспективи подальших досліджень

У роботі оцінювався вплив зміни значення довжини носія S_k на швидкодію. Після зміни знака ВП при переході через оптимум визначається діапазон зміни координат екстремуму як $c[n-1] \le c^* < c[n]$. Далі визначається довжина носія ВФ S_k для пошуку наступного екстремуму в наборі даних. Значення S_k в процесі дослідження змінюється від 18 до 34. При цьому через інтегральний характер ВП Хаара при збільшенні довжини носія ВФ координата наступного кроку пошуку в більшому чи меншому ступеню зміщується до екстремуму наступного періоду. Так, при зміні значень S_k від 18 до 34, і величині $\gamma = 1.7$ час пошуку координат R-піків змінювався не більше, ніж на 12% (при початковому значенні $s_1 = 4$). Також проведено оцінку впливу величини кроку $\gamma[n]$ на швидкодію. Так (при початковому значенні S_k =4) та зміні γ від 1.4 до 2.0 час визначення координат перших 7 екстремумів (рис.2, б) збільшився на 6%. У порівнянні з базовим методом [28], де пошук інтервалів здійснювався шляхом послідовної багатоетапної обробки спочатку з допомогою ВФ Хаара, а далі – з допомогою гіперболічних ВФ, швидкодію було покращено на 11%.

Відносні погрішності при визначенні склали: для координат R-піків не більше 1,6%, для довжини інтервалів між ними - не більше 2,6%. При дослідженні завадостійкості відносна погрішність визначення довжини R-R інтервалів - менше 4% при відношенні сигнал/шум по амплітуді 20…10. Ці результати дозволяють рекомендувати розроблений метод для використання в інформаційних технологіях телемедицини, при підвищенні рівня шумів в сигналах ЕКГ та малих вибірках досліджуваних даних. При подальших дослідженнях планується оцінка впливу величины $\gamma[n]$ та довжини носія ВФ S_k на швидкодію та зниження впливу крайових ефектів на достовірність визначення координат екстремумів.

Література

1. Polyakova, M., & Krylov, V. (2014). Classification of methods of the signal semantic wavelet transform for image contour segmentation. International *Journal of Computing*, 7(1), 51–57. https://doi.org/10.47839/ijc.7.1.489

2. Gonzalez, R., & Woods, R. (2002). Digital Image Processing (2nd ed.). Prentice Hall.

3. Janocki, M., Becker, A., Jakab, L., Grof, R., & Takacs, T. (2013). Automatic optical inspection of soldering. In Y. Mastai (Ed.), *Materials Science: Advanced Topics* (pp. 387–442). IntechOpen. http://www.intechopen.com/books/materials-science-advanced-topics/automatic-optical-inspection-of-soldering

4. Shcherbakova, G., Krylov, V., Pisarenko, R., & Kuzmenko, V. (2013). Wavelet transform domain adaptive clustering for electronic product quality

inspection. In Proceedings of the 7th *IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications* (IDAACS'2013) (Vol. 1, pp. 153–156). Berlin, Germany, 12–14 September 2013.

5. Gayani, K. S., Jacob, V., & Nair, K. (2014). Automation of ECG heart beat detection using morphological filtering and Daubechies wavelet transform. *IOSR Journal of Engineering* (IOSRJEN), 4(12), 53–58.

6. Prashar, N., Sood, M., & Jain, S. (2021). Design and implementation of a robust noise removal system in ECG signals using dualtree complex wavelet transform. *Biomedical Signal Processing and Control*, 63, 102212. https://doi.org/10.1016/j.bspc.2020.102212

7. Atamanyuk, I., et al. (2023). Computational method of the cardiovascular diseases classification based on a generalized nonlinear canonical decomposition of random sequences. *Scientific Reports*, 13, 59. https://doi.org/10.1038/s41598-022-27318-0

8. Shcherbakova, G., Antoshchuk, S., Koshutina, D., & Sakhno, K. (2024). Adaptive clustering for distribution parameter estimation in technical diagnostics. *Proceedings of International Conference on Applied Innovation in IT*, 12(1), 123–128. https://doi.org/10.25673/115667

9. Antoshchuk, S., Shcherbakova, G., Kondratiev, S., Koshutina, D., & Usov, O. (2024). Method of optimization based on wavelet transform for approximate depth estimation. In S. Vychuzhanin (Ed.), *Information Control Systems and Technologies (ICST-Odesa-2024): Materials of XII International Scientific-Practical Conference*, 23–25 September 2024 (pp. 123–127). Odesa: Helvetika.

10. Zheng, Y., Shcherbakova, G., Rusyn, B., Sachenko, A., Volkova, N., Kliushnikov, I., & Antoshchuk, S. (2025). Wavelet transform cluster analysis of UAV images for sustainable development of smart regions due to inspecting transport infrastructure. Sustainability, 17, 927. https://doi.org/10.3390/su17030927

11. Polyak, B. T. (1987). Introduction to Optimization. New York: Optimization Software.

12. Shcherbakova, G., Gerganov, M., Antoshchuk, S., Polyakova, M., & Sachenko, A., Krylov, V. (2018). Areal multistart method of optimization for image recognition. In IEEE *Second International Conference on Data Stream Mining & Processing*, 605–608. Lviv, Ukraine.

13. Tsypkin, Y. Z. (1971). Adaptation and learning in automatic systems. New York and London: *Academic Press.*

14. Walsh, J. L. (1923). A property of Haar's system of orthonormal functions. *Mathematische Annalen*, 90, 38–45.

15. Price, J. J., & Zink, R. E. (1965). On sets of completeness of Haar functions. *Transactions of the American Mathematical Society*, 119(2), 262–269.

16. Strongin, R. G., & Sergeyev, Y. D. (2000). Global optimization with non-convex constraints: Sequential and parallel algorithms. Dordrecht: *Kluwer Academic Publishers.*

17. Kearfott, R. B., & Kreinovich, V. (1996). Review of techniques in the verified solution of constrained global optimization problems. In R. B. Kearfott & V. Kreinovich (Eds.), *Applications of interval computations* (pp. 23–60). Dordrecht: Kluwer.

18. Hansen, E. (2004). Global optimization using interval analysis. New York: Marcel Dekker.

19. Moore, R. E., Kearfott, R. B., & Cloud, M. J. (2009). *Introduction to interval analysis*. Philadelphia: SIAM.

20. Shary, S. P. (2008). Randomized algorithms in interval global optimization. *Numerical Analysis and Applications*, 1, 376–389.

21. Shary, S. P. (2001). A surprising approach in interval global optimization. *Reliable Computing*, 7, 497–505.

22. Jaulin, L., Kieffer, M., Didrit, O., & Walter, É. (2001). *Applied interval analysis*. London: Springer. https://doi.org/10.1007/978-1-4471-0249-6

23. Zhigljavsky, A., & Zilinskas, A. (2008). *Stochastic global optimization*. Berlin: Springer.

24. Mikhalev, A. S., & Rouban, A. I. (2015). Algorithms of global optimization of functions on set of the mixed variables: Continuous and discrete with unordered possible values. *Advances in Modelling & Analysis*: Series D, 20(1), 56–75.

25. Kumari, M., & Mehta, P. (2012). QRS complex detection of ECG signal using wavelet transform. *International Journal of Applied Engineering Research*, 7(11).

26. Gayani, K. S., Jacob, V., & Nair, K. (2014). Automation of ECG heart beat detection using morphological filtering and Daubechies wavelet transform. IOSR *Journal of Engineering* (IOSRJEN), 4(12), 53–58.

27. Antoshchuk, S. G., Arsiriy, E. A., Babilunga, O. Y., Volkova, N. P., Galchonkov, O. M., Shcherbakova, G. Y., Nikolenko, A. A., Kondratiev, S. B., Kostenko, V. L., & Yadrova, M. V. (2022). Analysis and recognition of images in the wavelet transform space. Odessa: Bondarenko M. A. ISBN 978-617-8005-62-7

28. Shcherbakova, G., Shi, H.-S., Krylov, V., Bilous, N., & Antoshchuk, S. (2017). Estimation of the duration of RR-intervals of electrocardiograms by means of multi-start optimization based on wavelet transformation. In IEEE 9th *International Workshop on Intelligent Data Acquisition and Advanced Computing Systems*, 438–440. Bucharest, Romania.

29. Kreinovich, V., & Kearfott, R. B. (2005). Beyond convex global optimization is feasible only for convex objective functions: A theorem. *Journal of Global Optimization*, 33(4), 617–624.

30. Pardalos, P. M., Žilinskas, A., & Žilinskas, J. (2017). Non-convex multi-objective optimization. *Springer Optimization and Its Applications*, 123. https://doi.org/10.1007/978-3-319-61007-8

31. Gill, P. E., Murray, W., & Wright, M. H. (1981). Practical optimization. New York: *Academic Press*.

32. Nocedal, J., & Wright, S. J. (1999). Numerical optimization. Springer-Verlag.

33. Avriel, M. (2003). Nonlinear programming: Analysis and methods. Dover Publishing.

34. Liu, W., Li, S., Chen, M., Fang, Y., Cha, L., & Wang, Z. (2020). Fault diagnosis for attitude sensors based on analytical redundancy and wavelet transform. 2020 *Chinese Automation Congress* (CAC), 6471–6476. Shanghai, China. https://doi.org/10.1109/CAC51589.2020.9327370

35. Prashar, N., Sood, M., & Jain, S. (2021). Design and implementation of a robust noise removal system in ECG signals using dualtree complex wavelet transform. *Biomedical Signal Processing and Control*, 63, 102212. https://doi.org/10.1016/j.bspc.2020.102212

36. Kumar, M., Pachori, R. B., & Acharya, U. R. (2017). Automated diagnosis of myocardial infarction ECG signals using sample entropy in flexible analytic wavelet transform framework. *Entropy*, 19(9), 488. https://doi.org/10.3390/e19090488

37. Ferreira Jr, M., & Zanesco, A. (2016). Heart rate variability as important approach for assessment autonomic modulation. Motriz: *Revista de Educação Física*, 22, 3–8. https://doi.org/10.1590/S1980-65742016000200001

38. Sundararaj, V. (2019). Optimised denoising scheme via opposition-based self-adaptive learning PSO algorithm for wavelet-based ECG signal noise reduction. *International Journal of Biomedical Engineering and Technology*, 31(4), 325–345. https://doi.org/10.1504/IJBET.2019.103242

39. Jacob, V., & Nair, K. N. R. (2014). Automation of ECG heart beat detection using morphological filtering and Daubechies wavelet transform. IOSR *Journal of Engineering* (IOSRJEN), 4(12), 53–58. https://doi.org/10.9790/3021-041215358

40. Shebanin, V., et al. (2017). Canonical mathematical model and information technology for cardio-vascular diseases diagnostics. 2017 14th International Conference *The Experience of Designing and Application of CAD Systems in Microelectronics*, 438–440. https://doi.org/10.1109/CADSM.2017.7916170

41. Dyvak, M., & Pukas, A. (2005). Criterion of design of experiments for tasks of decision support interval model creation. 2005 IEEE Intelligent Data Acquisition and Advanced Computing Systems: *Technology and Applications*, 495–497. https://doi.org/10.1109/IDAACS.2005.283032

42. Gutierrez-Ojeda, J., Ponomaryov, V., Almaraz-Damian, J.-A., Reyes-Reyes, R., & Cruz-Ramos, C. (2023). ECG arrhythmia classification using recurrence plot and ResNet-18. *International Journal of Computing*, 22(2), 140–148. https://doi.org/10.47839/ijc.22.2.3083

43. Porplytsya, N., Dyvak, M., Spivak, I., & Voytyuk, I. (2015). Mathematical and algorithmic foundations for implementation of the method for structure identification of interval difference operator based on functioning of bee colony. *The Experience of Designing and Application of CAD Systems in*

Microelectronics, 196–199. https://doi.org/10.1109/CADSM.2015.7230834

44. Bhuvaneswari Amma, N. (2012). Cardiovascular disease prediction system using genetic algorithm and neural network. In 2012 *International Conference on Computing*, Communication and Applications, 1–5. https://doi.org/10.1109/ICCCA.2012.6179185

45. Daubechies, I. (1992). Ten lectures on wavelets (CBMS-NSF Lecture Notes,no.61).https://jqichina.files.wordpress.com/2012/02/ten-lectures-ofwavelet

46. Gupta, V., Saxena, N. K., Kanungo, A., Gupta, A., & Kumar, P. (2022). A review of different ECG classification/detection techniques for improved medical applications. *International Journal of System Assurance Engineering and Management*, 1–15. https://doi.org/10.1007/s13198-021-01548-3

47. Xie, L., Li, Z., Zhou, Y., He, Y., & Zhu, J. (2020). Computational diagnostic techniques for electrocardiogram signal analysis. *Sensors*, 20(21), 6318. https://doi.org/10.3390/s20216318

48. Huan, W., Shcherbakova, G., Sachenko, A., Yan, L., Volkova, N., Rusyn, B., & Molga, A. (2023). Haar wavelet-based classification method for visual information processing systems. *Applied Sciences*, 13(9), 5515. https://doi.org/10.3390/app13095515

49. Cesari, M., Mehlsen, J., Mehlsen, A. B., & Sorensen, H. B. D. (2017). A new wavelet-based ECG delineator for the evaluation of the ventricular innervation. *IEEE Journal of Translational Engineering in Health and Medicine*, 5, 1–15. https://doi.org/10.1109/JTEHM.2017.2722998

50. MIT/BIH Arrhythmia Database. (2025). Retrieved April 15, 2025, from http://physionet.org/physiobank/database/mitdb/

IMPROVEMENT OF ELECTROCARDIOGRAM PARAMETER SEARCH USING WAVELET-FUNCTION-BASED OPTIMIZATION

D. Koshutina[1[0009-0004-1326-8775]], **Dr.Sci. S. Antoshchuk**[2[0000-0002-9346-145X]], **Dr.Sci. G. Shcherbakova**[3[0000-0003-0475-3854]]
Odessa Polytechnic National University, Ukraine
EMAIL: [1]d.v.koshutina@op.edu.ua,[2]asg@op.edu.ua, [3]galina.sherbakova@op.edu.ua

Abstract. *Optimization methods are investigated and developed for multimodal noised goal function in the automation technical and medical diagnostics tasks. Using the wavelet transform capabilities, this method allows to determine the coordinate of the extremum of the noisy or multi extremal objective function. Such a property may be necessary in a wide range of applications, for example, in monitoring the quality for critical applications equipment, in electrocardiogram monitoring. In the development of automated systems (AS) and their procedures based on image processing, often a necessity arises to fall within a given range of efficiency, corresponding to processing objectives (the*

area of pragmatic sufficiency). This can occur with the improvement of medicines in medicine, with changing observation conditions in tracking systems for moving objects. Therefore, it is important to create procedures for such systems that allow for the adjustment of parameters. Since a significant part of the procedures of such ASs (identification, clustering and segmentation) is implemented on the basis of optimization methods, optimization methods are needed that allow such ranges of parameters to be determined. The wavelet analysis apparatus allows to extend the capabilities of the optimization method. When determining extremum's coordinate, a well-known property of the wavelet transformation can be used to change the sign when passing through the extremum. Method of optimization on Haar's wavelet function based is proposed and investigated. Implementation of this method is described. Proposed technique can be used in a wide range of the important applications which are characterized by the high level of a noise

Keywords: *optimization, wavelet transformation, Haar's wavelet function, wavelet domain, sub gradient.*

Наукове видання

СИСТЕМИ КОНТРОЛЮ ІНФОРМАЦІЇ ТА ІНТЕЛЕКТУАЛЬНІ ТЕХНОЛОГІЇ. ДОСЯГНЕННЯ ТА ЗАСТОСУВАННЯ

Монографія

Під наук. ред. проф. В. Вичужаніна

Укр. та англ. мовами

Підписано до друку 25.09.2025. Формат 60×84/16.
Папір офсетний. Гарнітура Times. Цифровий друк.
Умовно-друк. арк. 23,37. Тираж 300.
Замовлення № 1025-088. Ціна договірна.
Віддруковано з готового оригінал-макета.

Українсько-польське наукове видавництво «Liha-Pres»
79000, м. Львів, вул. Технічна, 1
87-100, м. Торунь, вул. Лубіцка, 44
Телефон: +38 (050) 6 58 08 23
E-mail: editor@liha-pres.eu
Свідоцтво суб'єкта видавничої справи
ДК № 6423 від 04.10.2018 р.