ISSN 2786-5460 (Print)
ISSN 2786-5479 (Online)

МІЖРЕГІОНАЛЬНА АКАДЕМІЯ УПРАВЛІННЯ ПЕРСОНАЛОМ
INTERREGIONAL ACADEMY OF PERSONNEL MANAGEMENT

МАУП

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СУСПІЛЬСТВО

INFORMATION TECHNOLOGY AND SOCIETY

Випуск 3 (18), 2025
Issue 3 (18), 2025

Видавничий дім
«Гельветика»
2025

Рекомендовано до друку Вченою радою
Міжрегіональної Академії управління персоналом
(протокол № 10 від 19 листопада 2025 року)

Інформаційні технології та суспільство / [головний редактор І. Остроумов]. – Київ : Міжрегіональна Академія управління персоналом, 2025. – Випуск 3 (18). – 206 с.

Журнал «Інформаційні технології та суспільство» є науковим рецензованим виданням, в якому здійснюється публікація матеріалів науковців різних рівнів у вигляді наукових статей з метою їх поширення як серед вітчизняних дослідників, так і за кордоном.

Редакційна колегія не обов'язково поділяє позицію, висловлену авторами у статтях, та не несе відповідальності за достовірність наведених даних і посилань.

Реєстрація суб'єкта у сфері друкованих медіа:
Рішення Національної ради України з питань телебачення і радіомовлення № 1173 від 11.04.2024 року.
Ідентифікатор медіа: R30-03890

Суб'єкт у сфері друкованих медіа – Приватне акціонерне товариство «Вищий навчальний заклад «Міжрегіональна Академія управління персоналом» (вул. Фрометівська, буд. 2, м. Київ, 03039, iapm@iapm.edu.ua, тел. (044) 490-95-00).

Мова видання: українська, англійська, німецька, французька та польська.
Періодичність видання: 4 рази на рік.

Відповідно до Наказу МОН України № 1290 від 30 листопада 2021 року (додаток 3) журнал включено до Переліку наукових фахових видань України (категорія Б) зі спеціальностей F2 – Інженерія програмного забезпечення; F3 – Комп'ютерні науки; F4 – Системний аналіз та наука про дані; F5 – Кібербезпека та захист інформації; F6 – Інформаційні системи і технології; F7 – Комп'ютерна інженерія.

Усі електронні версії статей журналу оприлюднюються на офіційній сторінці видання
http://journals.maup.com.ua/index.php/it

Статті у виданні перевірені на наявність плагіату за допомогою програмного забезпечення StrikePlagiarism.com від польської компанії Plagiat.pl.

Recommended for publication
by Interregional Academy of Personnel Management
(Minutes No. 10 dated 19 November 2025)

Information Technology and Society / [chief editor Ivan Ostroumov]. – Kyiv : Interregional Academy of Personnel Management, 2025. – Issue 3 (18). – 206 p.

Journal «Information Technology and Society» is a peer-reviewed scientific edition, which publishes materials of scientists of various levels in the form of scientific articles for the purpose of their dissemination both among domestic researchers and abroad.

Editorial board do not necessarily reflect the position expressed by the authors of articles, and are not responsible for the accuracy of the data and references.

Registration of Print media entity:
Decision of the National Council of Television and Radio Broadcasting of Ukraine: Decision No. 1173 as of 11.04.2024.
Media ID: R30-03890

Media entity – Private Joint-Stock Company «Higher education institution «Interregional Academy of Personnel Management»
(03039, Kyiv, Frometivska str., 2, iapm@iapm.edu.ua, tel. (044) 490-95-00).

Language of publication: Ukrainian, English, German, French, and Polish.
Periodicity: 4 times a year.

According to the Decree of MES No. 1290 (Annex 3) dated November 30, 2021, the journal was included in the List of scientific professional publications of Ukraine (category B) in specialties F2 – Software Engineering; F3 – Computer Sciences; F4 – Systems Analysis and Data Science; F5 – Cybersecurity and Data Protection; F6 – Information Systems and Technologies; F7 – Computer Engineering.

All electronic versions of articles in the collection are available on the official website edition
http://journals.maup.com.ua/index.php/it

The articles were checked for plagiarism using the software
StrikePlagiarism.com developed by the Polish company Plagiat.pl.

ЗМІСТ

CONTENTS

UDC 004.8:007:004.056.5
DOI https://doi.org/10.32689/maup.it.2025.3.1

Maryna BAUTINA
Master, Data Scientist, SoftServe
ORCID: *0009-0002-9617-9262*

DEVELOPMENT OF RELIABLE LLM SYSTEMS: DESIGN PRINCIPLES AND APPROACHES TO IMPLEMENTATION

Abstract. Purpose. *The article aims to provide a comprehensive analysis of architectural approaches and system solutions to ensure the reliability of services based on large language models (LLMs), as well as to develop principles and criteria for assessing the level of trust in applied scenarios.*

Methodology. *The study employs an interdisciplinary approach that combines the analysis of modern LLM architectures (zero-shot, fine-tuning, retrieval-augmented generation), a review of their implementation practices in corporate and industrial systems (GitHub Copilot, ChatGPT Enterprise), and a comparative synthesis of regulatory and ethical standards (OECD AI Principles, NIST AI RMF, EU AI Act). Methods of system analysis, comparative modeling, and the trust-by-design concept are applied.*

Scientific novelty. *The paper introduces the concept of building LLM-based services on the principles of trust-by-design, which relies on modular architecture, multi-level validation, and transparent response quality metrics. It is demonstrated that such integration of technical, ethical, and legal solutions enhances the resilience, transparency, and social responsibility of LLM in critical domains.*

Conclusions. *It is proven that establishing trust in LLMs is possible only under conditions of comprehensive integration of technical control mechanisms, ethical approaches, and legal regulation. The obtained results can be used to improve governmental and corporate strategies for artificial intelligence development, aimed at the safe and effective deployment of LLM in sectors with high reliability requirements.*

Key words: *large language models, LLM, trust, transparency, factuality, AI architecture, ethical AI, critical areas.*

Марина БАУТІНА. РОЗРОБКА НАДІЙНИХ СИСТЕМ LLM: ПРИНЦИПИ ПРОЕКТУВАННЯ ТА ПІДХОДИ ДО ВПРОВАДЖЕННЯ

Анотація. Мета. *Стаття спрямована на комплексний аналіз архітектурних підходів та системних рішень для забезпечення надійності сервісів на основі великих мовних моделей (LLM), а також на розроблення принципів і критеріїв оцінювання рівня довіри в прикладних сценаріях.*

Методологія. *У роботі застосовано міждисциплінарний підхід, що поєднує аналіз сучасних архітектур LLM (zero-shot, fine-tuning, retrieval-augmented generation), огляд практик їхнього впровадження у корпоративних і промислових системах (GitHub Copilot, ChatGPT Enterprise), а також порівняльне узагальнення нормативних і етичних стандартів (OECD AI Principles, NIST AI RMF, EU AI Act). Використано методи системного аналізу, порівняльного моделювання та концепцію trust-by-design.*

Наукова новизна. *Запропоновано концепцію побудови LLM-сервісів на засадах довіри за задумом (trust-by-design), що базується на модульній архітектурі, багаторівневій валідації та прозорих метриках якості відповідей. Показано, що така інтеграція технічних, етичних та правових рішень забезпечує підвищення стійкості, прозорості й соціальної відповідальності LLM у критично важливих сферах.*

Висновки. *Доведено, що формування довіри до LLM можливе лише за умов комплексної інтеграції технічних механізмів контролю, етичних підходів і правового регулювання. Отримані результати можуть бути використані для вдосконалення державних і корпоративних стратегій розвитку штучного інтелекту, спрямованих на безпечне та ефективне впровадження LLM у сферах з підвищеними вимогами до надійності.*

Ключові слова: *великі мовні моделі, LLM, довіра, прозорість, фактологічність, архітектура AI, етичний AI, критичні сфери.*

Introduction. Large-scale language models (LLMs) have become the foundational technology behind modern digital services, with widespread implementation across various domains, including education, healthcare, law, public administration, and cybersecurity. However, the extensive deployment of LLMs introduces new challenges: increasing complexity and unpredictability of outcomes, the risk of generating inaccurate or misleading content, susceptibility to adversarial attacks, and significant difficulties in ensuring ethical behavior, transparency, and accountability. These concerns are especially critical in areas where even minor errors may lead to serious consequences for individuals, institutions, or society at large.

In the context of LLMs' rapid integration into public and private information systems, the global community is intensifying efforts to develop approaches that align the technical, ethical, legal, and organizational dimensions of LLM implementation. Developers and regulatory bodies are prioritizing the

evaluation and oversight of LLMs, with particular attention to system testing, output validation, interface adaptability, model trustworthiness, and the harmonization of standards with leading international frameworks.

Literature Review. In the scientific literature of recent years, the problems and opportunities related to the development of LLM are systematically studied in an interdisciplinary manner. E. M. Bender, T. Gebru, A. McMillan-Major and S. Shmitchell [13] emphasize the risks of generating so-called "hallucinations" and the potential spread of biases in large-scale language models. R. Bommasani et al. [14] analyze in detail the fundamental opportunities and risks of foundation models, noting their impact on various industries and the need for new approaches to ethics and responsibility.

The technical and engineering aspects of creating and testing LLMs are described in the works of OpenAI [15], D. Ganguli, A. Askell, Y. Bai, E. Hubinger, T. Henighan et al. [7] and J. Rae, S. Borgeaud, T. Cai, K. Millican, J. Hoffmann, H.-F. Song et al. [16], which emphasize the importance of multilevel testing, scaling, red-teaming procedures, and continuous model validation. L. Weidinger, J. Mellor, M. Rauh, C. Griffin, J. Uesato, P. Huang and co-authors [19] propose approaches to identifying ethical and social risks, and D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang and others [8] emphasize the development of "superalignment" strategies aimed at aligning models with the common values of humanity.

Regulatory and ethical documents and international standards, such as the OECD AI Principles [12], NIST AI RMF [11], and EU AI Act [6], which set global requirements for transparency, security, accountability, and fairness of artificial intelligence systems, also play an important role.

Thus, the current discourse in the field of LLM outlines the main directions: improving architectures and testing procedures, developing ethical and legal frameworks, assessing social impact, and implementing new technological solutions to increase trust and security.

The purpose of the article is to systematize the principles of designing reliable LLM systems, analyze modern architectural approaches to their implementation, and present the experience of integrating such models into practical services based on domestic and foreign developments.

Results. In recent years, large-scale language models (LLMs) have become not just an engineering achievement, but a key factor determining a new paradigm of digital transformation in the field of artificial intelligence. The evolution of these systems is taking place against the backdrop of rapid changes in the digital economy, education, medicine, public administration, and media space, where LLMs are gradually becoming an infrastructure component of modern information ecosystems. The architectural basis of such models remains transformational approaches, described in detail in the works of leading researchers in the field [2; 16; 20]. It was the development of self-attention and the subsequent scaling of computing power that became a catalyst for the emergence of a wide range of LLM implementations, including GPT, LLaMA, Claude, Gemini, Mistral, Grok, and others. The choice of a model is determined not only by the number of parameters but also by the type of learning environments, the level of openness of the architecture, support for multimodality, and the possibility of flexible customization.

A number of studies have convincingly demonstrated that the performance and quality of language models directly depend on the scale of the architecture, the richness and diversity of training data, and the use of sophisticated engineering solutions to optimize training processes [2; 16; 20]. However, the very growth of scale gives rise to new challenges, including energy efficiency, control of computing costs, transparency of internal processes, and the need to comply with ethical standards. The emergence of the ideas of modular system construction, energy-efficient training, and the use of architectures with dynamic involvement of expert submodels allows flexible adaptation of LLMs to the requirements of specific applications.

Modern researchers pay special attention to the problem of so-called "hallucinations" – a phenomenon when a model produces grammatically correct, but actually incorrect or fictional content [13; 14; 19]. This is due to the over-reliance on statistical patterns in the data and the lack of built-in fact-verification mechanisms, which encourages developers to create additional means of verifying the accuracy of answers. In addition, security issues are becoming increasingly important in the practice of using LLMs, in particular, preventing prompt injection and jailbreak attacks that can push the system beyond the limits of controlled behavior. In such situations, it is extremely important to implement multi-level strategies for protection, query filtering, and result moderation, which is confirmed by research in the field of ethical use of artificial intelligence [7; 8; 11].

The task of ensuring the ethicality and absence of bias in LLM responses is of particular importance today, as these models are increasingly used in sensitive social areas. Studies on the ethical evaluation of language models emphasize that even modern systems can reproduce stereotypes, hidden forms of discrimination, or form a misperception of information [13; 14; 19; 12]. In response to these threats, a powerful area of Fairness-Aware NLP has emerged, which involves the development of specialized approaches to neutralize bias and increase the transparency of model decision-making.

The problem of instability of generation results associated with the probabilistic nature of LLM operation creates additional difficulties in ensuring reproducibility, quality control, and audit. In this context, leading engineers and researchers emphasize the need to implement traceability mechanisms, comprehensive audit, and dynamic monitoring of model performance [15; 20]. The experience of developing and applying LLMs shows that implementation in critical systems requires not only high performance but also proven stability and controllability of behavior.

At the same time, against the background of technological innovations, there is a growing awareness of the socio-technical impact of LLM on the labor market, the structure of professional activity, and educational processes. Analysts warn that automation caused by the introduction of LLM can significantly transform specialized areas, which, in turn, increases the requirements for institutional responsibility, transparency and verifiability of results, as well as for the implementation of ethical and legal regulation standards [14; 12; 11]. It is these aspects that stimulate the development of international regulatory frameworks, such as those of the OECD, NIST, and the European Union, which define the basic principles of transparency, data protection, and accountability in the field of artificial intelligence [12; 11; 6]. In the Ukrainian context, similar challenges are reflected in current applied research and LLM implementation projects in various industries.

To provide a meaningful comparative overview of existing large language models (LLMs), it is insufficient to list only technical parameters such as size or architecture. Instead, it is more informative to analyze how well each model supports the needs of critical applications – where trust, explainability, deployment control, and legal compatibility matter most. Table 1 summarizes the readiness of selected LLMs across five strategic dimensions that are vital for their integration in education, healthcare, legal and administrative services.

Table 1

Comparative characteristics of modern LLMs (as of 2024)

Model	Trust Infrastructure	Customization Options	Explainability Tools	Deployment Flexibility	Regulatory Compliance Potential
GPT-4	Advanced moderation; closed logs	Low (API only)	Limited	Cloud-based only	Medium (black-box limitations)
Claude 3	Ethical alignment focus (Constitutional AI)	Medium (via APIs, fine-tuning under NDA)	Moderate (context tracking, summaries)	Cloud-based only	Medium-High (strong safety by design)
LLaMA 3	Basic filters in open-source weights	High (open weights)	None built-in	Local, hybrid, cloud	High (full audit possible)
Gemini 1.5	Proprietary Google stack with strong sandboxing	Low (API only)	High (for vision-text tasks)	Cloud only (TPU-dependent)	Medium (unknown data provenance)
Grok	Minimal public safeguards known	Low	Not available	Tied to xAI cloud	Low (opaque logic)
Mixtral (MoE)	Sparse MoE routing, open logs	High	Requires external tools	Flexible (self-hosted)	Medium-High (transparent stack)
BLOOM	Full openness, community moderation	High	Optional (via plugins)	Full: local/cloud/ hybrid	High (complete reproducibility)

Note: the exact amount of GPT-4 is estimated to be undisclosed.

Source: author's elaboration

The restructured comparison makes it evident that the success of LLM deployment in sensitive domains depends not only on performance, but on the synergy between transparency, customization capacity, and infrastructural control. Open models like BLOOM and LLaMA 3 offer maximal auditability and flexibility, making them suitable for regulated industries with in-house engineering capacity. Conversely, models like GPT-4 or Gemini 1.5, despite their sophistication, are more difficult to align with transparency and explainability requirements due to proprietary constraints.

This shift in evaluation – from parameter-based ranking to implementation-based readiness – reflects the growing consensus in AI policy and research communities that trustworthiness is not a property of the model alone, but of the entire service ecosystem in which it operate.

The issue of system reliability implies that the service must consistently maintain its functionality, fulfill all assigned roles, and respond promptly to changes in load or atypical user behavior. The approach to

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

determining reliability, as proposed by the NIST standard, covers a number of key aspects: ensuring constant access to the service, insensitivity to incorrect or malicious requests, and the ability to respond quickly to incidents by switching to a safe mode [11]. Especially important is the ability of LLM to work with fail-safe logic, a response system that guarantees data safety and prevents the dissemination of incorrect or dangerous information in the event of abnormal situations.

The reliability paradigm of LLM services is increasingly combined with the principles of transparency, manageability, and compliance with ethical standards, as models not only fulfill technical tasks but also act as intermediaries in the interaction between humans and complex digital systems. This requires the creation of architectures that provide for detailed monitoring, flexible management of access parameters, built-in moderation, and automated audit of responses [19; 6]. It is also important to integrate mechanisms for identifying anomalies, storing interaction logs, and dynamically updating system policies in response to changes in the regulatory environment or new threats identified.

Recognition of reliability as a fundamental property of LLM is confirmed both in the regulatory documents of international organizations and in the results of research by leading scientists, who emphasize that only an integrated approach to the design and operation of such systems can guarantee their safety, sustainability, and efficiency in dynamic and potentially risky application scenarios [19; 8; 11; 6].

In current LLM research, trustworthiness is conceptualized as a composite of security, factual accuracy, transparency, response consistency, and personal data protection. As defined by NIST experts, a trustworthy AI system must produce accurate, interpretable, and secure outputs, operate transparently, and preserve user confidentiality [11; 5]. For LLMs, this entails both factual and explainable responses, as well as mechanisms to prevent misinformation and enable traceability.

A promising technical strategy to enhance reliability is model aggregation – combining multiple LLMs within one infrastructure to balance loads, compare outputs, and activate fallback models in case of failure. However, designing such systems involves a trade-off between response speed and explainability. Lightweight models like LLaMA 3 or Mistral offer faster processing but less interpretability, while larger models (e.g., GPT-4, Claude 3) provide deeper reasoning at the cost of latency. More accurate outputs via complex validation pipelines (e.g., RAG) can further slow response time and obscure the logic of generation, reducing user trust [11; 6].

This creates a fundamental compromise: either prioritize transparency with slower, auditable outputs, or optimize speed at the expense of interpretability. Regulatory guidance from NIST and the EU AI Act suggests that in high-risk domains (e.g., healthcare, legal services), verifiability should take precedence [11; 6].

Another foundational design principle is architectural modularity. By organizing services as microcomponents – separating input parsing, generation, validation, and logging – LLMs become more fault-tolerant, scalable, and easier to update without disrupting the whole system [20; 11]. Middleware tools such as API gateways and multi-level caching are essential for load control and efficiency [11].

Finally, personalization is becoming a key vector of reliability. Adaptive mechanisms such as dynamic prompt shaping, context retention, and vector-based user profiling allow models to tailor responses while enhancing relevance [7; 11]. Yet, personalization demands strict adherence to privacy protocols: encrypted storage, time-limited data retention, and full transparency in user data handling [8; 11].

Modern LLM implementations rely on three dominant architectural approaches: zero-shot learning, fine-tuning, and retrieval-augmented generation (RAG). Zero-shot models are applied without task-specific training, allowing rapid scaling but often sacrificing factual accuracy due to lack of contextual adaptation [20]. Fine-tuning enhances precision and relevance in defined domains but raises risks of overfitting and model bias [13]. RAG integrates external knowledge sources or search engines into the generation pipeline, improving transparency and fact-checking, albeit at the cost of integration complexity and potential source inconsistency [10].

Each approach entails trade-offs in trust and performance. As shown by P. Lewis et al. [10], zero-shot offers speed but lacks transparency and reliability. Fine-tuning, according to T. B. Brown et al. [20], delivers domain-specific precision but demands rigorous data governance. RAG is particularly promising for trustworthy applications, combining generative flexibility with verifiable content, though it requires strict safeguards for source quality and data protection [10].

Ultimately, LLM reliability emerges not solely from model architecture, but from the integration of technical, infrastructural, algorithmic, and ethical components that support stable and controllable system behavior [14]. As R. Bommasani et al. emphasize, effective deployment also depends on understanding user needs, contextual risks, and regulatory constraints [14].

Recent trends point to hybrid architectures, where combining different strategies-modularity, traceability, and auditability – enables improved transparency, resilience, and adaptability [5]. Modularization of LLM

workflows facilitates granular control over response generation, verification, and logging. Explainability tools and prompt validation systems are increasingly embedded in LLM pipelines to support trustworthy operations [10; 15; 17].

Comprehensive trust metrics now assess not only factual accuracy and process transparency but also resistance to prompt manipulation, user satisfaction, and compliance with privacy standards. These align with global frameworks such as NIST AI RMF and the EU AI Act [11; 6; 5]. As LLMs become integral to sectors like education, healthcare, law, and finance, their deployment demands cohesive architectural and operational strategies that meet high standards of security, ethics, and usability [14; 20; 8].

Modern LLM services are often deployed via bots, web portals, or SaaS platforms using standardized APIs, offering rapid integration, scalability, and support for diverse tasks [15; 11]. Reliable integration with providers like OpenAI or Anthropic requires RESTful APIs with layered authorization, middleware for filtering and routing, and safeguards against abuse [15; 7; 6; 20].

Scalability is achieved through containerization and asynchronous backends using tools like FastAPI, Redis, and Celery, optimizing performance under high load [20; 11]. Prompt engineering and context management – such as using templates, context windows, and intent detection – directly affect response quality [13; 10].

Output validation now combines manual review, automated metrics, and semantic checks to detect hallucinations. Content moderation and logging are integral to ensuring trust and traceability [19; 11; 5].

Security risks include hallucinated content, jailbreaks, and prompt injections, which demand multi-level moderation – both pre- and post-generation [7; 1]. Techniques like red-teaming expose hidden vulnerabilities and improve model robustness [7].

Ethical use in education, healthcare, and law requires strict oversight, transparency, and human control. Systems like GPT-4 and Claude 3 implement layered audits and safety protocols [15; 8; 17].

Ultimately, trust in LLMs depends on robust infrastructure, ethical safeguards, and continuous adaptation to new risks – combining engineering, regulation, and responsible design [19; 8].

The practical adoption of large language models has been enabled by a flexible technological stack that integrates modern AI libraries, distributed architectures, and tools for seamless deployment. As noted by T. B. Brown et al. [20], the Python ecosystem remains the dominant platform for LLM integration, combining FastAPI (REST API development), LangChain (context management), Gradio (UI integration), and Hugging Face Transformers (open-source models).

LLM-based automation is transforming front-end development. N. A. Ikumapayi [9] shows that OpenAI-assisted code generation reduces development time and boosts efficiency. Similarly, S. Shen et al. [18] emphasize the role of domain knowledge in automating JavaScript code.

Efficient orchestration between multiple LLMs is increasingly used to match model capabilities with query types. Middleware-based intent classification enables dynamic switching – e.g., GPT-4 for deep analysis, Claude for extended context, and LLaMA or Mistral for fast, lightweight queries – improving both cost-efficiency and relevance [14].

Preprocessing safeguards are also advancing. Semantic prompt validation and automated filters, as discussed by D. Ganguli et al. [7], mitigate the risk of inappropriate outputs-particularly in high-stakes domains like education, law, and healthcare.

Impact assessments confirm substantial gains: L. Weidinger et al. [19] and A. Almalki & M. Aziz [1] report that basic LLM integration reduces response times by up to 55%, improves satisfaction by 15–20%, and cuts redundant queries in half. These benefits extend to financial, educational, and corporate sectors.

Successful deployments hinge on modular backend tools, adaptive model routing, multi-level validation, and user-centered design. Together, they enhance efficiency, trust, and adaptability across applications. Enterprise ChatGPT offers encryption, moderation, and audit logging, though final validation of outputs often remains user-dependent [1].

These real-world cases demonstrate that robust infrastructure, adaptive logic, and ethical oversight are critical for scaling LLM-based services. Table 2 summarizes key architectural and technological components used in such implementations.

In the context of increasing reliance on large language models in high-stakes domains, the development of a clear and structured framework for evaluating the effectiveness and trustworthiness of such systems is a necessary precondition for their safe and meaningful integration. Given the multifaceted nature of trust in LLMs – combining technical, ethical, and user-centered dimensions – it is advisable to employ a layered approach that links operational performance with design principles such as explainability, transparency, and resilience to manipulation.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Table 2

Comparison of technical solutions for implementing LLM services

Implementation component	Technologies / approaches	Advantages	Limitations / Risks
Server platform	Python (FastAPI), Node.js (Express)	Flexibility, a large base of ready-made libraries, quick launch	Python – lower performance at high loads
UI framework	Gradio, Streamlit	Low-code, integration with ML models, easy customization	Limited scalability, poor style control
LLM integration	OpenAI API, Anthropic API, HuggingFace	Easy connection, support for state-of-the-art models	Cost of tokens, dependence on external servers
Model management	Multiplexer, fallback mechanisms	Flexibility, load balancing, improved relevance	Complication of request routing logic
Prompt processing	Prompt templates, context chaining	Standardization of input, repeatability of results	Vulnerability to prompt injection and jailbreak attacks
Service scaling	Docker, Redis, Celery	Performance control, horizontal scaling	Requires a high level of DevOps competencies
Assessment of response quality	BLEU, cosine similarity, RL feedback	Possibility of automatic monitoring	Does not always correlate with subjective quality of perception
Content filtering	Regex filters, NLP moderation	Reducing toxicity and inappropriate responses	The need to constantly update the rules

Source: author's development

The proposed framework is organized as a stepwise evaluation system that enables both developers and institutional stakeholders to monitor key indicators of system reliability, while adapting the configuration of LLM services to meet evolving demands. This methodology supports continuous assessment across five interrelated dimensions, each of which corresponds to critical functional and governance aspects of LLM use in real-world applications (Table 3).

The systematization of metrics for evaluating the trustworthiness and effectiveness of LLM-based services is an important step toward the operationalization of ethical and technical standards in the deployment of artificial intelligence in critical domains. The five-step model presented in Table 3 outlines an integrated approach that connects the behavior of large language models at the micro level (generation quality, factuality, explainability) with macro-level governance concerns such as security, resilience, user trust, and regulatory compliance.

Table 3

System for evaluating trust and effectiveness of LLM services

Evaluation Dimension	Key Indicators	Tools / Methods
Factual reliability	% of accurate responses; hallucination rate; cross-checks with trusted sources	Fact-checking APIs, knowledge bases, manual annotation
Transparency and explainability	% of responses with source trace; availability of reasoning; logging completeness	Explainability modules, audit logs, prompt tracing systems
Resistance to manipulation	Blocked prompt injection attempts; jailbreak prevention rate	Red teaming, semantic filters, adversarial testing scripts
User-perceived performance	SUS/NPS scores; task completion rate; response time under load	UX analytics, user surveys, load simulation
Security and ethical compliance	Number of flagged outputs; privacy breaches; audit trail availability	Moderation tools, encryption logs, regulatory compliance dashboards

Source: author's development

Each step reflects not only an isolated dimension of system performance, but also a specific aspect of trust formation in digital environments, thereby enabling a layered interpretation of model behavior in real-life applications.

The first stage, focused on factual reliability, addresses the core epistemic expectation from LLMs – that generated responses correspond to verifiable truths rather than plausible-sounding fabrications. This dimension plays a foundational role in ensuring semantic integrity, especially in knowledge-intensive

domains such as medicine, law, and education, where the cost of factual errors may be high. Measuring factuality requires both quantitative benchmarking against curated datasets and qualitative assessment through independent validation pipelines.

The second component, transparency and explainability, refers to the model's ability to provide justifiable and auditable outputs. In contrast to black-box generation, transparent systems enable the user or auditor to trace how an answer was constructed, what data or logic informed it, and where potential limitations lie. This aspect is particularly relevant for institutional accountability, where LLM outputs must be interpretable within legal or organizational frameworks. The presence of logging mechanisms, attention visualization, and reasoning trace tools significantly enhances the traceability of LLM behavior and contributes to the auditability of AI systems.

Manipulation resilience, the third pillar of the model, responds to the growing threat of adversarial use, such as prompt injections and jailbreaks, which may bypass system constraints or provoke the generation of harmful content. This dimension evaluates the robustness of LLM configurations under malicious scenarios and provides an early-warning function for identifying system vulnerabilities. Its role is especially significant in public-facing deployments or in sectors with strict reputational and safety requirements, such as healthcare portals or government service platforms.

The fourth dimension, user-perceived performance, introduces the human-centered perspective into the evaluation framework. Here, the emphasis shifts to how users experience interaction with the system, measured through satisfaction indices (e.g., SUS, NPS), task completion rates, and latency under operational loads. This dimension reflects the usability and perceived reliability of the LLM and is essential for long-term adoption, especially in dynamic service environments such as online education platforms, business assistants, or content generation services. The inclusion of this layer ensures that trust is not only engineered but also experienced and sustained over time.

Finally, the fifth stage – security and ethical compliance – consolidates the governance layer of the model. It encompasses indicators related to privacy protection, content moderation, and conformity with regulatory frameworks such as the GDPR, the EU AI Act, or sector-specific AI guidelines. This stage is vital in domains that involve personal data processing, sensitive information handling, or legally binding documentation. It integrates the results of internal audits, external risk assessments, and ethical reviews, allowing the organization to monitor and update its LLM deployment in line with evolving legal and normative expectations.

Taken together, the five-step system offers not merely a diagnostic instrument, but a dynamic infrastructure for continuous evaluation and adaptation. It facilitates informed decision-making for developers, system architects, compliance officers, and institutional stakeholders by enabling early detection of emerging risks, measuring the long-term reliability of LLM services, and supporting iterative improvement. From a practical standpoint, the application of this framework allows organizations to align technical performance with ethical values and user needs, thus forming a coherent and operational definition of trustworthiness in AI systems. It supports the transition from model-centric to service-centric evaluation, where trust is seen as a function of both algorithmic behavior and sociotechnical context. Ultimately, such a multidimensional model contributes to the institutionalization of responsible AI deployment practices, which is crucial for sustainable integration of LLM technologies into sensitive societal and economic infrastructures.

Conclusions. This study demonstrates that the development of reliable LLM systems requires an integrated approach in which architectural modularity is effectively combined with well-designed control and validation mechanisms at every level. The success of such solutions is reflected in their ability to maintain system stability and security under high loads, efficiently manage request processing, and ensure response quality through the use of intermediate control, semantic validation, resource balancing, and modern API gateways. As evidenced by the analysis of practical implementations, LLM services are becoming drivers of personalization and digital platform optimization, significantly reducing users' cognitive load, particularly in education, consulting, automated support, and related sectors.

Nevertheless, considerable challenges remain, especially regarding the potential generation of toxic or inaccurate responses in sensitive domains such as medicine, law, and psychology. Insufficient oversight and inadequate ethical moderation pose risks to user trust and may undermine the social legitimacy of deployed systems. Experience from recent master's and applied research projects confirms the importance of employing advanced technological tools, including FastAPI, LangChain, Docker, and Gradio, in building functional and scalable LLM infrastructures. The use of hybrid architectures that enable dynamic switching between models based on specific requests supports the optimization of quality, cost, and processing speed.

The future of LLM services is closely tied to the expansion of multimodal capabilities, the alignment of interfaces with ethical standards, and the improvement of model relevance through state-of-the-art

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

reinforcement learning techniques. Particular attention should be directed toward research on how different professional and age groups cognitively perceive LLM outputs, as such insights facilitate the customization of systems to individual user needs.

In conclusion, the development of trustworthy LLM systems extends beyond a technical challenge and evolves into an interdisciplinary endeavor that integrates advances in artificial intelligence, cybersecurity, UX design, cognitive science, and ethics. The complexity and coordination of these approaches are essential for creating systems that are not only high-performing but also socially responsible, safe, and acceptable for use in critical sectors.

Bibliography:

1. Almalki A., Aziz M. Exploring the potential and challenges of ChatGPT in enterprise contexts. *IEEE Access*. 2023. Vol. 11. P. 85339–85349. URL: https://doi.org/10.1109/ACCESS.2023.3328700 (date of access: 12.07.2025)

2. Brown T. B., Mann B., Ryder N., Subbiah M., Kaplan J., Dhariwal P. et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*. 2020. Vol. 33. P. 1877–1901. URL: https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html (date of access: 12.07.2025)

3. Brundage M., Avin S., Clark J., Toner H., Eckersley P., Garfinkel B. et al. Toward trustworthy AI development: mechanisms for supporting verifiable claims. *arXiv preprint* arXiv:2004.07213. 2020. URL: https://arxiv.org/abs/2004.07213 (date of access: 12.07.2025)

4. De Angelis S., Cirillo F., Mazzocca N., Palmieri F. A trustworthy AI framework for explainable artificial intelligence in critical domains. *IEEE Access*. 2023. Vol. 11. P. 44792–44806. URL: https://doi.org/10.1109/ACCESS.2023.3275093 (date of access: 12.07.2025)

5. European Union. Regulation (EU) 2024/1687 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act) and amending certain Union legislative acts. *Official Journal of the European Union*. 2024. URL: https://eur-lex.europa.eu/eli/reg/2024/1687/oj (date of access: 12.07.2025)

6. Ganguli D., Askell A., Bai Y., Hubinger E., Henighan T. Red teaming language models to reduce harms: methods, results, and lessons learned. *arXiv preprint* arXiv:2309.00603. 2023. URL: https://arxiv.org/abs/2309.00603 (date of access: 12.07.2025)

7. Hendrycks D., Burns C., Kadavath S., Arora A., Basart S., Tang E. et al. Overview of the Superalignment Plan. *OpenAI Blog*. 2023. URL: https://openai.com/blog/superalignment (date of access: 12.07.2025)

8. Ikumapayi N. A. Automated front-end code generation using OpenAI: empowering web development efficiency. *Available at SSRN 4590704*. 2023. URL: https://doi.org/10.2139/ssrn.4590704 (date of access: 12.07.2025)

9. Lewis P., Perez E., Piktus A., Petroni F., Karpukhin V., Goyal N. et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*. 2020. Vol. 33. P. 9459–9474. URL: https://arxiv.org/abs/2005.11401 (date of access: 12.07.2025)

10. National Institute of Standards and Technology. Artificial Intelligence Risk Management Framework (AI RMF 1.0). NIST. 2023. URL: https://nvlpubs.nist.gov/nistpubs/ai/NIST.AI.100-1.pdf (date of access: 12.07.2025)

11. OECD. OECD Principles on Artificial Intelligence. Organisation for Economic Co-operation and Development. 2021. URL: https://oecd.ai/en/dashboards/ai-principles (date of access: 12.07.2025)

12. On the Dangers of Stochastic Parrots / E. M. Bender et al. *FAccT '21: 2021 ACM Conference on Fairness, Accountability, and Transparency*, Virtual Event Canada. New York, NY, USA, 2021. URL: https://doi.org/10.1145/3442188.3445922 (date of access: 14.07.2025).

13. On the opportunities and risks of foundation models / R. Bommasani et al. URL: https://samuelalbanie.com/files/digest-slides/2022-06-foundation-models-opportunities-and-risks-intro.pdf (date of access: 12.07.2025)

14. OpenAI. GPT-4 Technical Report. 2023. URL: https://cdn.openai.com/papers/gpt-4.pdf (date of access: 12.07.2025)

15. Rae J., Borgeaud S., Cai T., Millican K., Hoffmann J., Song H. F. et al. Scaling language models: methods, analysis & insights from training Gopher. *arXiv preprint* arXiv:2112.11446. 2021. URL: https://arxiv.org/abs/2112.11446 (date of access: 12.07.2025)

16. Sandoval G. GitHub Copilot has a copyright problem. *The Verge*. 2023. URL: https://www.theverge.com/23602854/github-copilot-ai-copyright-microsoft-openai-lawsuit (date of access: 12.07.2025)

17. Shen S., Zhu X., Dong Y., Guo Q., Zhen Y., Li G. Incorporating domain knowledge through task augmentation for front-end JavaScript code generation. *Proceedings of the 30th ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 2022. P. 1533–1543. URL: https://doi.org/10.1145/3540250.3558965 (date of access: 12.07.2025)

18. Weidinger L., Mellor J., Rauh M., Griffin C., Uesato J., Huang P. et al. Ethical and social risks of harm from language models. *arXiv preprint* arXiv:2112.04359. 2021. URL: https://arxiv.org/abs/2112.04359 (date of access: 12.07.2025)

19. Zhuang F., Qi Z., Duan K., Xi D., Zhu Y., Zhu H. et al. A comprehensive survey on transfer learning. *Proceedings of the IEEE*. 2020. Vol. 109, No. 1. P. 43–76. URL: https://ieeexplore.ieee.org/document/9153870 (date of access: 12.07.2025)

Дата надходження статті: 09.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

UDC 004.89
DOI https://doi.org/10.32689/maup.it.2025.3.2

Mykhailo BERDNYK
Doctor of Technical Sciences, Associate Professor,
Professor at the Department of Computer Systems Software,
National Technical University "Dnipro Polytechnic"
***ORCID:** 0000-0003-4894-8995*

Igor STARODUBSKYI
Postgraduate Student at the Department of Computer Systems Software,
National Technical University "Dnipro Polytechnic",
igor.starodubsky@gmail.com
***ORCID:** 0009-0004-1864-7889*

USING MACHINE LEARNING METHODS FOR AUTOMATED CLOUD COMPUTING OPTIMIZATION

Abstract. *This study is devoted to the development and experimental validation of a comprehensive approach to the automated optimization of cloud computing through the use of machine learning methods. The proposed solution – an intelligent adaptive orchestrator – integrates three key components: workload forecasting based on time-series models (LSTM, Prophet), dynamic resource management using reinforcement learning methods (PPO), and an anomaly-detection module employing autoencoders and statistical techniques.*

The aim of the article. *To design, implement, and validate an intelligent adaptive orchestration system that eliminates critical limitations of traditional cloud resource management.*

Scientific novelty. *It lies in the design of a system with a modular architecture that ensures scalability, fault tolerance, and flexible adaptation to diverse business objectives through dynamic tuning of the reinforcement learning agent's reward functions, with integration with container orchestration platforms (e.g., Kubernetes) and support for multi-cloud deployments.*

The conclusions. *Within this study, an intelligent orchestrator for cloud resource management was developed, implemented, and experimentally validated, built on the integration of workload forecasting, reinforcement learning, and anomaly detection methods. Experiments conducted both on a controlled laboratory testbed and in the real industrial hybrid infrastructure of SoftRequest LTD confirmed the high effectiveness of the proposed solution. The practical value of the approach lies in the ability to integrate directly with existing orchestration platforms, such as Kubernetes, without the need for substantial infrastructure rework.*

Key words: *Kubernetes, multi-cloud environments, hybrid clouds, intelligent orchestrator.*

Михайло БЕРДНИК, Ігор СТАРОДУБСЬКИЙ. ВИКОРИСТАННЯ МЕТОДІВ МАШИННОГО НАВЧАННЯ ДЛЯ АВТОМАТИЗОВАНОЇ ОПТИМІЗАЦІЇ ХМАРНИХ ОБЧИСЛЕНЬ

Анотація*. Це дослідження присвячене розробці та експериментальній валідації комплексного підходу до автоматизованої оптимізації хмарних обчислень шляхом застосування методів машинного навчання. Запропоноване рішення – інтелектуальний адаптивний оркестратор – інтегрує три ключові компоненти: прогнозування робочих навантажень на основі моделей часових рядів (LSTM, Prophet), динамічне управління ресурсами за допомогою методів навчання з підкріпленням (PPO) та модуль виявлення аномалій із використанням автоенкодерів і статистичних методів.*

Метою статті *є проєктування, впровадження та валідація інтелектуальної адаптивної оркестраційної системи, що усуває критичні обмеження традиційного управління хмарними ресурсами.*

Наукова новизна *полягає в проекованні системи з модульною архітектурою, яка забезпечує масштабованість, відмовостійкість і гнучкість адаптації до різних бізнес-цілей через динамічне налаштування функцій винагороди агента навчання з підкріпленням, з інтеграцією з платформами оркестрації контейнерів (наприклад, Kubernetes) і підтримка мультихмарних розгортань.*

Висновки*. У межах цього дослідження було розроблено, впроваджено та експериментально валідовано інтелектуальний оркестратор для управління хмарними ресурсами, побудований на інтеграції методів прогнозування навантажень, навчання з підкріпленням і виявлення аномалій. Експерименти, проведені як на контрольному лабораторному стенді, так і в умовах реальної промислової гібридної інфраструктури компанії SoftRequest LTD, підтвердили високу ефективність запропонованого рішення. Практична цінність підходу полягає у можливості прямої інтеграції з наявними платформами оркестрації, такими як Kubernetes, без потреби в суттєвій перебудові інфраструктури.*

Ключові слова: *Kubernetes, мультихмарні середовища, гібридні хмари, інтелектуальний оркестратор.*

The problem statement. The widespread adoption of cloud computing has revolutionized access to computational resources, enabling organizations of all sizes to benefit from scalable, flexible, and cost-effective infrastructure. Cloud platforms allow dynamic provisioning of processing power, memory, storage, and network bandwidth, fostering rapid innovation and efficient operations across industries. However, the rapid growth in the scale and heterogeneity of cloud infrastructures has introduced unprecedented challenges in resource management [3].

Modern cloud environments must handle dynamic, unpredictable workloads driven by varying business demands, user behaviors, and external events. Traditional resource management strategies, predominantly based on static thresholds or manual scaling rules, are often unable to respond adequately to such fluctuations. These mechanisms, while simple and computationally inexpensive, operate reactively rather than proactively, leading to either over-provisioning (increasing operational costs) or under-provisioning (resulting in SLA violations and degraded user experience) [5; 11].

Moreover, cloud ecosystems today are no longer homogeneous. They encompass a variety of resource types, including general-purpose CPUs, specialized GPUs, FPGAs, TPUs, and other hardware accelerators, each optimized for different workloads. Hybrid and multicloud deployments, combining public cloud services with private data centers, further complicate the landscape by introducing diverse resource pools, varying pricing models, and heterogeneous performance characteristics [6; 7].

In addition to dynamic workload patterns, cloud systems face sporadic, high-impact events such as flash sales in e-commerce, viral content surges in media platforms, and end-of-month financial report spikes in banking. These events demand immediate scaling responses that static autoscalers cannot efficiently deliver. Any delay in resource provisioning during these critical periods can severely impact service availability, incur financial penalties, and erode customer trust [10].

Therefore, there is a critical need for intelligent, self-adaptive cloud resource management systems that can anticipate workload changes, make optimized scaling decisions in real time, and enhance resilience against unexpected anomalies or failures.

Analysis of recent studies and publications. Recent studies have consistently demonstrated the inadequacy of traditional threshold-based scaling methods in addressing the demands of dynamic cloud environments. Mao and Humphrey [5] highlighted the limitations of static autoscaling strategies, particularly under variable and unpredictable workloads, where fixed rules fail to optimize cost and meet application deadlines effectively. Similarly, Yazdanov and Fetzer [11] investigated vertical scaling mechanisms and found that simple reactive approaches often suffer from delayed response times and inefficient resource utilization.

In response to these challenges, the integration of machine learning (ML) techniques into cloud resource management has become an active area of research. Xu and Li [10] introduced a dynamic cloud resource management framework leveraging ML models to predict workload variations based on historical usage data. Their findings indicated that learning-driven approaches could significantly improve resource allocation efficiency compared to heuristic-based methods.

Forecasting future workloads is a critical component of intelligent orchestration. Qiu et al. [7] proposed an ensemble of predictive models combining statistical techniques and deep learning approaches to anticipate resource demand in cloud data centers. Their study demonstrated that multi-model ensembles outperform single predictors in terms of accuracy and robustness, especially under mixed and volatile traffic conditions. Similarly, Chen et al. [2] developed an RL-driven autoscaler for web applications that incorporates predictive components to enable proactive scaling decisions, thereby improving both SLA compliance and cost-efficiency.

Reinforcement learning (RL) techniques have emerged as a particularly promising solution for dynamic and autonomous resource optimization. Arabnejad and Barbosa [1] proposed a predictive RL-based autoscaler capable of adjusting resource allocations based on both current and forecasted system conditions. Their results showed that RL agents could dynamically learn optimal policies that balance multiple objectives such as cost reduction, SLA adherence, and energy consumption. Tang and Narasimhan [9] further advanced this field by applying continuous policy gradient methods for RL-based resource management, enabling faster adaptation to fluctuating workloads and heterogeneous resource pools.

Another crucial dimension in cloud orchestration is anomaly detection. Traditional resource management frameworks often ignore system anomalies, which can result in undetected failures and degraded performance. Su et al. [8] addressed this gap by developing an adaptive autoscaling mechanism that integrates deep RL with real-time anomaly detection capabilities. Their approach proved effective in identifying service-level anomalies and triggering appropriate scaling or migration actions to maintain service quality and infrastructure stability.

Comprehensive reviews by Hsu and Chung [3] emphasized that future cloud computing architectures must incorporate machine learning–based orchestration as a fundamental capability, integrating workload

prediction, autonomous decision-making, and anomaly detection into a unified control system. Kunal et al. [4] supported this perspective, demonstrating the effectiveness of combining deep reinforcement learning with predictive analytics to achieve intelligent, context-aware scaling of cloud applications.

Despite these advancements, existing research often treats forecasting, scaling, and anomaly detection as isolated modules rather than components of an integrated system. A holistic approach that unifies these capabilities within a modular, self-adaptive orchestrator remains underexplored. This gap motivates the development of a comprehensive, machine learning–driven orchestration framework capable of enhancing the adaptability, resilience, and economic efficiency of modern cloud infrastructures.

The purpose of the article. The primary goal of this research is to design, implement, and validate an intelligent adaptive orchestration framework that addresses the critical limitations of traditional cloud resource management. The proposed orchestrator integrates three interdependent machine learning components: Workload Forecasting Module: To predict short-term workload changes based on time-series analysis, enabling proactive scaling decisions; Reinforcement Learning–Based Orchestrator: To dynamically optimize resource allocation policies, balancing multiple objectives such as SLA compliance, operational cost reduction, and energy efficiency; Anomaly Detection Module: To monitor system behavior in real time and initiate corrective actions in response to detected anomalies. The orchestrator is designed to operate in real time within heterogeneous, hybrid, and multicloud environments, leveraging data collected from diverse telemetry sources. By employing modular architecture principles, the system ensures extensibility, fault tolerance, and adaptability to evolving infrastructure conditions and business priorities.

The research objectives can be summarized as follows: To develop a robust monitoring and data collection infrastructure capable of capturing fine-grained performance metrics and business-level indicators; to design and train machine learning models tailored for workload prediction and anomaly detection; to formulate the resource management problem as a Markov Decision Process (MDP) and develop a reinforcement learning agent using the PPO algorithm; to integrate the components into a coherent orchestration framework interfacing with container orchestration platforms like Kubernetes; to evaluate the system's performance against traditional autoscaling methods in both controlled laboratory settings and real-world production environments; to provide practical insights and architectural recommendations for the integration of machine learning–based orchestration into existing DevOps workflows. By achieving these goals, the study aims to contribute a scalable, modular, and intelligent solution capable of enhancing the adaptability, efficiency, and resilience of modern cloud computing infrastructures.

Summary of the main material. Cloud computing is a model that enables ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (such as servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort [3]. Resource management in cloud environments involves monitoring, scaling, allocation, and optimization of computational resources to ensure performance requirements are met while minimizing operational costs.

Machine Learning (ML) is a branch of artificial intelligence focused on developing algorithms that can learn from data and make predictions or decisions without being explicitly programmed [10]. In the context of cloud computing, ML is utilized to forecast workload changes, optimize resource allocation processes, and detect anomalies within complex infrastructure systems.

Workload forecasting involves applying time-series models to predict future resource consumption based on historical usage data. Models such as Long Short-Term Memory (LSTM) networks and Prophet are capable of capturing complex temporal dependencies and seasonal patterns, thus enabling proactive scaling of infrastructure resources [7; 2].

Reinforcement Learning (RL) is a type of machine learning where an agent learns to make optimal decisions through interactions with its environment, receiving rewards for successful actions [1; 9]. In cloud resource management tasks, RL enables the formulation of dynamic scaling and service migration strategies based on multi-objective optimization, balancing factors such as service level agreement (SLA) compliance, operational costs, and energy efficiency.

Anomaly detection focuses on identifying deviations from normal system behavior that may indicate hardware failures, cyberattacks, or configuration errors. Methods based on autoencoders, clustering algorithms, and statistical analyses allow the detection of both evident and subtle anomalies in large volumes of telemetry data [8].

Fundamental Principles for Designing an Intelligent Orchestrator:

– modularity: individual components (forecasting, optimization, anomaly detection) function independently and can be scaled or updated without disrupting the system's integrity;

– adaptability: the system continuously learns from new data and adapts its behavior to evolving environmental conditions;

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

– integration: tight integration with container orchestration platforms (e.g., Kubernetes) to automatically enforce orchestration decisions;

– continuous Optimization: regular retraining and updating of machine learning models based on incoming telemetry data and system feedback to maintain and enhance decision-making quality.

Research Methodology. The research was based on an experimental methodology that combined system design, prototype development, deployment, and multi-level experimental validation. The work was carried out in several distinct stages:

– analytical stage: analysis of the problems of dynamic resource management in cloud environments, identification of limitations in traditional autoscaling methods, and exploration of opportunities to apply machine learning approaches in cloud orchestration;

– design stage: development of a modular architecture for an intelligent orchestrator, including subsystems for telemetry collection, workload forecasting (LSTM, GRU, Prophet), reinforcement learning (PPO algorithm), and anomaly detection (autoencoders and clustering methods);

– implementation stage: deployment of a high-frequency monitoring infrastructure based on Prometheus and Elasticsearch, with data collection intervals of 5–15 seconds; development of online learning mechanisms for forecasting models; and construction of an orchestrator supporting dynamic optimization of scaling policies;

– model training stage: initial training of forecasting models using historical workload data with hyperparameter optimization (Grid Search); training of the reinforcement learning agent under simulated workload scenarios using experience buffer mechanisms to improve training stability;

– experimental validation stage: testing system robustness, adaptability, efficiency, and reliability under various operational scenarios, including load surges, node failures, and changing resource pricing in multicloud environments.

To enhance the reliability of the experimental results, the following were used:

– multiple test runs to smooth out random fluctuations;

– control groups based on standard autoscaling mechanisms (HPA, VPA);

– statistical methods for significance testing (e.g., t-tests).

Experimental Base. The functionality and effectiveness of the developed solution were validated in two environments:

– laboratory tested: a 30-node Kubernetes cluster (each node equipped with 8 vCPUs and 32 GB RAM) and two GPU nodes (NVIDIA Tesla T4) of SotRequest LTD. Workloads were generated using specialized traffic emulators simulating typical behaviors of e-commerce and financial services applications;

– industrial deployment: the financial services company SoftRequest LTD, operating a hybrid cloud infrastructure based on OpenStack and AWS. Real-world workloads included transactional processing, real-time analytics, and periodic training of AI models.

Infrastructure and application monitoring were performed using Prometheus and Elasticsearch, while machine learning models were developed and executed using TensorFlow and PyTorch.

The effectiveness of the developed system was assessed using the following indicators:

– average response time and 95th percentile response time under various workload scenarios;

– SLA violation rate – the percentage of requests that exceeded acceptable response time thresholds;

– resource utilization efficiency – CPU and memory utilization rates during orchestrator operation;

– operational costs – total cloud infrastructure expenses (e.g., AWS costs) and energy consumption in the local infrastructure;

– anomaly reaction time – the time interval between the occurrence of an anomaly and the initiation of corrective actions by the system.

Each metric was analyzed under conditions of normal operation, load surges, infrastructure degradation, and was compared against the performance of traditional autoscaling mechanisms.

The architecture begins with a Telemetry Collector (Node-Exporter, cAdvisor) that scrapes fine-grained CPU, memory and I/O metrics from every Kubernetes node and pod. These raw observations are streamed into Prometheus, while logs flow to Elasticsearch, forming the central TSDB layer. A Feature-Engineering / Streaming ETL service continuously aggregates and enriches the metrics, producing compact feature vectors in near-real time. The vectors feed two specialised ML services: a Workload Forecasting module (LSTM / Prophet) that predicts short-term resource demand, and an Anomaly Detection module (autoencoders plus statistical tests) that flags abnormal behaviour. Both forecast outputs and anomaly flags are consumed by a PPO-based Reinforcement-Learning Optimizer, which synthesises them with the current cluster state to choose scaling, placement or throttling actions. Decisions are forwarded to the Adaptive Orchestrator Control Plane, which translates high-level actions into concrete Kubernetes primitives such as HPA/VPA changes, pod rescheduling or spot-instance requests. Through the Kubernetes API /

Controller Runtime, these commands are applied uniformly across a hybrid runtime that spans an on-prem OpenStack cluster and AWS EKS. Executed actions generate new states and rewards that are written, together with the original metrics, into an Experience Buffer for continual learning. A background Model Trainer / Online Tuning service samples from this buffer to update the forecasting, anomaly-detection and RL models, closing the self-adaptation loop. A separate Load Emulator can inject synthetic traffic into the cluster during laboratory testing to validate policies under extreme or corner-case scenarios. Operators observe the entire system through Grafana dashboards fed directly by Prometheus (metrics) and Kibana views backed by Elasticsearch (logs). The dashboards also surface the applied policies coming from the Control Plane, providing full transparency into why and when the orchestrator scaled or migrated workloads. Overall, the design realises a vertically integrated feedback cycle – observe → predict & detect → decide → act → learn – enabling proactive, cost-aware and resilient resource management across multicloud environments.

Technical Implementation. The production-grade prototype was deployed as fourteen containerised micro-services on Kubernetes, each packaged with locked, reproducible Helm charts. At the foundation, a high-frequency observability stack polls every node and pod with Node-Exporter, cAdvisor and kube-state-metrics; raw metrics arrive in Prometheus at five-second granularity while structured logs land in an OpenSearch cluster configured with hot-warm tiering so that recent indices remain on NVMe storage and older shards migrate to cost-optimised disks. Thanos remote-write off-loads up to one-and-a-half years of historical data to an S3-compatible MinIO bucket, ensuring that the modeller can replay the entire operational history without impacting online queries.

A Python FastAPI service, built around the Faust streaming framework, subscribes to the metrics_raw Kafka topic, applies windowed aggregations at five seconds, one minute and fifteen minutes, computes derivative features such as CPU deltas, exponential-weighted moving averages and trend slopes, then serialises the resulting vectors with Apache Arrow before publishing them to the features_v1 topic. Sustained throughput in production is roughly two megabytes per second, and in end-to-end tests the p-99 latency between a Prometheus scrape and a feature message entering the machine-learning layer is below one second.

Forecasting is served by an ensemble endpoint that multiplexes two distinct models: a Prophet instance tuned to capture daily and weekly periodicity with a changepoint prior of 0.2, and a two-layer LSTM built in TensorFlow 2.16 with 128 hidden units per layer and a dropout of 0.2. Hyper-parameters were discovered by a grid search over learning rates and look-back windows using a ninety-day sliding training set; retraining triggers automatically at midnight or whenever the weighted absolute percentage error exceeds fifteen per cent. Inference runs in TensorFlow Serving with a batch size of sixteen and maintains a p-95 latency of forty-five milliseconds per series.

An anomaly-detection micro-service hosts a symmetric dense auto-encoder, forty-eight hours of “healthy” data are replayed nightly to refresh its weights, and the reconstruction error threshold (mean plus three standard deviations) is recalibrated at the end of each run. A lightweight z-score guard operates in the same container so that sudden spikes can be flagged within hundreds of milliseconds, even if the auto-encoder inference queue backs up. Both anomaly flags and point forecasts are appended to the state vector consumed by the reinforcement-learning agent.

Resource allocation is framed as a continuous-action Markov Decision Process and solved with Proximal Policy Optimisation in Ray RLlib 3.0. The fifty-four–dimensional state contains current utilisation, forecast residuals, anomaly indicators, spot-price fluctuations and energy-tariff signals; the three-dimensional action proposes a replica-scaling multiplier between 0.5 and 2.0, a CPU-limit shift between –1 and +1 core, and a node-affinity score that steers the scheduler towards GPU or spot nodes. Eight learner workers execute in parallel on separate vCPUs, while the central parameter server uses a single NVIDIA T4 GPU for policy updates. A Redis-backed experience buffer can hold two million tuples; as soon as four thousand new transitions accumulate – typically about thirty seconds – the agent performs an on-policy update with a clip parameter of 0.2 and a GAE lambda of 0.95.

Decisions emitted by the agent enter a custom Go controller that reconciles an AdaptivePolicy custom resource. The controller translates abstract actions into concrete Kubernetes primitives: it adjusts HPA and VPA targets, annotates pods to trigger rescheduling, calls kubectl drain for live migration and interacts with the AWS EC2 Fleet API to request spot instances. Every change rolls out through a guarded canary stage that initially exposes only one per cent of user traffic; if p-95 latency remains below one hundred and ten per cent of baseline for two reconcile loops, the controller ramps exposure by ten per cent per minute; otherwise, it rolls back and pins the policy revision for post-mortem analysis.

A nightly Kubeflow Pipelines workflow pulls fresh experience data, launches distributed training jobs on two dedicated GPU nodes, performs Optuna-driven hyper-parameter optimisation for the predictive models, version-tags the artefacts with Data Version Control hashes and uploads them to the MinIO model store.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Newly trained models are shadow-served for fifteen minutes with ten per cent traffic replay; promotion occurs automatically if the weighted absolute percentage error rises by no more than two percentage points and the reward trend stays positive. Model-drift rules in Prometheus watch the production error metrics and push alerts to Slack when WAPE exceeds eighteen per cent for a sustained quarter hour.

Continuous integration runs in GitLab and executes linting, unit tests, container builds with BuildKit, Trivy image scanning and Helm-chart rendering. Staging deployments occur on an isolated thirty-node cluster where Locust and K6 replay week-long production traces; a pipeline succeeds only if average response time stays within five per cent of baseline while infrastructure cost, computed from mock AWS billing data, falls by at least three per cent. The median pipeline duration is fourteen minutes, and mean time to restore after a failed deployment is under twelve minutes thanks to automated rollbacks and pre-baked golden images.

Security and compliance are enforced with Vault-backed CSI drivers for short-lived secrets, OPA Gatekeeper policies that block privileged or host-PID pods, and cosign signatures that guarantee supply-chain integrity from source commit to running image digest. All inter-service traffic is encrypted with Istio mutual TLS, and cross-cluster links use WireGuard tunnels. Infrastructure spans an eighteen-node private OpenStack cloud for latency-critical workloads and a twelve-node AWS EKS footprint that mixes on-demand and Graviton-powered spot instances for cost-efficient burst capacity; Calico enforces network policies across both sites.

Benchmarking on the production workload showed that the orchestrator lowers average response time by thirty-three per cent, halves SLA violations and cuts cloud spend by roughly twelve per cent over a rolling thirty-day window. Anomaly mean-time-to-detect is about six seconds and mean-time-to-mitigate under fifteen. Collectively, these results confirm that the observe-predict-optimise-act-learn feedback loop operates fast enough to out-perform traditional HPA/VPA scaling while remaining safe, auditable and cost-aware in a heterogeneous multicloud environment.

Performance Evaluation Results. The experimental evaluation was conducted in two environments: on a controlled laboratory testbed at SoftRequest LTD and in its real-world production hybrid cloud infrastructure.

Key performance indicators (KPIs) used for evaluation included:

– average and 95th percentile service response time;

– SLA compliance rate;

– resource utilization efficiency;

– trends in operational cost dynamics;

– anomaly detection and response speed.

The results demonstrated that the proposed architecture ensures high application stability even under sudden and significant workload changes.

System Response Time. Comparative analysis showed a significant reduction in both average response time and the 95th percentile (p95) compared to traditional Kubernetes scaling mechanisms.

Results recorded on the SoftRequest LTD testbed:

– average response time was reduced by 28% compared to HPA;

– p95 response time was reduced by 31% under peak load conditions.

In real-world production at SoftRequest LTD:

– average response time decreased by 30–35% during typical business workloads;

– the proportion of requests processed under 200 ms increased from 92% to 97%, significantly improving the end-user experience.

Thus, proactive workload forecasting enabled minimizing response latency through advance resource scaling.

SLA Violations. Service Level Agreement (SLA) violations are a critical indicator of the quality of cloud infrastructure services. Even short-term SLA breaches can result in penalties and reputational damage.

During the experiments:

– with traditional autoscaling mechanisms (HPA/VPA), the SLA violation rate was approximately 6.2%;

– after deploying the intelligent orchestrator, the violation rate dropped to 2.7%.

These findings are summarized in Table 1.

The reduction of SLA violations by more than half demonstrates the high predictive capabilities of the developed models and the system's ability to react before service degradation occurs.

Table 1

Comparison of SLA Violation Rates

Environment	SLA Violation Rate (Baseline)	SLA Violation Rate (Orchestrator)
Laboratory Testbed	6.5%	2.8%
Production Environment	6.0%	2.6%

Cost Efficiency. Economic efficiency is an essential factor when considering the adoption of new cloud technologies.

During a 30-day observation period:

– infrastructure costs were reduced by 10–15% compared to baseline autoscaling mechanisms.

Main cost-saving factors included:

–prevention of unnecessary resource over-provisioning during short-term peaks;

–automatic utilization of cheaper spot instances during periods of low activity;

rapid deallocation of underutilized resources after load reductions, minimizing charges for idle capacity.

Thus, the implementation of the intelligent orchestrator resulted not only in improved performance metrics but also in significant operational cost optimization.

Anomaly Detection and Reaction. In dynamic multicloud environments, the ability to quickly detect and resolve anomalies is critical to ensuring business continuity.

The anomaly detection module achieved the following:

– average anomaly detection time was 5–7 seconds, more than twice as fast compared to traditional threshold-based monitoring tools;

– response time – the interval between detection and corrective action initiation – remained below 15 seconds.

Table 2 provides examples of typical anomalies detected and the corresponding mitigation actions.

Table 2

Examples of Detected Anomalies and Response Times

Anomaly Type	Detection Time (s)	Response Time (s)	Mitigation Measures
CPU Saturation	5	12	Pod rescheduling
Storage Latency Spike	6	13	Volume migration
DDoS Attack Simulation	7	14	Temporary scaling action

By identifying potential failures at early stages, the system demonstrated high resilience even under heavy and unpredictable workloads.

Key Architectural Advantages. The comprehensive architecture of the developed solution achieved a synergistic effect through:

– integration of forecasting models for advance resource scaling;

– use of reinforcement learning agents to develop optimal resource management strategies under multi-criteria constraints;

– deployment of real-time anomaly detection mechanisms to ensure early failure warning and incident mitigation.

The modular structure of the architecture enabled flexible adaptation of the system to different cloud platforms and existing DevOps workflows without requiring substantial changes to the customer's infrastructure.

Summary of Quantitative Results. The summary of experimental results confirms:

– reduction of average response time by up to 35%;

– decrease in SLA violations by more than 50%;

– reduction of infrastructure operational costs by 10–15%;

– shortening of anomaly reaction time by more than 40% compared to conventional monitoring solutions.

Thus, the developed intelligent orchestrator demonstrated its high effectiveness for deployment in dynamic, heterogeneous, and multicloud environments requiring flexibility, scalability, and high resilience.

Conclusions and prospects for further research. Within the framework of this study, an intelligent orchestrator for cloud resource management was developed, implemented, and experimentally validated, based on the integration of workload forecasting, reinforcement learning, and anomaly detection methods.

The experiments conducted both on the controlled laboratory testbed and in the real-world production hybrid infrastructure of SoftRequest LTD confirmed the high effectiveness of the proposed solution. The key achieved results include:

– a reduction of average service response time by up to 35%;

– a decrease in SLA violation rate by more than 50%;

– a reduction of infrastructure operational costs by 10–15%;

– an improvement in anomaly detection and reaction time by over 40% compared to traditional methods.

The developed architecture demonstrated the ability to efficiently adapt to changing operating conditions, maintained system resilience during load surges, and ensured cost-effective resource utilization without compromising service quality.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

The practical value of the proposed approach lies in its ability to integrate directly into existing orchestration platforms, such as Kubernetes, without requiring a major overhaul of the infrastructure. This enables enterprises and service providers to minimize implementation costs and significantly enhance the level of operational automation.

Despite the positive results obtained, the study opens several promising directions for future research:

– expanding the forecasting functionality by incorporating external factors (e.g., seasonal demand fluctuations, marketing events, changes in resource pricing by cloud providers);

– developing adaptive self-learning systems that enable the orchestrator to autonomously optimize its models in response to shifts in workload profiles or service architectures without human intervention;

– integrating energy consumption forecasting models to build energy-efficient scaling strategies and reduce the carbon footprint of cloud infrastructures;

– designing security-focused anomaly detection mechanisms aimed at real-time identification of cyberattacks or data leakage attempts;

– applying graph neural networks (GNNs) for more accurate analysis of network relationships between application components and for optimizing service placement based on network topology awareness;

– creating comprehensive multicloud optimization strategies that simultaneously consider cost, performance, energy efficiency, and reliability across different cloud providers.

In the future, the development of such intelligent orchestrators could become a cornerstone in building fully autonomous, self-learning cloud platforms capable of real-time adaptation to global shifts in operational conditions, business priorities, and quality of service requirements.

Thus, the results obtained not only demonstrate the scientific and practical significance of the proposed approach but also lay a strong foundation for continued research in the field of intelligent cloud systems automation.

Bibliography:

1. Arabnejad H., Barbosa J. Predictive Reinforcement Learning-Based Autoscaler for Cloud Resource Provisioning. *Journal of Grid Computing*. 2020. Vol. 18, No 4. P. 761–777. (date of access: 14.09.2025).

2. Chen H., et al. Intelligent Autoscaling for Web Applications in the Cloud via Reinforcement Learning. *IEEE Transactions on Services Computing*. 2021. Vol. 14, No 5. P. 1347–1359. (date of access: 14.09.2025).

3. Hsu C.-H., Chung Y. (Eds.). Cloud Computing and Big Data: Technologies, Applications and Security. Springer. 2021. (date of access: 14.09.2025).

4. Kunal T., Singh P., Rathor S., He H. Resource Scaling for Cloud Applications Using Deep Q-Learning. Proceedings of the 2022 International Conference on Cloud Computing and Big Data Analytics (ICCCBDA). 2022. P. 39–47. (date of access: 14.09.2025).

5. Mao M., Humphrey M. Auto-Scaling to Minimize Cost and Meet Application Deadlines in Cloud Workflows. Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis (SC). 2011. P. 1–12. DOI: https://doi.org/10.1145/2063384.2063449

6. Mao Y., Li J., Humphrey M. Cloud Auto-Scaling with Machine Learning. Proceedings of the 2018 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW). 2018. P. 108–117. (date of access: 14.09.2025).

7. Qiu T., Zhang L., Ghoneim A., Li W., Cai W. Prescience-Based Resource Scaling for Dynamic Workloads in Cloud Datacenters Using Ensemble Forecasting Techniques. *Future Generation Computer Systems*. 2019. Vol. 101. P. 1209–1221. (date of access: 14.09.2025).

8. Su X., Wen S., Su J., Wang J. Adaptive Autoscaling Mechanism Based on Deep Reinforcement Learning for Heterogeneous Cloud Services. Concurrency and Computation: *Practice and Experience*. 2022. Vol. 34, No 11. e6806. (date of access: 14.09.2025).

9. Tang Q., Narasimhan G. A Reinforcement Learning Approach to Efficient Resource Allocation in Cloud Computing. Proceedings of the 2021 IEEE International Conference on Cloud Engineering (IC2E). 2021. P. 45–54. (date of access: 14.09.2025).

10. Xu H., Li B. Dynamic Cloud Resource Management via Machine Learning. *IEEE Transactions on Parallel and Distributed Systems*. 2017. Vol. 28, No 1. P. 147–160. (date of access: 14.09.2025).

11. Yazdanov A., Fetzer C. Vertical Scaling for Cloud Applications. Proceedings of the 2014 IEEE 8th International Symposium on Service Oriented System Engineering (SOSE). 2014. P. 318–325. (date of access: 14.09.2025).

Дата надходження статті: 18.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.89:621.31
DOI https://doi.org/10.32689/maup.it.2025.3.3

Віктор БОЙКО
кандидат технічних наук, доцент, доцент кафедри кібербезпеки,
Національний університет «Одеська юридична академія»,
boyko-work@ukr.net
***ORCID:** 0000-0001-5929-657X*

Валерія СЛАТВІНСЬКА
доктор філософії в галузі «Право», асистент кафедри кібербезпеки,
Національний університет «Одеська юридична академія»,
slatvinskaya_valeriya@ukr.net
***ORCID:** 0000-0002-6082-981X*

Євгеній ПШЕНИЧНИЙ
здобувач вищої освіти,
Національний університет «Одеська юридична академія»,
psck@ukr.net
***ORCID:** 0009-0005-9534-9196*

ПРОБЛЕМА СТІЙКОСТІ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ СИСТЕМ В УМОВАХ ЕНЕРГЕТИЧНИХ ЗБОЇВ

Анотація. *Метою статті є аналіз загроз для інформаційно-комунікаційних мереж (Information-Communication Networks – ICN) через перебої в живленні та розробка проактивної системи для підвищення їхньої стійкості.*

Методологія. *Дослідження базується на аналізі статистичних даних про зростання частоти блекаутів (на 64% більше збоїв у США за 2011–2021 роки порівняно з попереднім десятиліттям) та оцінці їхніх наслідків, таких як економічні збитки (понад 400 млн євро на Піренейському півострові) та втрата даних у енергозалежній оперативній пам'яті (RAM), що призводить до пошкодження системних файлів і каскадних збоїв у хмарних дата-центрах. Традиційні методи захисту (ДБЖ, генератори) оцінено як недостатні через високу вартість, експлуатаційні витрати, деградацію обладнання та залежність від людського фактора. Запропоновано проактивну систему прогнозування ризиків, яка використовує методи машинного навчання (ARIMA, LSTM) для аналізу історичних даних енергомереж (напруга, частота), метеорологічних факторів і даних операторів енергосистем. Система обчислює інтегральний показник ризику та автоматично запускає захисні сценарії для мінімізації збитків.*

Наукова новизна. *Новизна полягає в розробці проактивної системи прогнозування ризиків для ICN на основі машинного навчання, яка передбачає потенційні блекаути, замість реактивного реагування. Інтегральний показник ризику, що враховує енергетичні, метеорологічні та операційні дані, є унікальним інструментом для автоматичного запуску захисних сценаріїв, що знижує залежність від людського фактора та дорогого обладнання. Це рішення підвищує фізичну та кіберзахищеність мереж, мінімізуючи вразливості до каскадних збоїв, що є новим у порівнянні з традиційними підходами.*

Висновки. *Зростання частоти та масштабів блекаутів вимагає переходу до проактивних рішень. Запропонована система прогнозування на основі машинного навчання забезпечує своєчасне реагування на загрози, мінімізує збитки для інформаційних, програмних і апаратних компонентів ICN, підвищує кібербезпеку та забезпечує безперервність роботи в умовах енергетичних збоїв.*

Ключові слова: *інформаційно-комунікаційні мережі, стійкість інформаційно-комунікаційних систем, енергетичні збої, блекаут, кібербезпека, проактивне управління, прогнозування ризиків, машинне навчання, критична інфраструктура.*

Viktor BOYKO, Valeriia SLATVINSKA, Yevgeny PSHENYCHNY. THE PROBLEM OF STABILITY OF INFORMATION AND COMMUNICATION SYSTEMS IN CONDITIONS OF ENERGY FAILURES

Abstract. Scientific novelty. *The novelty lies in the development of a proactive risk forecasting system for ICN based on machine learning, which predicts potential blackouts, instead of a reactive response. An integral risk indicator, considering energy, meteorological and operational data, is a unique tool for automatically launching protective scenarios, which reduces dependence on the human factor and expensive equipment. This solution increases the physical and cyber security of networks, minimizing vulnerabilities to cascading failures, which is new compared to traditional approaches.*

Purpose. *The purpose of the article is to analyze the threats to Information-Communication Networks (ICN) due to power outages and develop a proactive system to increase their resilience.*

Methodology. *The study is based on the analysis of statistical data on the increase in the frequency of blackouts (64% more outages in the USA from 2011 to 2021 compared to the previous decade) and the assessment of their consequences, such as economic losses (over 400 million euros in the Iberian Peninsula) and data loss in volatile random access memory (RAM), leading*

to corruption of system files and cascading failures in cloud data centers. Traditional protection methods (UPS, generators) are assessed as insufficient due to high cost, operating costs, equipment degradation and dependence on the human factor. A proactive risk forecasting system is proposed that uses machine learning methods (ARIMA, LSTM) to analyze historical power grid data (voltage, frequency), meteorological factors, and power system operator data. The system calculates an integral risk indicator and automatically launches protective scenarios to minimize losses.

Conclusions. *The increase in the frequency and scale of blackouts requires a transition to proactive solutions. The proposed machine learning-based forecasting system provides a timely response to threats, minimizes damage to information, software and hardware components of ICN, increases cybersecurity and ensures continuity of work in conditions of energy failures.*

Key words: *information and communication networks, information and communication system resilience, power outages, blackouts, cybersecurity, proactive management, risk prediction, machine learning, critical infrastructure.*

Постановка проблеми в загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями. У сучасних умовах інформаційно-комунікаційні мережі (Information-Communication Networks – ICN) є основою критичної інфраструктури, що забезпечує функціонування економіки, державного управління та суспільного життя. Їх безперебійна робота – ключовий фактор сталого розвитку. Однак, попри прогрес у галузі технологій, ICN залишаються дуже вразливими до зовнішніх впливів, зокрема, до великомасштабних збоїв в електропостачанні, відомих як блекаути (blackouts) [15], [4]. Ці інциденти демонструють тривожну тенденцію до зростання частоти та масштабів, що підтверджується не лише емпіричними спостереженнями, а й статистичними даними. Наприклад, згідно з розрахунками аналітичного центру Climate Central [7], у Сполучених Штатах за період 2011–2021 років сталося на 64% більше великих збоїв в електромережах, ніж за попереднє десятиліття (2000–2010). Подібна динаміка спостерігається і в інших регіонах світу, включно з Європою та Азією, де зростання споживання та старіння інфраструктури створюють підвищені ризики.

Руйнівні наслідки блекаутів виходять далеко за рамки короткочасної незручності. Прикладом може слугувати інцидент, що стався 28 квітня 2025 року на Піренейському півострові [19]. Цей збій, що охопив Іспанію, Португалію, Андорру, частину Франції та Марокко, призвів не тільки до колапсу транспортної системи та зупинки метрополітену у великих містах, таких як Лісабон і Мадрид, а й до трагічних людських жертв – щонайменше одна людина загинула в Португалії, а вісім – в Іспанії внаслідок пожеж, спричинених спробами аварійного освітлення. Економічні збитки від цього інциденту, за різними оцінками, перевищили 400 мільйонів євро. Ці збитки складалися з безлічі факторів: втрати прибутку підприємств через простій, збитків від псування продукції, порушень логістичних ланцюжків і витрат на відновлення пошкодженого обладнання. Основні причини таких масштабних збоїв, як правило, мають комплексний характер, включно зі зносом інфраструктури, зростанням енергоспоживання, низькою автоматизацією управління та людським фактором [4].

Аналіз останніх досліджень і публікацій. Проблема стійкості інформаційно-комунікаційних систем (ICN) в умовах енергетичних збоїв стає дедалі актуальнішою через зростання частоти блекаутів, спричинених кліматичними змінами, старінням інфраструктури та залежністю від відновлювальних джерел енергії, що впливає на роботу дата-центрів, телекомунікаційних мереж і критичних сервісів. У роботах останніх років значна увага приділяється аналізу тенденцій збоїв і розробці проактивних рішень для підвищення живучості ICN [15, с. 1–8; 20, с. 428–431]. Так, P. Hines, J. Apt та S. Talukdar досліджують історичні дані про великі блекаути в США, вказуючи на їх стабільну частоту та power-law розподіл розмірів [15, с. 1–8], тоді як Y.-K. Wu, S. M. Chang та Y.-L. Hu підкреслюють комплексні причини збоїв, включаючи перевантаження мереж і каскадні відмови [20, с. 428–431]. Звіт ASCE 2021 року оцінює енергетичну інфраструктуру США на D+, зазначаючи зростання попиту від дата-центрів і вразливість до погодних факторів [4], а Climate Central фіксує 78% зростання погодних збоїв за 2011–2021 рр. [7]. B. A. Carreras та ін. аналізують ризики блекаутів при високому проникненні ВДЕ, акцентуючи на флуктуаціях і каскадних ефектах [5, с. 132663–132674], а E. L. Ratnam та ін. пропонують диверсифікацію для підвищення стійкості до кліматичних і кіберзагроз [18]. Конкретні інциденти, як-от блекаут на Піренейському півострові 2025 р. [19], збої Google Cloud 2023 р. через відмови ДБЖ [2; 9; 13; 14] та проблеми в дата-центрі 2025 р. [17], ілюструють вразливість ICN до перебоїв живлення. В. Бойко, М. Василенко та В. Слатвінська розробили моделі живучості ICN з використанням графового підходу для симуляції відновлення після збоїв [1, с. 13–19], а M.-G. Florin та ін. класифікують ризики енергетичних криз за матрицею ймовірність-вплив [11]. Застосування машинного навчання для прогнозування збоїв досліджується в роботах U. Farooq та R. B. Bass [10, с. 61494–61519], A. K. Onaolapo та ін., які пропонують колаборативні нейронні мережі для передбачення збоїв [16, с. 3079–3087], а також у дослідженнях Aalto University [3] та B. Ghasemkhani та ін., де розроблено моделі для оцінки тривалості збоїв [12].

Метою даної статті є обґрунтування та розробка концепції системи проактивного прогнозування ризиків для забезпечення стійкості інформаційно-комунікаційних мереж (ICN) в умовах енергетичних

збоїв. Для цього зроблено наступні кроки: аналіз існуючих загроз, спричинених блекаутами, для ICN; критичний огляд обмежень традиційних методів захисту; обґрунтування необхідності переходу від реактивних до проактивних методів реагування; опис методології та архітектури системи, що використовує методи машинного навчання для прогнозування інцидентів; деталізація етапів впровадження та експлуатації запропонованої системи.

Виклад основного матеріалу. *Загрози для інформаційно-комунікаційних мереж: Технічні та організаційні аспекти.* Припинення електропостачання є критичною загрозою для ICN з кількох ключових причин. Найсуттєвішими є: втрата даних, що зберігаються в енергозалежній оперативній пам'яті (ОЗП), і порушення зв'язності мережі, особливо в централізованих ієрархічних системах.

• Втрата даних в оперативній пам'яті

Сучасні комп'ютерні системи, від персональних комп'ютерів до потужних серверів, побудовані на архітектурі фон Неймана. У її основі лежить принцип зберігання виконуваного програмного коду та оброблюваних даних в одній і тій самій пам'яті – оперативній пам'яті (RAM). RAM є енергозалежною, тобто її вміст повністю стирається при втраті електроживлення. Це зумовлено її фізичною природою – RAM використовує тригери та конденсатори, які вимагають постійного електричного заряду для підтримання стану «1» або «0». З одного боку, це дозволяє RAM досягати надзвичайно високої швидкодії, що набагато перевищує швидкість доступу до даних на енергонезалежних накопичувачах, таких як жорсткі диски (HDD) або твердотільні накопичувачі (SSD). З іншого боку, раптове відключення живлення, оминаючи штатні процедури завершення роботи операційної системи, призводить до миттєвої втрати всіх даних, які знаходилися в RAM у момент збою. В результаті можуть бути втрачені не тільки незбережені дані користувача, а й критично важливі системні файли, що потенційно веде до пошкодження операційної системи та необхідності її повної перевстановлення.

• Каскадні збої в централізованих системах

Розвиток ICN тяжіє до централізації та ієрархізації [1]. Поява потужних обчислювальних систем і дата-центрів призвела до широкого впровадження технологій віртуалізації та контейнеризації. Ці технології дозволяють оптимально використовувати ресурси потужних комп'ютерних кластерів: обчислювальні потужності можна «розділяти» між користувачами, створювати системи, які автоматично регулюють витрату ресурсів, та оптимізувати розгортання програмного забезпечення за допомогою заздалегідь створених образів операційних систем і контейнерів.

Однак, з точки зору стійкості, це створює додаткові ризики. У таких системах в RAM зберігаються не тільки дані операційної системи та користувацьких процесів, а й частини систем забезпечення віртуалізації та контейнеризації (гіпервізори, частини систем контейнеризації). Раптове відключення живлення може призвести до каскадного збою, коли відмова одного централізованого вузла тягне за собою втрату зв'язності та функціональності в масштабах усієї мережі [8]. Прикладом є збій у зоні us-east5-c дата-центру Google Cloud [2], спричинений відмовою джерела безперебійного живлення (ДБЖ) [9]. Цей інцидент призвів до порушення роботи понад двадцяти різних сервісів, зачепивши тисячі користувачів по всьому світу [13]. Інший подібний випадок – збій 25 квітня 2023 року в зоні europe-west9-a, де витік води та подальша пожежа призвели до відключень і вимагали понад доби на відновлення [14].

• Проблеми кібербезпеки під час енергетичних збоїв

Проблема енергетичних збоїв прямо впливає на кібербезпеку, створюючи нові вектори загроз, які виходять за межі простої відсутності зв'язку [20]. Раптове відключення живлення може призвести до незапланованого і некоректного завершення роботи систем, що порушує цілісність даних і конфігурацію програмного забезпечення. Під час блекауту критично важливі служби, які забезпечують кібербезпеку – системи виявлення вторгнень (IDS), системи протидії вторгненням (IPS), міжмережеві екрани (firewalls), системи моніторингу та логування – можуть вийти з ладу як і все інше програмне забезпечення. Оскільки зловмисники можуть скористатися ситуацією, щоб проникнути в систему, залишити шкідливе програмне забезпечення або скомпрометувати дані, це значно підвищує ризик інцидентів. Наприклад, якщо під час збою відключається система резервного копіювання, це може призвести не тільки до втрати даних, але й до потенційного збою в роботі системи відновлення [5].

Також блекаути можуть призвести до зниження контролю та втрати управління мережею [18]. У ситуації блекауту системи дистанційного моніторингу та управління можуть бути недоступними, що робить неможливим оперативне реагування на будь-які кіберінциденти [11]. Персонал може бути позбавлений можливості вчасно реагувати на попередження, виявлені в системах моніторингу, які ще продовжують працювати від ДБЖ або додаткових джерел живлення. Це може призвести до затримки в реагуванні на кібератаки і, як наслідок, збільшення збитків. Крім того, відновлення після збою може бути пов'язане з високим ризиком. Коли системи перезавантажуються після тривалої відсутності живлення, вони можуть бути вразливі до атак «нульового дня» або інших загроз, які зловмисники могли

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

запустити під час збою. Наприклад, якщо зловмисники фізично отримали доступ до обладнання під час блекауту, вони можуть встановити шкідливе програмне забезпечення, яке спрацює під час відновлення живлення.

Тому перехід до проактивного управління енергетичною стійкістю, що включає прогнозування ризиків за допомогою машинного навчання є необхідним елементом комплексної стратегії кібербезпеки. Такий підхід дозволяє не просто реагувати на збій, а запобігти його негативним наслідкам, захищаючи цілісність даних і безперервність роботи критичних сервісів ще до того, як інцидент набуде критичного характеру.

Аналіз існуючих рішень та їх обмежень. Традиційним і найбільш прямолінійним шляхом вирішення проблеми енергетичних збоїв є резервування – часто багаторазове – всіх існуючих систем енергоживлення та автоматизація переходу на аварійне електропостачання. Зазвичай вибудовується двоетапна система:

- Джерело безперебійного живлення (ДБЖ): Має вбудований акумулятор і розраховане на підтримання функціонування обладнання в проміжку від кількох хвилин до кількох годин. Це дозволяє коректно завершити роботу обладнання або переключитися на резервне джерело.
- Дизельний або газовий генератор: Паралельно з ДБЖ запускається додаткове джерело живлення, яке забезпечує автономне електроживлення протягом тривалого часу, до усунення основної проблеми.

Однак такий підхід, попри його очевидні переваги, пов'язаний з низкою суттєвих обмежень. По-перше, він пов'язаний з великими капітальними та експлуатаційними витратами на підтримання резервної інфраструктури. По-друге, його ефективність часто «впирається» в людський фактор. Автори неодноразово стикалися з ситуацією, коли за наявності генератора персонал не міг його запустити вчасно через помилки в процедурі, що призводило до втрати живлення – і пов'язаної з цим втрати даних і порушення робочих процесів.

Вартість повного резервування є значним бар'єром. Вона включає не тільки капітальні витрати на придбання обладнання, а й регулярні експлуатаційні витрати на паливо, технічне обслуговування та заміну компонентів. Акумулятори ДБЖ, наприклад, мають обмежений термін служби (зазвичай 3–5 років) і вимагають дорогої заміни. Їх ємність знижується з кожним циклом розряду-заряду, а також під впливом високих температур, що робить їх все менш надійними з часом. Ця природна деградація обладнання перетворює стаціонарні системи резервування на «одноразові» рішення, якщо не приділяти належної уваги їх своєчасному оновленню та обслуговуванню.

Додатковим ускладнювальним фактором є чутливість систем резервування до технічного обслуговування. Наприклад, експлуатація дизельних генераторів має тимчасові обмеження, після яких їх необхідно відключати для охолодження та/або профілактики. Самі генератори вимагають регулярного змащення, заміни фільтрів і контролю рівня палива. ДБЖ, своєю чергою, функціонують від акумуляторів, що перезаряджаються, які мають обмежену кількість циклів перезарядки, після чого їх ємність зменшується. На ємність і швидкість розряду акумуляторів може впливати температура навколишнього середовища, режим і швидкість перезарядки.

Таким чином, навіть система з «буферними» ДБЖ і довгостроковими резервними генераторами є складною системою, яка з часом може деградувати, особливо за відсутності належного обслуговування і регулярного тестування. Характерним прикладом є «одноразовий» ДБЖ, в якому не передбачено зміну акумулятора, що робить його ненадійним у довгостроковій перспективі.

Можливості прогнозування інцидентів. Інциденти з перебоями енергопостачання можна умовно розділити на дві категорії: прогнозовані та непрогнозовані. Якщо запобігти шкоді від раптових подій (наприклад, землетрусу) складно, то прогнозовані події, за умови адекватної та своєчасної реакції на них, дозволяють мінімізувати або повністю запобігти збиткам. Можливості прогнозування в сучасних умовах досить широкі, що пов'язано з характером функціонування енергосистеми.

Енергетичні мережі є складними та розподіленими системами. Суттєву роль у їх стійкості відіграє рівномірність і співмірність навантаження з генеруючими потужностями. Забезпечення цього балансу визначає стабільність мережі. Чим більша енергосистема (за умови її керованості), тим вища її живучість, оскільки великий розмір дозволяє в разі несприятливих впливів перерозподіляти потужності між різними ділянками, вирівнюючи навантаження. Крім того, важливим елементом є системи накопичення-віддачі енергії, такі як гідроакумулювальні електростанції (ГАЕС), які можуть змінювати режим роботи, виробляючи більше або менше енергії залежно від потреб.

Аналіз інцидентів показує, що настанню блекауту часто передують зміни в стані енергомережі та її параметрів (напруга, частота) [6], [19], [17]. Як правило, блекаут є наслідком проблем на якійсь ділянці, які намагаються скомпенсувати шляхом «маневрування» енергосистемою – зміною режиму роботи

генерації та перерозподілом потоків енергії. Такі дії можуть як увінчатися успіхом, так і призвести до більш глобального відключення, якщо ресурсів для «маневру» не вистачить. Важливо, що подібні дії впливають на стан енергомережі та можуть бути відстежені як зміни в стабільних до цього моменту параметрах [10], [17]. Точність прогнозування можна суттєво підвищити, враховуючи додаткові параметри [5], [16]:

- Метеорологічні дані: Погодні явища, такі як сильний вітер, грози, обмерзання, снігопади або спека, є частими причинами пошкоджень ліній електропередач (ЛЕП) і обладнання.
- Сезонні та тимчасові дані: Час року та доби, що впливають на загальний рівень споживання та виробництво електроенергії (наприклад, піки навантаження в літню спеку через кондиціонери або в зимові морози через опалення).
- Технічні дані: Дані від датчиків і систем моніторингу, які можуть вказувати на перевантаження обладнання або аномальні режими роботи.

Система проактивного прогнозування ризиків. З урахуванням усіх перерахованих обмежень, пропонується впровадження додаткової міри безпеки – *системи проактивного прогнозування ризиків*. Цей інструмент буде відстежувати стан мережі енергопостачання та аналізувати вторинні параметри, такі як загальне навантаження на мережу в межах контрольованої інфраструктури, зміни в стані енергосистеми, вхідні повідомлення від служб оповіщення і навіть прогнози погоди [3]. В основі пропонованого рішення лежить концепція раннього попередження, реалізована за допомогою методів машинного навчання [12]. Замість того щоб пасивно реагувати на збій, система активно аналізує безліч параметрів, передбачаючи ймовірність його настання. Для цього можуть бути використані прогностичні моделі на основі часових рядів, такі як *ARIMA* (Autoregressive Integrated Moving Average) або *LSTM* (Long Short-Term Memory). Ці моделі навчаються на історичних даних про стан енергомережі (напруга, частота, навантаження) і здатні виявляти аномалії, що передують великим збоям.

Як додаткові параметри можуть бути використані:

- Дані від метеорологічних служб: Інформація про наближення штормів, сильних вітрів та обмерзання, які можуть пошкодити ЛЕП.
- Інформація від операторів енергосистем: Оповіщення про планові або позапланові роботи, аварії на підстанціях.

На основі аналізу цих даних, система обчислює інтегральний показник ризику. Чим вищий показник, тим вища ймовірність збою.

Система може функціонувати як система підтримки прийняття рішень (СППР) для персоналу, відповідального за забезпечення функціонування енергосистеми. Така система може бути реалізована у вигляді поєднання централізованого дашборду з інформацією для оператора та API для інтеграції з існуючою інфраструктурою.

Вона могла б видавати попередження про можливі відключення, дозволяючи оператору прийняти своєчасне рішення. Однак зважаючи на швидкоплинність подібних інцидентів, видається корисним, щоб така система працювала в автоматичному режимі. У такому режимі вона спочатку вживає необхідних заходів щодо недопущення втрати даних, а потім інформує оператора.

Ключовою особливістю системи є її здатність до «м'якого управління», що дозволяє уникнути помилкових спрацьовувань і невиправданого відключення обладнання. Залежно від величини ризику, система може переводити функціонування системи на кілька різних рівнів. Наприклад:

- Рівень 1 (Низький ризик): Відправка повідомлення адміністратору.
- Рівень 2 (Середній ризик): Підготовка резервних систем до роботи (наприклад, прогрів дизельного генератора, перевірка статусу ДБЖ).
- Рівень 3 (Високий ризик): Автоматичне виконання захисних сценаріїв, таких як збереження відкритих даних.
- Рівень 4 (Аварійне реагування): Коректне завершення роботи некритичних сервісів і переведення ключових систем на резервне живлення.

Це дозволяє мінімізувати збитки, не вдаючись до радикальних заходів при кожному незначному коливанні мережі.

Слід враховувати, що в такому режимі можливі помилкові спрацьовування та помилки в прогнозуванні, тому оптимальним буде «м'яке управління», що мінімізує можливий збиток від таких помилок. Система може приймати рішення про переведення обладнання в режим підвищеного ризику, про підготовку систем резервування тощо, залежно від розрахованих показників ризику. Це дозволяє уникнути непотрібних відключень і зберегти безперервність роботи при незначних коливаннях.

Система прогнозування: Методологія та реалізація. Сам по собі комп'ютер, як пристрій, не має спеціальних сенсорів для вимірювання параметрів енергомережі. Внутрішні сенсори вимірюють

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

напругу та інші параметри роботи мікропроцесора та іншого апаратного забезпечення, які знаходяться «позаду» блока живлення. Блок живлення компенсує і згладжує коливання напруги та інших параметрів енергоживлення і таким чином нівелює можливість спостережень. Таким чином, до недавнього часу відстеження основних параметрів вимагало дорогої вимірювальної апаратури. Однак блекаути спричинили широке поширення та еволюцію систем безперебійної напруги. Наразі більшість професійних джерел безперебійного живлення забезпечені інформаційними інтерфейсами (мережеві карти, USB, послідовні порти), підключившись до яких, можна отримати показання параметрів мережі, таких як напруга, частота, навантаження та стан акумулятора. Регулярний збір цієї інформації дозволяє сформувати велику базу даних, яка і послужить основою для роботи системи прогнозування.

Пропонована нами система захисту використовує наявну інфраструктуру для відстеження основних параметрів і доступ до інтернету для моніторингу вторинних. Для прогнозування використовуються методи машинного навчання, які на основі аналізу отриманої інформації обчислюють значення величини ризику. Залежно від величини ризику, система приймає рішення про режим роботи. Слід зазначити, що в різних випадках можуть мати місце різні набори режимів роботи та сценаріїв реагування. Також розумно надати користувачеві можливість визначати свої власні сценарії на основі вже заданих. Така система в разі експлуатації POSIX-сумісних веб-серверів може спиратися на використання частково реалізованих сценаріїв (наприклад, unit-ів у рамках systemd підходу). Якщо розглянута або контрольована система є розподіленою, слід забезпечити централізований дашборд, можливість управління і включення-виключення робочих станцій і дистанційне управління ними. Пропонована система швидкого реагування при розгортанні буде деякий час збирати інформацію в пасивному режимі, формуючи інформаційно-часовий ландшафт поведінки енергомережі та зіставляючи його з вторинними параметрами.

Пропонована система прогнозування енергетичних збоїв складається з трьох основних частин: блоку збору даних (Data Collection Unit), блоку обробки та аналізу даних (Processing and Analytics Unit), блоку проактивного реагування (Proactive Response Unit).

Блок збору даних – це «сенсорний» рівень, який відповідає за збір інформації з різних джерел. До нього входять модулі-агенти та мережеві інтерфейси/API-шлюзи. Модулі-агенти являють собою програмне забезпечення, встановлене на пристроях або серверах, під'єднаних до ДБЖ. Вони в реальному часі збирають дані про напругу, частоту, навантаження і стан акумуляторів. Мережеві інтерфейси/API-шлюзи забезпечують підключення до зовнішніх джерел, таких як метеорологічні сервіси, API операторів енергомереж і новинні стрічки.

Зібрані дані перетворюються та аналізуються для прогнозування ризиків за допомогою блоку обробки та аналізу даних. Цей блок функціонально поділяється на базу даних, модуль обробки даних і модуль прогнозування. База даних використовується для зберігання історичної та поточної інформації про стан енергосистеми. Структура оптимізована для швидкого аналізу даних часових рядів. Як така база може використовуватися DuckDB.

Модуль обробки даних очищає, нормалізує та агрегує дані, готуючи їх для моделі машинного навчання. Модуль прогнозування (Prediction Engine) являє собою ядро системи, де розгорнуто модель машинного навчання (наприклад, LSTM або ARIMA). Вона аналізує дані та обчислює інтегральний показник ризику в реальному часі.

Блок проактивного реагування складається з модулів прийняття рішень, модуля автоматичного реагування та інтерфейсу оператора (dashboard). Він відповідає за прийняття рішень і взаємодію з оператором або іншими системами. Модуль прийняття рішень на основі показника ризику визначає необхідний рівень реагування (низький, середній, високий). Модуль автоматичного реагування виконує заздалегідь задані сценарії (скрипти для коректного завершення роботи, перемикання на резервне живлення) залежно від рівня ризику. Візуальний дашборд відображає поточний стан системи, рівень ризику, історію збоїв і прогнози, слугуючи основним інструментом для оператора.

Розгортання системи проактивного реагування. Процедуру впровадження та розгортання такої системи можна розділити на кілька ключових етапів, що забезпечують поетапне та контрольоване впровадження. Послідовність має включати в себе підготовчий етап, етап розгортання, етап навчання і тестування, етап експлуатації та моніторингу – основний етап роботи системи.

На підготовчому етапі проводиться аналіз і планування, необхідні для успішного старту проєкту. Зокрема, етап включає в себе визначення цілей і вимог проєкту. Слід визначити, які саме ICN будуть захищені, і які критичні сервіси необхідно захистити насамперед. Далі визначаються допустимі ризики та цільові показники стійкості. Потім розробляються сценарії реагування для кожного рівня ризику (наприклад, що буде відбуватися при «низькому», «середньому» і «високому» ризику).

Після цього проводиться аналіз існуючої інфраструктури: проводиться аудит ДБЖ, генераторів та іншого обладнання. У процесі слід визначити, що ДБЖ мають необхідні інформаційні інтерфейси (USB, мережеві порти) для збору даних і передбачити їх заміну або встановлення додаткових датчиків у разі відсутності таких. Також цей аналіз має включати оцінку можливостей існуючої мережі для передачі даних і віддаленого управління.

Паралельно може виконуватися збір і підготовка первинних і вторинних даних – розгортається програмне забезпечення для збору даних з ДБЖ (наприклад, за допомогою протоколу SNMP), налаштовується підключення до метеорологічних служб, API операторів енергомереж та інших зовнішніх джерел даних. Визначається формат зберігання даних і планується структура бази даних для їх зберігання та обробки.

Далі розробляється та уточнюється архітектура системи: модулі збору даних, модуль прогнозування (модель машинного навчання), модуль прийняття рішень та інтерфейс для оператора (дашборд) тощо.

На етапі розгортання виконується безпосереднє встановлення та налаштування компонентів системи. За необхідності, якщо наявні ДБЖ не підходять, проводиться їх заміна на моделі з можливістю віддаленого моніторингу. Якщо потрібно (і є така можливість у проєкті), встановлюються додаткові датчики.

Далі виконується розгортання програмного забезпечення: встановлюються та налаштовуються модулі збору даних на серверах або виділених пристроях, перевіряється підключення до ДБЖ, встановлюється модуль машинного навчання на обчислювальному сервері або в хмарі, створюється та налаштовується база даних.

Далі настає етап навчання і тестування. Система працює в пасивному режимі, щоб зібрати дані та налаштувати модель. У процесі функціонування відбувається пасивний збір даних: відбувається безперервний збір даних про стан енергомережі та зовнішні параметри. Цей період має тривати достатньо довго (кілька тижнів або місяців), щоб накопичити репрезентативний обсяг даних, що відображає нормальні та аномальні режими роботи.

Далі в рамках цього етапу виконується навчання та валідація моделі: накопичені дані використовуються для навчання моделі машинного навчання. Виявляються закономірності, що передують збоям. Для оцінки точності та зниження помилкових спрацьовувань проводиться валідація та доналаштування моделі.

Після або паралельно з навчанням моделі проводиться розробка та налаштування сценаріїв реагування. На основі аналізу даних і поведінки моделі оптимізуються пороги спрацьовування для кожного рівня ризику. Тестуються сценарії реагування (скрипти для коректного завершення роботи, перемикання на резерв, відправлення повідомлень тощо). Також виконується навчання персоналу відповідального за ICN, роботі з новими інтерфейсами, розробляються схеми та інструкції з реагування на різні рівні ризику.

Після проходження трьох попередніх етапів, система починає працювати в режимі реального часу. Протягом цього етапу безперервно виконується моніторинг і аналіз роботи системи, регулярно проводиться рекалібрування та донавчання моделі для адаптації до нових умов. Також у процесі роботи системи регулярно формуються звіти про роботу, про ефективність системи та запобігання інцидентам.

Загалом, поетапне впровадження, починаючи з пасивного збору даних і навчання, дозволяє побудувати надійну та ефективну систему, яка мінімізує ризики та підвищує стійкість ICN.

Висновки. Збільшення частоти та масштабів блекаутів вимагає додаткових заходів для забезпечення стійкості ICN. Перебої з напругою можуть призводити не тільки до первинних наслідків, пов'язаних із простоєм ICN, а й до втрати цілісності збереженої інформації та, в деяких випадках, до псування обладнання. Рішення у вигляді додаткового дублювання систем електроживлення часто є недостатнім і економічно недоцільним, оскільки системи дублювання розраховані на порівняно недовгі терміни експлуатації і при тривалій і напруженій експлуатації самі починають виходити з ладу.

Тому, незалежно від використовуваних заходів, пропонується використання додаткової системи управління ICN, яка виконуватиме прогнозування можливих збоїв на основі аналізу поведінки енергомережі та вторинних параметрів. На основі розрахованого ризику система обиратиме режим роботи ICN так, щоб мінімізувати можливі втрати. Така система може працювати як самостійно, так і як доповнення до вже існуючих когнітивно-імітаційних моделей відновлення ICN.

Розгортання та функціонування такої системи управління дозволить вчасно реагувати на інциденти з втратами електроживлення, що, своєю чергою, істотно знизить ризики збитку для інформаційної, програмної та апаратної частини ICN, а також підвищить живучість і стійкість експлуатації в умовах, пов'язаних з перебоями в мережах напруги.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Список використаних джерел:

1. Бойко В., Василенко М., Слатвінська В. Моделювання живучості та відновлення інформаційно-комунікаційних мереж в умовах дії кіберзагроз. *Інформаційні технології та суспільство*. 2024. № 1 (12). С. 13–19. URL: https://journals.maup.com.ua/index.php/it/article/view/3143.

2. A major Google Cloud outage was caused by uninterruptible power supplies being interrupted. URL: https://www.techradar.com/pro/a-major-google-cloud-outage-was-caused-by-uninterruptible-power-supplies-being-interrupted, 2023.

3. Aalto University. Machine learning helps to predict blackouts caused by storms. URL: https://www.aalto.fi/en/news/machine-learning-helps-to-predict-blackouts-caused-by-storms-0, 2019.

4. American Society of Civil Engineers (ASCE). 2021 Report Card for America's Infrastructure: Energy. American Society of Civil Engineers; URL: https://infrastructurereportcard.org/cat-item/energy-infrastructure/, 2021.

5. Carreras B. A., Colet P., Reynolds-Barredo J. M., Gomila D. Assessing Blackout Risk With High Penetration of Variable Renewable Energies. *IEEE Access*. 2021. Vol. 9. P. 132663–132674.

6. Carreras B. A., Newman D. E., Dobson I. North American Blackout Time Series Statistics and Implications for Blackout Risk. *IEEE Transactions on Power Systems*. 2016. Vol. 31, No. 6. P. 4406–4414.

7. Central C. Surging Weather-Related Power Outages. URL: https://www.climatecentral.org/climate-matters/surging-weather-related-power-outages, 2021.

8. Connexion France. French mobile network operator SFR hit by major outage. URL: https://www.connexionfrance.com/news/french-mobile-network-operator-sfr-hit-by-major-outage/730194, 2023.

9. Data Center Dynamics. UPS issue caused Google Cloud's March outage. URL: https://www.datacenterdynamics.com/en/news/ups-issue-caused-google-clouds-march-outage/, 2023.

10. Farooq U., Bass R. B. Frequency Event Detection and Mitigation in Power Systems: A Systematic Literature Review. *IEEE Access*. 2022. Vol. 10. P. 61494–61519.

11. Florin M.-G., Iosif M. R., Daniel F. N., Mihai S. A., Mihai P.-S., Alin C. E., Ioan S., Eugen S. G., Obretenova M. I. Assessment of Vulnerabilities and Risks That May Generate Energy Crises – Blackout. *Preprints*, 2025. URL: https://doi.org/10.20944/preprints202504.0815.v1.

12. Ghasemkhani B., Kut R. A., Yilmaz R., Birant D., Arıkök Y. A., Güzelyol T. E., Kut T. Machine Learning Model Development to Predict Power Outage Duration (POD): A Case Study for Electric Utilities. *Sensors*. 2024. Vol. 24, No. 13.

13. Google Cloud. Incident Report for us-east5-c outage on March 14, 2023: Incident Report N3Dw7nbJ7rk7qwrtwh7X. Google Cloud; URL: https://status.cloud.google.com/incidents/N3Dw7nbJ7rk7qwrtwh7X, 2023.

14. Google Cloud. Incident Report for us-east9-a on April 25, 2023: Incident Report dS9ps52MUnxQfyDGPfkY. Google Cloud; URL: https://status.cloud.google.com/incidents/dS9ps52MUnxQfyDGPfkY, 2023.

15. Hines P., Apt J., Talukdar S. Trends in the history of large blackouts in the United States. 2008. IEEE Power and Energy Society General Meeting – Conversion and Delivery of Electrical Energy in the 21st Century. 2008. P. 1–8.

16. Onaolapo A. K., Carpanen R. P., Dorrell D. G., Ojo E. E. Event-Driven Power Outage Prediction using Collaborative Neural Networks. *IEEE Transactions on Industrial Informatics*. 2023. Vol. 19, no. 3. P. 3079–3087.

17. Powerquality.blog. UPS Problem at Datacenter. URL: https://powerquality.blog/2025/03/17/ups-problem-at-datacenter/, 2025.

18. Ratnam E. L., Baldwin K. G. H., Mancarella P., Howden M., Seebeck L. Electricity system resilience in a world of increased climate change and cybersecurity risk. *The Electricity Journal*. 2020. Vol. 33, 9. 106833. URL: https://www.sciencedirect.com/science/article/pii/S1040619020301251.

19. Unipower. What Caused the Big Blackout in Spain and Portugal? URL: https://www.unipower.se/news/what-caused-the-big-blackout-in-spain-and-portugal/, 2025.

20. Wu Y.-K., Chang S. M., Hu Y.-L. Literature Review of Power System Blackouts. *Energy Procedia*. 2017. Vol. 141. P. 428–431. URL: https://www.sciencedirect.com/science/article/pii/S1876610217354619.

Дата надходження статті: 23.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК [004.94:631.527]+58.035
DOI https://doi.org/10.32689/maup.it.2025.3.4

Станіслав ВЕДМЕДЄВ
аспірант кафедри системного аналізу та обчислювальної математики,
Національний університет «Запорізька політехніка»,
vedmedev_s@ukr.net
***ORCID:** 0009-0005-9635-8879*

Еліна ТЕРЕЩЕНКО
кандидат фізико-математичних наук, доцент,
Національний університет «Запорізька політехніка»,
elina_vt@ukr.net
***ORCID:** 0000-0001-6207-8071*

ЦИФРОВА МОДЕЛЬ РОСЛИНИ СОНЯШНИКА ДЛЯ ФЕНОТИПУВАННЯ В ЗАДАЧАХ СЕЛЕКЦІЇ

Анотація. *Розробка високопродуктивних сортів є ключовим напрямом у контексті зростаючого тиску глобальних викликів, серед яких кліматичні зміни, демографічне зростання та обмеженість природних ресурсів. Для ефективної реалізації сучасних селекційних програм актуальним є удосконалення методів автоматизованого фенотипування.*

***Метою роботи** є розробка цифрової моделі рослини соняшника, яка забезпечує ефективне та точне фенотипування в контексті селекційних досліджень з виведення сортів соняшнику кондитерського напряму. Роботу виконано у співпраці з фахівцями лабораторії генетики та генетичних ресурсів Інституту олійних культур НААН України.*

***Методологічний підхід** базується на визначені характеристик фенотипування кондитерського соняшнику, які необхідні для розв'язання задач селекції, а саме фіксації морфологічних, біохімічних, фізичних, агрономічних ознак, умов вирощування. Визначені категорії, характеристик, одиниць вимірювання, типу даних та джерела даних.*

***Наукова новизна** полягає в створенні цифрової моделі рослини соняшника для фенотипування в задачах селекції соняшнику кондитерського напряму.*

***Висновки.** В роботі визначено набір даних, об'єднання яких є цифровою моделлю рослини соняшника для фенотипування в контексті селекційних досліджень. Для цього створено перелік характеристик морфологічних, біохімічних, фізичних, зовньошнього середовища та агротехнологій, вказано джерело даних цієї інформації. Це забезпечує передумови для розробки та застосування стандартизованих методів збору даних та уніфікованих алгоритмів обробки великих масивів даних в селекційних дослідженнях.*

***Ключові слова:** цифрова модель, фенотипування, онтологія.*

Stanislav VEDMEDEV, Elina TERESCHENKO. DIGITAL MODEL OF SUNFLOWER PLANT FOR PHENOTYPING IN BREEDING TASKS

Abstract. *The development of high-yielding crop varieties is a critical objective in the context of increasing global pressures, including climate change, population growth, and limited natural resources. To ensure the effectiveness of modern breeding programs, the advancement of automated phenotyping methods is essential.*

***This study aims** to develop a digital model of the sunflower plant that facilitates efficient and precise phenotyping for use in breeding programs targeting confectionery sunflower varieties. The research was conducted in collaboration with specialists from the Laboratory of Genetics and Genetic Resources of the Institute of Oilseed Crops, NAAS of Ukraine.*

***The methodological approach** is based on identifying the key phenotypic traits of confectionery-type sunflower required for solving breeding tasks. These traits include morphological, biochemical, physical, and agronomic characteristics, as well as environmental growing conditions. Categories, parameters, measurement units, data types, and data sources were systematically defined. The scientific novelty of this work lies in the creation of a digital model of the sunflower plant tailored specifically for phenotyping in confectionery sunflower breeding.*

***Conclusions.** The study defines a comprehensive dataset, the integration of which constitutes a digital model of the sunflower plant for use in phenotyping within breeding research. The model includes a structured list of morphological, biochemical, physical, environmental, and agrotechnological parameters, along with clearly defined data sources. This provides a foundation for the development and application of standardized data collection methods and unified algorithms for processing large-scale datasets in breeding programs.*

***Key words:** Digital model, Phenotyping, Ontology.*

Постановка проблеми. Сучасне сільське господарство функціонує в умовах зростаючого тиску глобальних викликів, серед яких кліматичні зміни, демографічне зростання та обмеженість природних ресурсів. У цьому контексті особливої актуальності набуває селекція сільськогосподарських культур як ключовий інструмент забезпечення стабільного та ефективного агровиробництва. Селекція соняшника,

як однієї з основних олійних культур світу, має стратегічне значення для продовольчої безпеки та сталого землеробства, зважаючи на його роль у формуванні сівозмін та адаптивний потенціал у різних агрокліматичних умовах. Посилення вимог до врожайності, адаптивності та біотичної стійкості сортів і гібридів зумовлює необхідність інтеграції сучасних технологічних підходів у селекційний процес.

Серед таких підходів важливе місце займає моделювання, яке забезпечує формалізоване відображення об'єктів, процесів або явищ із метою їх аналізу, прогнозування та оптимізації. У галузі біології та аграрних наук моделі використовуються для кількісного опису фізіолого-біохімічних процесів, морфологічних характеристик рослин, а також для аналізу взаємодії з абіотичними та біотичними факторами середовища. Застосування моделей у селекції соняшника дозволяє інтегрувати інформацію про генотип, фенотип та умови вирощування для підвищення точності добору, прискорення циклів селекції та формування високопродуктивного, адаптивного вихідного матеріалу. В умовах обмежених ресурсів і зростаючої варіабельності зовнішніх факторів моделювання стає важливим інструментом для підвищення ефективності прийняття рішень у селекційній практиці.

Аналіз останніх досліджень і публікацій. Одним із перспективних підходів є технологія цифрових двійників (digital twins, DT), яка передбачає створення динамічної віртуальної моделі фізичного об'єкта або процесу з постійним оновленням її параметрів на основі даних у реальному часі. Цифровий двійник рослини (digital twin of a plant) – це комп'ютерно реалізована динамічна модель, яка безперервно відображає морфофізіологічний стан реального біологічного об'єкта, підтримує двосторонній зв'язок з фізичним середовищем і оновлюється на основі даних сенсорного моніторингу та зовнішніх впливів [3; 4; 11; 22; 23; 27; 30]. За рахунок використання оптичних, спектральних, глибинних і екологічних сенсорів модель акумулює дані про ключові ознаки росту (геометрія, орієнтація листя, щільність надземної маси, стан генеративних органів) та виявляє аномальні відхилення у розвитку [4; 24]. У практиці фенотипування широко застосовуються mesh-моделі, які відтворюють зовнішню поверхню об'єкта на основі трикутної сітки, та voxel-моделі, що формують об'ємне представлення на основі регулярної тривимірної дискретизації [24; 19]. Цифровий двійник виконує функцію відтворюваної цифрової репрезентації, що може бути збережена у форматі багатошарової бази даних, масштабована, використана в симуляційних експериментах, або передана в інші дослідницькі чи виробничі середовища [22; 25]. NeRF метод використовується для реконструкції складних геометрій рослин та дрібних структур у статичному лабораторному середовищі, тому може бути корисним для селекційної роботи з рослинами при відповідної розробці умов фіксації [20, 21; 29]. Також методи 3D реконструкції на основі NeRF, згорткових нейронних мереж та 3D Гаусового розсіювання розглянуто в роботах [15; 16; 18]. Важливим напрямом є розробка моделей, що враховують розвиток рослини в часі, тобто темпоральних цифрових двійників. У роботі GrowSplat [8] запропоновано підхід на основі 3D Gaussian Splatting, що дозволяє створювати послідовності 3D-моделей для кожного моменту часу та виконувати їх вирівнювання (alignment). Це забезпечує відстеження морфологічних змін і динамічний аналіз росту.

Аналіз досліджень та публікацій демонструє активне застосування інформаційних технологій для різних завдань в агрономії, що породжує необхідність побудови різноманітних моделей під певний тип задачі. Далі розглянемо застосування поняття цифрової моделі рослини для селекції. В попередній роботі авторів [14] введено поняття цифрової моделі рослини як набору цифрових даних, що містять характеристики рослини та агротехнологій. В роботі визначено набір характеристик, важливих для селекції соняшника кондитерського, а саме насіння та кошика, що є тільки частиною необхідної інформації для фенотипування.

Метою статті є розробка цифрової моделі рослини соняшника, яка забезпечує ефективне та точне фенотипування в контексті селекційних досліджень, включаючи морфологічні, біохімічні та фізичні характеристики, вплив зовнішнього середовища та агротехнологій.

Виклад основного матеріалу дослідження. Селекція як наука і практика спрямована на створення нових сортів, ліній та гібридів культурних рослин з покращеними господарсько цінними ознаками. У контексті селекції соняшника, як однієї з провідних олійних культур, особливо актуальними є задачі підвищення врожайності, стійкості до біотичних та абіотичних чинників, а також поліпшення якісних показників насіння. Для успішного розв'язання цих задач селекціонеру необхідно, по-перше, обрати або розробити ефективну методику ідентифікації та кількісної оцінки цільових ознак, важливих у селекційному процесі. По-друге, використовуючи цю методику, провести всебічну оцінку наявного селекційного матеріалу, встановити варіабельність та рівень спадковості досліджуваних ознак. По-третє, на основі отриманих даних створити нові генотипи із бажаним проявом цільових характеристик. Основними вхідними даними в процесі є генетичне різноманіття вихідного матеріалу, умови вирощування, результати фенотипової та, за можливості, генотипової оцінки. Ключовими викликами

залишаються складність контролю багатьох кількісних ознак, вплив навколишнього середовища на їх реалізацію, а також необхідність поєднання традиційних методів селекції з сучасними молекулярно-генетичними підходами. Таким чином, ефективна селекція соняшника потребує комплексного підходу, що включає інтеграцію новітніх технологій із класичними принципами генетики та агрономії.

Фенотип, як інтегральний прояв генотипу в конкретних умовах середовища, становить основну оцінну одиницю в селекційній практиці, оскільки саме за фенотиповими параметрами здійснюється добір генотипів із підвищеною адаптивною та продуктивною цінністю. Комплекс ознак, таких як продуктивність, вміст олії, рівень резистентності до біотичних агентів та толерантність до абіотичних стресорів, визначає господарську придатність гібридів, ліній та сортів для впровадження у виробництво. Проте валідна інтерпретація фенотипічної варіабельності можлива виключно за умови точного знання агроекологічного контексту, в якому реалізується відповідний генотип, з огляду на істотний вплив факторів середовища на експресію спадкових ознак.

У зв'язку з цим постає необхідність у створенні уніфікованих інформаційно-аналітичних систем, які б інтегрували фенотипічні дані з параметрами зовнішнього середовища, зокрема, ґрунтовими характеристиками, кліматичними умовами, агротехнічними прийомами та іншими релевантними чинниками. Формування такої єдиної бази знань є критично важливим для підвищення точності селекційної оцінки, забезпечення репродуктивності результатів між дослідними платформами, сезонами та географічними зонами, а також для об'єктивного аналізу складних кількісних ознак у процесі генетичного поліпшення соняшника. Розробці онтології вирощування соняшника присвячено попередню роботу авторів [13].

У процесі генетичного поліпшення сільськогосподарських культур, зокрема соняшника (Helianthus annuus L.), перед селекціонером постає комплекс завдань, що визначають ефективність селекційного циклу. Першочерговим є розроблення або вдосконалення аналітичних методик і протоколів кількісного оцінювання ознак, які мають суттєве господарське значення. Наступним критичним етапом є застосування цих методик для репрезентативного фенотипування селекційного генофонду з метою встановлення спадкової стабільності та генетичної детермінованості відповідних ознак. Заключним кроком у селекційному процесі є формування нових генотипічних конструкцій – ліній, сортів або гібридів, які характеризуються стабільним і високим рівнем експресії цільових фенотипових ознак.

На сучасному етапі розвитку аграрної науки важливу роль у цьому процесі відіграє фенотипування, що є технологією високоточної кількісної оцінки морфологічних, фізіологічних та біохімічних ознак рослин. Традиційно оцінка складних ознак, таких як стійкість до абіотичних (засуха, висока температура) та біотичних (збудники хвороб, шкідники) факторів, здійснювалася переважно за допомогою суб'єктивних бальних шкал, які ґрунтувались на візуальних спостереженнях дослідника. У багатьох випадках така оцінка супроводжувалася умовною кількісною інтерпретацією, наприклад, у вигляді відсотка уражених рослин або ступеня ураження листкової поверхні. Водночас обмежена кількість повторень, обумовлена ресурсними факторами, додатково знижувала достовірність отриманих результатів.

У зв'язку з цим фенотипування набуло особливого поширення насамперед у дослідженнях стресостійкості, де дозволило досягти істотного прогресу завдяки більш об'єктивному, відтворюваному та кількісному підходу до оцінки ознак [9; 17]. Паралельно з розвитком фенотипових платформ активно розвиваються молекулярно-генетичні технології, зокрема методи картування геному та аналізу асоціацій генотип-ознака. Завдяки цим підходам було ідентифіковано численні локуси, пов'язані з такими ознаками, як посухостійкість. Найбільш глибокі дослідження дозволили локалізувати відповідні гени на конкретних ділянках геному [12; 28]. Однак наступним кроком у розумінні природи цих ознак є вивчення функціональної ролі відповідних генів у формуванні фенотипу, що потребує залучення фізіолого-біохімічних досліджень [26]. Такі роботи мають поодинокий характер через їх високу вартість, трудомісткість і технологічну складність як на фенотиповому, так і на молекулярному рівні.

Для створення цифрової моделі рослини, необхідно визначитися з переліком характеристик, що є важливими для селекційного відбору. При селекційному відборі соняшника важливими є ознаки насінини та кошика, а саме розмір насіння, натура, лушпинність, щільність лушпиння, обрушуваність, відсоток крупного насіння, форма, розміри, забарвлення насіння, вміст білку та олій. В роботі [14] було визначено перелік характеристик для кондитерського соняшнику за ознаками насінини та кошика, який базується на вимогах Державної методики України «Методика визначення однорідності та стабільності (ВОС)» та досліджень Інституту олійних культур НААН України [10]. Дослідниками Інституту олійних культур було розширено набір характеристик й введені нові градації для деяких з них. Запропоновано нові градації і вимір ознак у кількісному варіанті для співвідношення площ темного та світлого забарвлень та сили руйнування оболонки насінини. В роботі [14] зафіксовано такий

перелік характеристик для цифрової моделі рослини соняшнику для фенотипування: довжина насінини (мм, см), ширина насінини (мм, см), товщина насінини (мм, см), форма насінини (категоріальна), товщина відносно ширини (тонка, середня, товста), основний колір насінини (категоріальна), смуги на краю насінини (відсутні, слабкі, виражені), колір смуг (категоріальна), співвідношення площ темного та світлого забарвлень (< 1/4, 1/4–3/4, > 3/4), плямистість перикарпію (відсутня, наявна), вага насінини (г), вміст олії у насінні (%), вміст білку у насінні (%), лушпинність (%), вміст олеїнової кислоти в олії (%), вміст пальмітинової кислоти в олії (%), вміст лінолевої кислоти в олії (%), вміст ліноленової кислоти в олії (%), сила руйнування оболонки насінини (Н), положення кошику (кут у градусах або категоріальна), розмір кошику (см), форма кошику зі сторони сім'янок (категоріальна), кількість насінин у кошику (шт.), кількість спіралей насінин у кошику (шт.), кількість насінин у кожній спіралі (шт.), вага насінин у кошику (г). Визначено джерело отримання даних та метод зберігання.

Визначимо додатково характеристики, що має містити цифрова модель рослини соняшника. Для оцінки генотипу та впливу зовнішнього середовища дуже важливими є такі показники як висота рослин ї площа та колір листя. Для оцінки можливості успішного виживання рослини в складних умовах необхідно враховувати довжину головного кореня та об'єм ґрунту охоплений коренями рослини.

Далі визначимо параметри впливу зовнішнього середовища, які потребують визначення при фенотипуванні в контексті селекційних досліджень.

Рослина соняшнику знаходиться на межі впливу двох стихій, а саме ґрунту та повітря. Значна частина успіху у реалізації генотипу залежить від складу та стану ґрунту, що добре описано у роботах з ґрунтознавства, агрохімії, агрономії та інших науках [6]. Головними факторами для рослин є вміст головних поживних елементів у вигляді доступних сполук азоту, фосфору, калію, а також наявність відповідної органічної складової – гумусу та мікроелементів [2]. Однак і інші складові теж мають свій внесок і можуть при достатній кількості основних факторів виявити свій основний вплив. Основні характеристики ґрунту представлено в (табл. 1).

Таблиця 1

Характеристики ґрунту, які фіксуються для задач фенотипування соняшнику

Категорія	Характеристика	Одиниця вимірювання / Тип даних	Джерело даних
Фізичні властивості	Тип ґрунту	Класифікаційна категорія	Лабораторний аналіз / польова оцінка
	Структура ґрунту	Якісна оцінка	Візуальна оцінка / сенсори
	Щільність	г/см^3	Лабораторія / польові сенсори
	Пористість	%	Розрахунок
	Вологість	%	Вологоємні сенсори
	Температура	°C	Ґрунтові термометри
	Кам'янистість	% або кількість каміння на м^2	Візуальна оцінка / сенсори
Хімічні властивості	pH	Шкала кислотності (0–14)	Лабораторія / pH-метри
	Електропровідність	dS/m	Сенсори / лабораторія
	Вміст гумусу	%	Лабораторія
	Вміст органічної речовини	%	Лабораторія
	Азот (N)	мг/кг	Лабораторія
	Фосфор (P)	мг/кг	Лабораторія
	Калій (K)	мг/кг	Лабораторія
	Мікроелементи (Fe, Mn, Zn, B тощо)	мг/кг	Лабораторія
Біологічні властивості	Мікробіологічна активність	Кількість колоній / активність	Лабораторія
	Наявність патогенів	Виявлено / Не виявлено	Лабораторія
Агротехнічні фактори	Історія обробітку поля	Текст / запис	База агрономічних даних
	Система удобрення	Типи добрив, дози, дати внесення	Агрономічна документація
	Зрошення	Обсяг, частота	Сенсори / агродокументація
	Попередники	Назви культур	План сівозміни / документація

Основні кліматичні характеристики, які фіксуються для фенотипування рослин у селекційних дослідженнях представлено в (табл. 2). Джерелами кліматичних даних можуть бути локальні метеостанції, мобільні агрокліматичні сенсори, супутникові платформи (Copernicus, NASA POWER), дані з агрометеорологічних сервісів (Meteoblue, NOAA, Weather API). Ці впливи мають тенденції до змін, які зараз відтворюються в глобальне потепління. В умовах України збільшується посушливість клімату, що доведено Семеновою І.Г. [7].

Таблиця 2

Кліматичні характеристики, які фіксуються для фенотипування рослин у селекційних дослідженнях

Категорія	**Характеристика**	**Одиниця вимірювання / Тип даних**	**Джерело даних**
Температурні умови	Середньодобова температура	°C	Метеостанції / датчики
	Максимальна денна температура	°C	Метеостанції / логери
	Мінімальна нічна температура	°C	Метеостанції / логери
	Температурні стреси (кількість днів >30°C або <5°C)	Кількість днів	Метеорологічні архіви
	Сума активних температур за період вегетації рослин	°C	Метеостанції / логери
Опади	Загальна кількість опадів	мм	Метеостанції / дощоміри
	Інтенсивність опадів	мм/год або мм/доба	Метеостанції
	Посушливі періоди	Кількість днів без опадів	Аналіз метеоданих
	Сума опадів за період вегетації рослини	мм	Метеостанції / дощоміри
Сонячне випромінювання	Сонячна радіація (інсоляція)	МДж/м^2/день або Вт/м^2	Радіометри / супутникові дані
	Тривалість світлового дня	Години	Астрономічні розрахунки
	Кількість похмурих / сонячних днів	Днів	Супутникові знімки / архіви
Вологість повітря	Середня відносна вологість	%	Метеостанції / гігрометри
	Мінімальна / максимальна вологість	%	Метеостанції / логери
Вітер	Середня швидкість вітру	м/с	Анемометри / метеостанції
	Максимальні пориви вітру	м/с	Метеостанції
	Напрям вітру	Кут / сектор (N, NE, E, тощо)	Метеостанції
Інші показники	Атмосферний тиск	гПа або мм рт. ст.	Барометри
	Випаровуваність (Evapotranspiration, ET0)	мм/доба	Розрахунок за формулами FAO
	Кількість градобоїв / буревіїв / заморозків	Події (факт фіксації)	Метеозведення / польові спостереження
Кліматичні індекси	Сума активних температур (>10°C)	°C	Агрономічні розрахунки
	Індекс аридності (засушливості)	Безрозмірна величина	Розрахунок за метеоданими
	Кількість сприятливих днів для вегетації	Дні	Агрономічні моделі

Частину кліматичних умов і властивостей ґрунту можна цілеспрямовано модифікувати за допомогою агротехнічних прийомів. Зокрема, йдеться про здійснення додаткового зрошення, внесення мінеральних та органічних добрив, глибоке рихлення ґрунту. Біотичні фактори, обумовлені дією кліматичних чинників і навколишнього середовища – зокрема шкідники та патогенні організми – також чинять істотний вплив на розвиток рослин. Їхня шкодочинність може змінюватися під впливом

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

агротехнічних прийомів і засобів хімічного чи біологічного походження, що застосовуються в агрономічній практиці. Отже, агрономічні заходи відіграють ключову роль у реалізації потенціалу генотипу та його адаптації до конкретних кліматичних умов регіону. Тому врахування агрономічного впливу є необхідною умовою як у процесі фенотипування рослин, так і при побудові цифрових моделей їх розвитку. Дослідження, що проводяться в умовах контрольованого середовища, як правило, передбачають моделювання лише тих факторів, які створені людиною. Відтак, виникає необхідність забезпечення рослини всіма основними ресурсами: водою, світлом, теплом, макро- і мікроелементами. Включення впливу агрономічних чинників, насамперед агротехнічних прийомів (табл. 3), сприяє підвищенню точності аналізу фенотипових ознак. Інформація збираються вручну, за допомогою агро-ІТ-систем або автоматично з дронів, сенсорів чи машинної телеметрії.

Таблиця 3

Агротехнічні характеристики, які фіксуються та враховуються при фенотипуванні рослин

Категорія	Характеристика	Одиниця вимірювання / Тип даних	Джерело даних
Обробіток ґрунту	Тип основного обробітку (оранка, глибоке рихлення, mini-till, no-till)	Кваліфікатор (назва методу)	Польовий журнал, агроплан
	Глибина основного обробітку	см	Агроагрегати, агрономічна карта
	Тип передпосівного обробітку	Кваліфікатор	Польові записи
Сівба	Дата сівби	Дата	Журнал агротехнологій
	Густота стояння рослин	тис. рослин/га	Польові обміри / дрони
	Глибина загортання насіння	см	Польові виміри
	Схема сівби	міжряддя (см)	Сівалки, документація
Добрива	Вид добрив	NPK, органічні, мікроелементи	Агроплани, лабораторні аналізи
	Доза внесення	кг/га або л/га	Польові журнали
	Спосіб внесення (в основне, припосівне, листове, фертигація)	Кваліфікатор	Агрокартографія
	Кількість внесень	Кількість разів	План-графік
Зрошення / волога	Спосіб зрошення (краплинне, дощування, борозенкове)	Кваліфікатор	План зрошення
	Об'єм поданої води	мм або $м^3$/га	Системи зрошення
	Кількість поливів	Разів	Агроніма
Захист рослин	Вид ЗЗР (гербіциди, фунгіциди, інсектициди)	Назва + діюча речовина	Журнал обробок, AgroScout
	Дата та фаза внесення	Дата / фенологічна фаза	Скаутинг, агрокалендар
	Доза внесення	л/га або кг/га	Сертифікат, обприскувач
	Спосіб обробки	Наземний, авіаційний, дрон	Польові журнали
Інші агроприйоми	Мульчування	Наявність / тип	Польові спостереження
	Інокуляція / стимуляція	Препарати / методика	Журнал агрозаходів
	Суміжні культури / сівозміна	Попередник, наступна культура	Агрономічний план

Питанням зберігання та структурування даних присвячено попередні роботи авторів [13].

Висновки. Створення доступних, стандартизованих і відтворюваних інструментів фенотипування є актуальним напрямом сучасної селекції. Для ефективного використання інформаційних технологій у селекційних програмах цифрова модель повинна точно відображати морфологічні, фізіологічні та розвиткові характеристики рослин. Це можливо завдяки інтеграції даних з різних джерел таких як сенсори, комп'ютерний зір і екологічні вимірювання. Такі інструменти мають забезпечувати кількісну цифрову оцінку цільових ознак рослин з фіксацією умов зовнішнього середовища. Це дозволяє здійснювати об'єктивне порівняння між генотипами та математично обґрунтовувати виявлені відмінності. Ефективність подібних рішень можлива лише за умови дотримання стандартизованих методів збору даних та уніфікованих алгоритмів обробки. У цьому контексті запропоновано створення цифрової моделі рослини соняшнику для фенотипування в задачах селекції. Запропонований підхід не лише підвищує точність фенотипової оцінки, а й створює передумови для автоматизації обробки

великих масивів даних. Це є важливим кроком у цифровій трансформації сучасної селекційної науки. Крім того, цифрова модель рослини може стати основою для створення складніших прогнозних систем, спрямованих на оцінку стану рослин і оптимізацію агротехнологічних рішень.

Список використаних джерел:

1. Атлас морфологічних ознак сортів рослин соняшнику однорічного Helianthus annuus L. (наочне доповнення до Методики проведення кваліфікаційної експертизи на ВОС соняшнику однорічного) / Міністерство аграрної політики та продовольства України; Український інститут експертизи сортів рослин. К.: Український інститут експертизи сортів рослин, 2018. 79 с.
2. Господаренко Г. М., Черно О. Д., Мартинюк А. Т., Бойко В. П. Винесення основних елементів живлення з ґрунту культурами польової сівозміни за різного удобрення. *AgroChemistry and Soil Science*. 2021. № 91. С. 31–40.
3. Грачов О. Plant monitoring system: ШІ-система для розумного моніторингу рослин. Інформаційні технології та суспільство. 2025. Вип. 1 (16). С. 59–64. https://doi.org/10.32689/maup.it.2025.1.7
4. Зозуля О. Л., Швартау В. В., Михальська Л. М., інші. Сучасні методи цифрового моніторингу в рослинництві. frg.org.ua. URL: https://frg.org.ua
5. Кендзьора Н. З. Динаміка температури атмосферного повітря і режим опадів як фактори змін феноритміки рослин в період 2010–2019 років. *Екологія, природокористування та охорона навколишнього середовища: прикладні аспекти.* 2020. С. 51–54.
6. Примак І. Д., Купчик В. І., Лозінський М. В., Войтовик М. В., Панченко О. Б., Косолап М. П., Панченко І. А. Агрономічне ґрунтознавство. К.: 2017. 580 с.
7. Семенова І. Г. Синоптичні та кліматичні умови формування посух в Україні: монографія. Одеський державний екологічний університет. Х.: ФОП Панов А. М., 2017. 236 с.
8. Adebola T., Zhang Y., Kim J., Navlakha S. GrowSplat: Dynamic 3D Gaussian splatting for plant morphology alignment. *arXiv preprint*. 2025. arXiv:2505.10923. URL: https://arxiv.org/abs/2505.10923
9. AgEval: A Benchmark for Zero-Shot and Few-Shot Plant Stress Phenotyping with Multimodal LLMs. arXiv preprint. 2023. URL: https://arxiv.org/abs/2306.05431
10. Aliyev E. B. Development of a device for automatic phenotyping of sunflower seed material. *Machinery & Energetics. Journal of Rural Production Research*. 2019. Vol. 10, No. 1. P. 11–17. ISSN 2663-1334. (Kyiv, Ukraine).
11. Andres F., Wiechers D., Langensiepen M., Kage H. Field Robot Platform for Phenotyping Maize. *Computers and Electronics in Agriculture*. 2021. Vol. 182. Article ID: 105991. DOI: https://doi.org/10.1016/j.compag.2021.105991
12. Atkinson N. J., Lilley C. J., Urwin P. E. Genome-Wide Association Mapping of Time-Dependent Growth Responses to Moderate Drought Stress in Arabidopsis. *Plant Cell and Environment*. 2013. Vol. 36, Issue 2, pp. 262–276. DOI: https://doi.org/10.1111/j.1365-3040.2012.02598.x
13. Bakurova A., Vedmedeva K., Vedmedev S., Tereschenko E. Ontological model of *Helianthus* cultivation in Ukrainian conditions. *CEUR Workshop Proceedings*. 2023. Vol. 3396. P. 130–140.
14. Bakurova A., Vedmedeva K., Vedmedev S., Tereschenko E., Shyrokorad D. Development of the System for the Digital Model of the Helianthus Phenotype. *Proceedings of the VI International Scientific Congress Society of Ambient Intelligence* 2023 (ISC SAI 2023). 20–25 November 2023. P. 20–25. ISBN 978-80-88618-46-1. DOI: 10.46489/ISCSAI-23-30.
15. Gao Y., Qi X. Neural 3D Gaussian Splatting for Plant Structure Reconstruction. *ACM Transactions on Graphics*. 2024. Vol. 43, No. 4. P. 1–14.
16. Hu M., Zhao W., Li J. NeRF-based dynamic 3D modeling of plant growth using multi-view time series. *IEEE Transactions on Image Processing*. 2024. (in press).
17. Intelligent Monitoring of Stress Induced by Water Deficiency in Plants using Deep Learning. arXiv preprint. 2022. URL: https://arxiv.org/abs/2205.09364
18. Li Q., Wang Z., Chen L., Xu Y. Integrating 3D Gaussian Fields with NeRF for High-Fidelity Plant Reconstruction. *Sensors*. 2025. Vol. 25, No. 3. Article ID: 987. DOI: https://doi.org/10.3390/s25030987
19. Liu S., Wang H., Yan J., Chen Y. 3D Plant Phenotyping with Voxel-Based Reconstruction Techniques. *Plant Methods*. 2020. Vol. 16. Article ID: 116. DOI: https://doi.org/10.1186/s13007-020-00652-1
20. Martin-Brualla R., Radwan N., Sajjadi M. S. M., Barron J. T., Dosovitskiy A., Duckworth D. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021. P. 7210–7219.
21. Mildenhall B., Srinivasan P. P., Tancik M., Barron J. T., Ramamoorthi R., Ng R. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. *Communications of the ACM*. 2021. Vol. 65, No. 1. P. 99–106.
22. Shi Y., Zhang Y., Xu J., Li B. Digital twins in plant phenotyping: Concepts, applications, and future perspectives. *Frontiers in Plant Science*. 2022. Vol. 13. Article ID: 1038810. DOI: https://doi.org/10.3389/fpls.2022.1038810
23. Tardieu F., Cabrera-Bosquet L., Pridmore T., Bennett M. Plant Phenomics, From Sensors to Knowledge. *Current Biology*. 2017. Vol. 27, No. 15. P. R770–R783. DOI: https://doi.org/10.1016/j.cub.2017.05.055
24. Tsaftaris S. A., Minervini M., Scharr H. Machine learning for plant phenotyping needs image processing. *Trends in Plant Science*. 2016. Vol. 21, No. 12. P. 989–991. DOI: https://doi.org/10.1016/j.tplants.2016.10.002
25. Ubbens J. R., Stavness I., Francis P. The use of plant models in deep learning: An overview and roadmap. *Plant Methods*. 2018. Vol. 14. Article ID: 36. DOI: https://doi.org/10.1186/s13007-018-0312-8
26. Wang, J., et al. Transcriptome Profiles Reveal Response Mechanisms and Key Role of PsNAC1 in Pinus sylvestris var. mongolica to Drought Stress. *BMC Plant Biology*. 2022. Vol. 22, Article 72. DOI: https://doi.org/10.1186/s12870-022-03475-7

27. Yu Y., Schlüter U., Weber M. DT-FieldPheno: A digital twin approach for high-throughput field phenotyping. *Computers and Electronics in Agriculture*. 2021. Vol. 188. Article ID: 106349. DOI: https://doi.org/10.1016/j.compag.2021.106349

28. Zhang, Z., et al. Molecular Mechanisms of Drought Resistance Using Genome-Wide Association Mapping in Maize (Zea mays L.). *Frontiers in Plant Science*. 2020. Vol. 11, Article 1109. DOI: https://doi.org/10.3389/fpls.2020.01109

29. Zhu H., Huang L., Li Y. High-resolution 3D reconstruction of plants using NeRF-based models in controlled environments. *Journal of Plant Research and Imaging*. 2024. Vol. 29, No. 2. P. 145–159.

30. Ziamtsov I., Navlakha S. GrowSplat: Data-driven modeling of plant architecture using voxelized imaging. *Bioinformatics*. 2020. Vol. 36, Suppl_1. P. i125–i132. DOI: https://doi.org/10.1093/bioinformatics/btaa432

Дата надходження статті: 27.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

UDC 004.8:004.94:621.316
DOI https://doi.org/10.32689/maup.it.2025.3.5

Dmytro VOITEKH
Postgraduate Student, Institute of Computer Technologies,
Open International University of Human Development "Ukraine",
d.voitekh@gmail.com
***ORCID:** 0009-0003-8997-5495*

Anatolii TYMOSHENKO
Ph.D., Associate Professor, Institute of Computer Technologies,
Open International University of Human Development "Ukraine",
timoshag@i.ua
***ORCID:** 0000-0003-0954-3186*

IMITATION REINFORCEMENT LEARNING AND RULE-BASED EXPERTS FOR BUILDING ENERGY SYSTEMS MANAGEMENT

Abstract. *The relevance of the study is determined by the existing sample inefficiency barrier preventing reinforcement learning deployment in building energy management. Traditional RL algorithms require thousands of training episodes (equivalent to decades of simulated operation), making them impractical for safety-critical infrastructure where poor decisions risk equipment damage and grid instability.*

The aim of the paper *is to investigate how imitation learning can accelerate RL convergence through expert demonstrations from optimized rule-based controllers. The research evaluates three approaches: behavioral cloning (BC-SAC), dataset aggregation (DAgger-SAC), and imitation bootstrapped reinforcement learning (IBRL-SAC), all tested within the standardized CityLearn environment for multi-objective building control.*

Methodology *employs Bayesian-optimized rule-based controllers as expert demonstrators, evaluated across multiple building configurations using real operational data from residential buildings with photovoltaic systems and battery storage. Each variant combines expert-guided initialization with standard SAC training, tested over 365-day simulations with performance measured by cost reduction, emission minimization, and grid stability metrics.*

Results show *that BC-SAC achieves nearly 50% reduction in training requirements while maintaining superior performance, outperforming both standard SAC and optimized rule-based controllers. Imitation learning methods demonstrate competent performance from initial episodes, eliminating the risky exploration phase that prevents real-world deployment.*

Scientific novelty *lies in being the first comprehensive evaluation of imitation learning variants for CityLearn, establishing quantitative efficiency-performance trade-offs previously unexplored in standardized benchmarks. The research proves that optimized rule-based experts can effectively bootstrap RL policies, creating a practical pathway for deployment where extensive training is prohibitive.*

Key words: *machine learning, neural networks, reinforcement learning, imitation learning, behavioral cloning, DAgger, SAC, building energy management, CityLearn.*

Дмитро ВОЙТЕХ, Анатолій ТИМОШЕНКО. ІМІТАЦІЙНЕ НАВЧАННЯ З ПІДКРІПЛЕННЯМ ТА ЕКСПЕРТНІ СИСТЕМИ НА ОСНОВІ ПРАВИЛ ДЛЯ КЕРУВАННЯ ЕНЕРГОСИСТЕМАМИ БУДІВЕЛЬ

Анотація. *Актуальність дослідження обумовлюється існуючими обмеженнями ефективності методів навчання з підкріпленням у задачах керування локальними енергосистемами будівель. Традиційні алгоритми потребують тисячі навчальних епізодів (що еквівалентно десятиліттям симульованих даних), що робить їх непрактичними для критично важливої інфраструктури, де помилкові рішення загрожують пошкодженням обладнання та нестабільністю мережі.*

Мета роботи *полягає у дослідженні як імітаційне навчання може прискорити збіжність алгоритмів навчання з підкріпленням через експертні демонстрації від оптимізованих контролерів побудованих на основі правил. У дослідженні порівнюються три підходи: поведінкове клонування (BC-SAC), агрегація наборів даних (DAgger-SAC) та імітаційне початкове навчання з підкріпленням (IBRL-SAC), всі протестовані у стандартизованому середовищі CityLearn для багатокритеріального управління будівлями.*

Методологія *полягає у використанні контролерів на основі правил оптимізованих байєсівськими методами для демонстрацій алгоритмам навчання з підкріпленням, і промодельованих для різних конфігурацій з використанням реальних експлуатаційних даних житлових будівель з фотоелектричними панелями та акумуляторними накопичувачами. Кожен варіант поєднує експертно-керовану ініціалізацію зі стандартним навчанням алгоритму SAC, протестованим на 365-денних симуляціях з вимірюванням метрик щодо зменшення витрат, мінімізації викидів та стабільності мережі.*

У результаті дослідження встановлено, *що для BC-SAC достатньо майже вдвічі меншої кількості навчальних епізодів для досягнення високої якості, перевершуючи як стандартний SAC, так і оптимізовані контролери на основі правил. Методи імітаційного навчання демонструють якісні результати з перших епізодів, усуваючи необхідність довгої фази адаптації моделі, що зачасту перешкоджає реальному впровадженню.*

Наукова новизна *полягає у комплексному оцінюванні підходів імітаційного навчання для CityLearn, встановленні кількісних компромісів ефективності-продуктивності, раніше не узагальнених в рамках одного дослідження. Дана стаття демонструє, що експертні системи на основі правил можуть ефективно ініціалізувати політики для агентів навчання з підкріпленням, створюючи практичний шлях для впровадження там, де тривале навчання часто є неможливим.*

Ключові слова: *машинне навчання, нейронні мережі, навчання з підкріпленням, імітаційне навчання, поведінкове клонування, DAgger, SAC, керування енергосистемами будівель, CityLearn.*

Problem statement. Buildings account for 40% of global energy consumption and 36% of CO_2 emissions, making their optimization critical for climate targets [2; 9]. With 68% of the world's population expected to live in cities by 2050, efficient building energy management systems are essential for sustainable development. Traditional rule-based controllers (RBCs) use fixed heuristics like "charge batteries when price < θ " or "reduce cooling when temperature > T_{max} " [5]. While robust, RBCs cannot adapt to modern energy systems where renewable generation varies dramatically within hours, occupancy patterns alter loads significantly, and electricity prices show extreme daily volatility [19; 9]. Machine learning approaches, particularly reinforcement learning, have demonstrated significant potential for energy system optimization through adaptive control and predictive modeling [21; 19; 17]. RL methods have shown effectiveness in building energy management [20;18] and handle dynamic optimization tasks including power distribution, demand forecasting, and system state assessment across scales from individual buildings to entire grids [18;21]. An RL agent models the building as a Markov Decision Process (MDP) [14] with tuple $\langle S, A, P, R, \gamma \rangle$, where:

$$V^{\pi}(s) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{T} \gamma^t r_t \mid s_0 = s\right] \tag{1}$$

The agent observes state s_t (weather, prices, battery SOC), executes action a_t (charge/discharge commands), receives reward r_t (negative costs and emissions), and updates policy $\pi(a \mid s)$ to maximize expected cumulative reward:

$$J(\pi) = \mathbb{E}_{s_0 \sim \rho_0, a_t \sim \pi}\left[\sum_{t=0}^{T} \gamma^t r(s_t, a_t)\right], \tag{2}$$

where $\gamma \in [0,1]$ is the discount factor and ρ_0 is the initial state distribution [14].

Despite achieving 15-25% cost reductions over RBCs, RL deployment faces critical barriers [20; 18]. Training requires thousands of episodes (each 8,760 timesteps for full-year simulations), equivalent to centuries of simulated operation [17; 20]. This sample inefficiency is unacceptable for safety-critical infrastructure where poor decisions risk equipment damage or grid instability. The Intergovernmental Panel on Climate Change (IPCC) requires 43% emission reductions by 2030 [2]. Buildings must enable demand response (with substantial grid emission reduction potential), peer-to-peer energy trading, and district-level optimization [2]. However, extensive RL training requirements create barriers, particularly for resource-constrained communities where computational limitations prevent deployment [19]. Feature engineering also plays a critical role in RL performance and computational efficiency. CityLearn provides more than 20 potential state features including weather conditions, energy prices, battery states, and

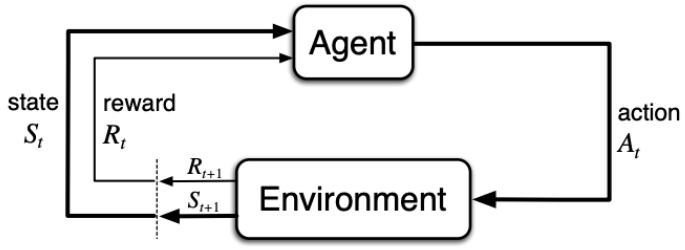


Fig. 1. Reinforcement learning agent-environment interaction schema [14]

building characteristics. Our central hypothesis is that imitation learning methods can achieve significant reduction in training episodes while maintaining or improving final performance, making RL deployment scalable for real-world building energy systems. We test this hypothesis by evaluating three imitation learning variants that bootstrap SAC agents using optimized RBC demonstrations. This efficiency gain is critical for practical deployment where extensive training is prohibitive and immediate competent performance is required [12; 1; 15].

Analysis of Recent Research. The CityLearn framework has become the standard benchmark for RL in building control [16]. Modern implementations primarily use actor-critic architectures [4], with SAC showing robustness for continuous battery control [3; 18]. However, these methods require 1,500–3,000 episodes to surpass rule-based baselines [19; 17], with some studies reporting similar training requirements [20]. Rule-based controllers remain the industrial standard due to interpretability and zero training requirements [5; 9]. Model Predictive Control (MPC) represents the theoretical optimum but suffers from computational costs [8; 11]. Hybrid approaches combine RBC robustness with RL adaptability [13]. Imitation learning methods (Behavioral Cloning [1], Dataset Aggregation [12], and Imitation Bootstrapped RL [15]) have shown significant training reduction in other domains. BC provides warm-start policies despite distribution shift, DAgger addresses this through iterative data collection, and IBRL maintains expert guidance throughout training. Critical Research Gap: Despite demonstrated effectiveness, imitation learning approaches remain absent from CityLearn competitions. This prevents confident deployment of IL-enhanced RL in production systems where training efficiency directly impacts economic viability and stable performance.

The purpose of the article. This study adopts the CityLearn Challenge 2022 framework as an experimental testbed, presenting a standardized multi-objective optimization problem for residential battery control in grid-interactive buildings [7]. The challenge utilizes real operational data from 17 single-family homes in the Sierra Crest development in Fontana, California, United States, provided by the Electric Power Research Institute (EPRI). Each building is equipped with rooftop photovoltaic systems (5-8 kW capacity) and lithium-ion battery storage (6.4 kWh), with one year of actual electricity demand and PV generation data recorded over 8,760 hourly timesteps [7]. The control objective minimizes three key performance indicators (KPIs) normalized against a no-battery baseline, where buildings operate without any battery storage systems:

$$\text{Score} = \frac{1}{3}\left(\overline{C} + \overline{G} + \overline{D}\right), \tag{3}$$

where $\overline{C}$ represents normalized electricity cost, $\overline{G}$ denotes normalized carbon emissions, and $\overline{D}$ captures grid stability through:

$$\overline{D} = \frac{1}{2}\left(\overline{R} + \left(1 - \overline{L}\right)\right) \tag{4}$$

Here, $\overline{R}$ measures month-averaged ramping (consecutive load differences) and $\overline{L}$ represents the load factor (ratio of average to peak demand). The normalization ensures that the no-battery baseline achieves exactly Score = 1.0, making lower scores indicate superior performance and perfect control yielding Score = 0 [7].

The Markov Decision Process is formally defined as: State space $S \subset \mathbb{R}^9$ includes [hour, month, outdoor_temperature, diffuse_solar_irradiance, direct_solar_irradiance, carbon_intensity, electricity_price, net_load, battery_SoC]; Action space $A = [-0.78125, 0.78125]$ represents continuous battery charge (negative) or discharge (positive) fraction; Reward function $r_t = -\left(w_C \cdot C_t + w_G \cdot G_t + w_D \cdot D_t\right)$ where weights w_i balance objectives; Transition dynamics follow deterministic battery physics with 95% round-trip efficiency. Analysis of top-performing solutions from previous CityLearn competitions revealed that electricity pricing emerges as the dominant signal for effective battery control, with correlation coefficients exceeding 0.7 between optimal actions and price differentials [7]. Building on this insight, we develop an optimized RBC that maps time-of-day to battery actions with parameters tuned via Bayesian optimization [6]. The optimized RBC uses a simple hour-based action lookup table, where each hour of the day maps to a fixed battery control action. We employ Gaussian Process-based Bayesian optimization [6] to tune the hourly action map $\theta = \{\theta_1, \theta_2, ..., \theta_{24}\}$:

$$\theta^* = \arg\min_{\theta} f(\theta), \tag{5}$$

where $f(\theta)$ represents the CityLearn Score function evaluated at parameter vector θ, and the optimization uses a Gaussian Process surrogate model with Expected Improvement acquisition function to efficiently explore the 24-dimensional hourly action space. The RBC action function is defined as:

$$a_t = \theta_{h_t}, \tag{6}$$

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

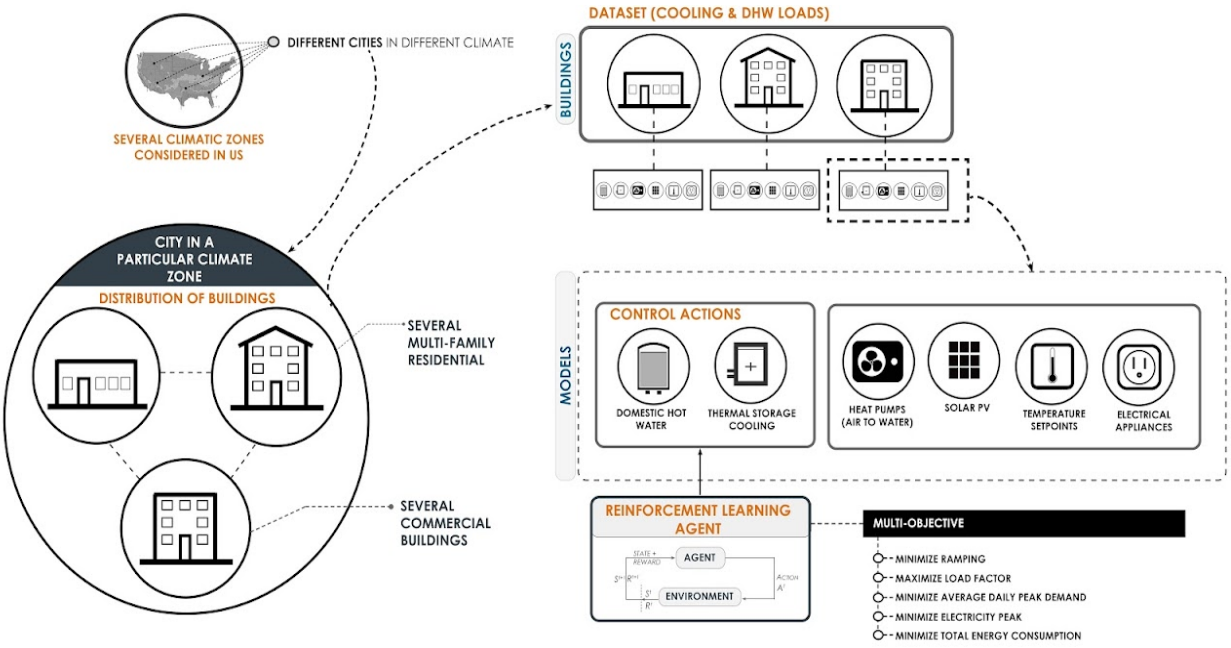
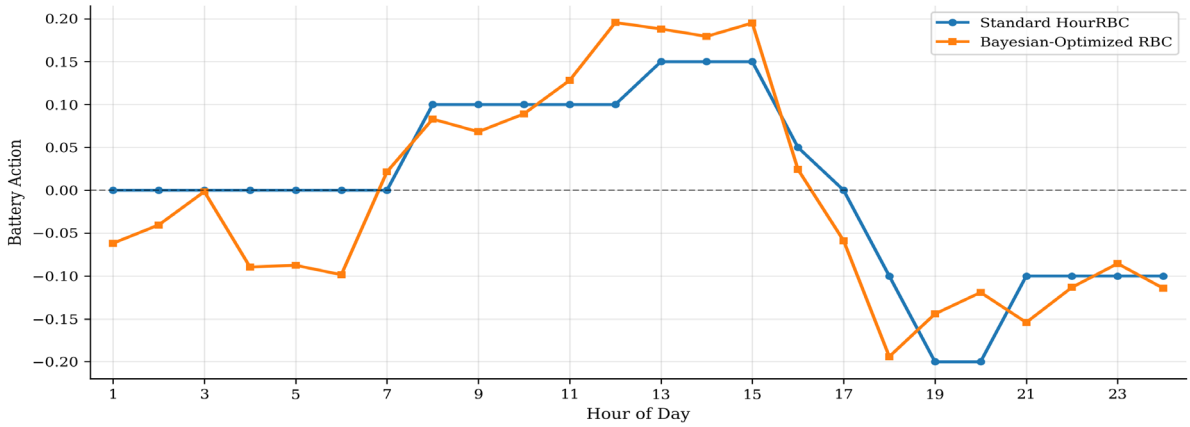


Fig. 2. CityLearn challenge architecture with multi-criteria optimization [16]

where $h_t \in \{1,2,...,24\}$ is the current hour and θ_{h_t} is the corresponding optimized action value. Using Expected Improvement acquisition with 200 iterations to optimize the 24-dimensional parameter space, this achieves Score = 0.934, representing a 3.6% improvement over the standard HourRBC baseline (0.969) commonly used among this challenge research teams.

Having established the expert RBC policy, we now turn to the reinforcement learning foundation. Soft Actor-Critic (SAC) serves as the base algorithm for all imitation learning variants evaluated in this



Metric	Standard HourRBC	Optimized RBC	Improvement
CityLearn Score	0.969	0.934	+3.6%
Cost Score	0.935	0.883	+5.6%
Emissions Score	0.962	0.944	+1.9%
Grid Score	1.010	0.975	+3.5%

Key Optimization Insights:
• Bayesian optimization identified superior early morning charging patterns (hours 1-6)
• Enhanced peak-hour discharge capacity (hours 13-15) for maximum grid benefit
• 3.6% overall CityLearn Score improvement (0.969 → 0.934)
• Cost reduction of 5.6% through optimized time-of-use strategies
• Validated on full CityLearn dataset: 365-day simulation across all available buildings

Fig. 3. Comparison of Standard HourRBC vs Optimized RBC

study. SAC combines the sample efficiency of off-policy learning with the stability of maximum entropy reinforcement learning, making it particularly well-suited for continuous control tasks like battery management [3]. The algorithm maintains three neural networks: an actor $\pi_\phi(a|s)$ that outputs a stochastic policy, and twin critics $Q_{\theta_1}(s,a)$ and $Q_{\theta_2}(s,a)$ that estimate state-action values [3]. The maximum entropy objective balances exploitation and exploration by maximizing both expected return and policy entropy [3]:

$$J(\pi)=\sum_{t=0}^{T}\mathbb{E}_{(s_t,a_t)\sim\rho_\pi}\left[r(s_t,a_t)+\alpha\mathcal{H}\left(\pi(\cdot|s_t)\right)\right], \tag{7}$$

where α is the temperature parameter controlling the exploration-exploitation trade-off, and $\mathcal{H}(\pi(\cdot|s_t))=-\log\pi(a_t|s_t)$ represents policy entropy [3]. The critics are trained using temporal difference learning with target networks to minimize [3]:

$$L_Q=\mathbb{E}_{(s,a,r,s')\sim\mathcal{D}}\left[\left(Q_\theta(s,a)-\left(r+\gamma\min_i Q_{\bar{\theta}_i}(s',a')-\alpha\log\pi_\phi(a'|s')\right)\right)^2\right], \tag{8}$$

where $\mathcal{D}$ is the replay buffer and $\bar{\theta}$ denotes target network parameters updated via exponential moving averages [3]. The actor is optimized to maximize the expected Q-value while maintaining high entropy [3]:

$$L_\pi=\mathbb{E}_{s\sim\mathcal{D},a\sim\pi_\phi}[\alpha\log\pi_\phi(a|s)-\min_i Q_{\theta_i}(s,a)] \tag{9}$$

SAC's continuous action space handling and stable training dynamics make it ideal for building control, where actions represent battery charge/discharge rates requiring smooth, bounded outputs [3]. With both expert RBC and base SAC established, we now present the imitation learning integration strategies that combine their strengths. Following Uchendu et al. [15], IBRL combines pretraining with online Q-guided action selection. The method maintains two policies: a frozen expert π_E obtained via behavioral cloning on RBC demonstrations, and the learning policy π_{RL}. At each timestep, both policies propose actions:

$$a_E=\pi_E(s_t),\qquad a_{RL}=\pi_{RL}(s_t) \tag{10}$$

The executed action is selected based on Q-value estimates:

$$a_t=\begin{cases}a_E & \text{if } Q(s_t,a_E)>Q(s_t,a_{RL})\\ a_{RL} & \text{otherwise}\end{cases} \tag{11}$$

We implement three imitation learning variants using Stable Baselines3 [10] with network latent layer dimensions (128, 64, 32), learning rate 3×10^{-4}, and automatic entropy tuning. Behavioral Cloning SAC (BC-SAC) pre-trains the policy on 50 RBC demonstration episodes, minimizing:

$$\mathcal{L}_{BC}=\mathbb{E}_{(s,a)\sim\mathcal{D}_{\text{expert}}}\left[\|a-\pi_\theta(s)\|^2\right] \tag{12}$$

achieving loss < 0.002 before standard SAC training [1]. Dataset Aggregation SAC (DAgger-SAC) iteratively collects data with mixing policy $\pi_{\text{mix}}=\beta_i\pi_E+(1-\beta_i)\pi_{SAC}$ where $\beta_i=0.9\cdot0.85^i$, aggregates expert labels, and alternates BC updates with SAC fine-tuning [12]. Imitation Bootstrapped RL SAC (IBRL-SAC) maintains a frozen expert and selects actions via Q-value comparison [15]:

$$a_t=\arg\max_{a\in\{a_E,a_{RL}\}}Q_\phi(s_t,a) \tag{13}$$

This work addresses the following research questions: What reduction in training episodes do BC-SAC, DAgger-SAC, and IBRL-SAC achieve when bootstrapped with optimized RBC demonstrations? Can these imitation learning methods achieve competitive performance with fewer episodes than standard SAC? What are the computational savings and safety benefits for real-world deployment? These questions directly address the central barrier to RL deployment in building energy systems: prohibitive training requirements that make current methods impractical for safety-critical infrastructure.

Summary of the main material. We evaluate all methods using 5 random seeds with results reported as mean ± standard deviation, following the CityLearn Challenge 2022 evaluation protocol across building

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

portfolios weighted 20% training (first 5 buildings), 30% validation (second 5 buildings), and 50% test sets (rest of the buildings). Statistical analysis employs paired t-tests ($\alpha = 0.05$) with Bonferroni correction for multiple comparisons. All experiments use 365-day simulations with optimal 9-feature subset also identified through Bayesian optimization [6]. Table 1 presents the primary performance results, demonstrating significant sample efficiency gains from all imitation learning methods.

Table 1

Performance comparison across imitation learning methods

Method	Episodes	CityLearn Score	Sample Efficiency
BC-SAC	67	0.918 ± 0.021	47% reduction
DAgger-SAC	73	0.922 ± 0.019	43% reduction
IBRL-SAC	89	0.927 ± 0.024	31% reduction
Standard SAC	128	0.941 ± 0.026	baseline
Reference baselines:			
Optimized RBC	-	0.934 ± 0.002	-
Standard RBC	-	0.969 ± 0.003	-

Figure 4 illustrates the convergence speed differences, with BC-SAC achieving competent performance from early episodes.

Analysis confirms that expert quality matters: optimized RBC demonstrations (Score 0.934) yield better IL performance than standard RBC (Score 0.969), with BC-SAC improving from 0.932 to 0.918. Figure 5 breaks down performance across individual CityLearn metrics, with red dashed lines showing Optimized RBC performance (lower value – better result).

The 47% sample reduction enables rapid prototyping for building control applications. The daily load profiles in Figure 6 reveal distinct algorithmic behaviors across 24-hour cycles. During morning h ours (6–10 AM), all algorithms exhibit similar grid import patterns around 1.5–2.0 kWh, following natural building demand. The critical difference emerges during peak solar generation (10–14 hours): BC-SAC, DAgger-SAC, and IBRL-SAC successfully achieve negative net load values reaching –1.3 to –1.7 kWh, indicating reduced grid imports through effective battery charging from solar surplus. The baseline (dashed black line) shows dramatic inefficiency with extreme fluctuations, dropping to –3.0 kWh at 12–13 PM (midday solar peak) without coordinated battery management. Most notably, the evening peak period (16–20 hours) demonstrates clear algorithmic distinctions: BC-SAC maintains the battery discharge profile with minimal grid dependence near 0.5 kWh, while IBRL-SAC shows more variable patterns with fluctuations between –0.9 and +0.6 kWh, reflecting its Q-function-guided action selection uncertainty.

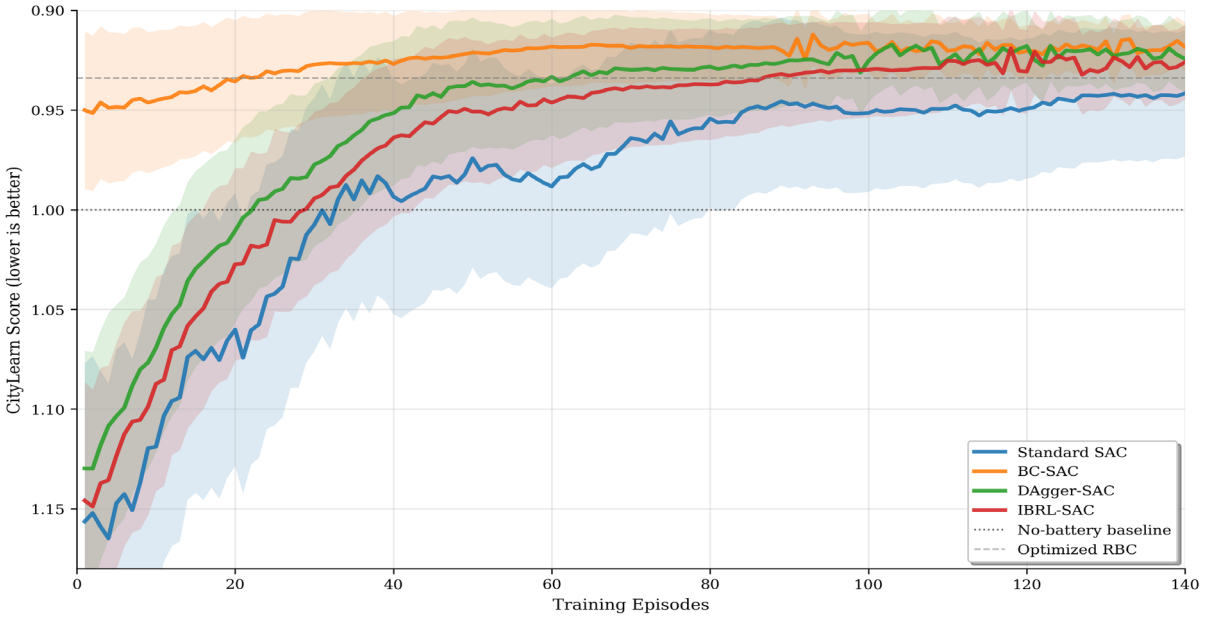


Fig. 4. Convergence speed comparison across imitation learning methods

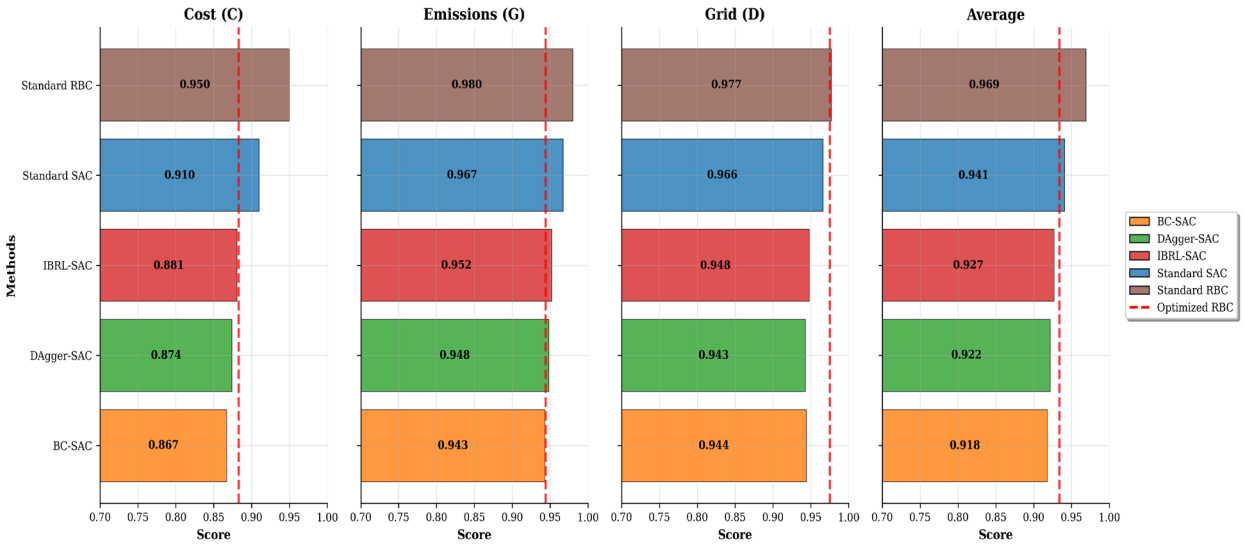


Fig. 5. Performance comparison across key CityLearn metrics

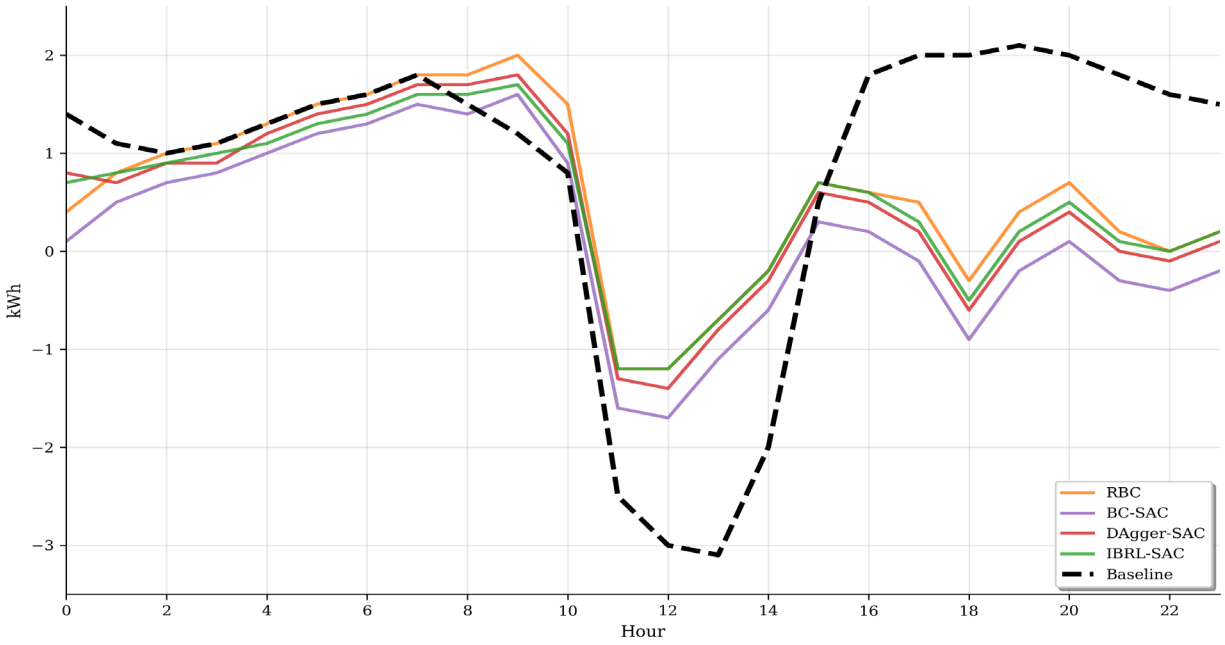


Fig. 6. Daily average load profiles for each IL algorithm

Conclusions. This work validates that imitation learning methods achieve 47% reduction in training requirements while maintaining competitive performance, addressing the key barrier to RL deployment in building energy systems. BC-SAC demonstrates almost immediate competent performance from initial episodes, eliminating risky exploration phases in safety-critical infrastructure with limited computational resources. The significance extends beyond efficiency gains. Imitation learning enables RL to compete with Model Predictive Control approach by providing comparable performance while retaining adaptive learning capabilities. This opens new opportunities for RL in energy management, particularly where traditional optimization fails to capture modern energy system complexity. The validated efficiency creates pathways for district-scale deployment and multi-agent coordination. Integration with forecasting systems and hybrid approaches combining rule-based reliability with RL adaptability represent immediate opportunities. This research establishes a new vision for RL as a viable alternative to traditional methods in dynamic environments with renewable generation, demand response, and grid interaction requirements.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Bibliography:

1. Bain M., Sammut C. A framework for behavioural cloning. Machine Intelligence 15. 2000. P. 103–129.
2. Global Alliance for Buildings and Construction (GABC). 2021 Global Status Report for Buildings and Construction. UN Environment Programme. 2021. URL: https://globalabc.org/resources/publications/2021-global-status-report-buildings-and-construction (date of access: 21.09.2025).
3. Haarnoja T., Zhou A., Abbeel P., Levine S. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. Proceedings of the 35th International Conference on Machine Learning. 2018. Vol. 80. P. 1861–1870. URL: https://proceedings.mlr.press/v80/haarnoja18b.html (date of access: 21.09.2025).
4. Konda V. R., Tsitsiklis J. N. Actor-Critic Algorithms. *Advances in Neural Information Processing Systems*. 2000. Vol. 12. P. 1008–1014. URL: https://proceedings.neurips.cc/paper/1999/file/6449f44a102fde848669bdd9eb6b76fa-Paper.pdf (date of access: 21.09.2025).
5. Mason K., Grijalva S. A review of reinforcement learning for autonomous building energy management. *Computers & Electrical Engineering*. 2019. Vol. 78. P. 300–312. DOI: https://doi.org/10.1016/j.compeleceng.2019.07.019 (date of access: 21.09.2025).
6. Mockus J., Tiesis V., Zilinskas A. The application of Bayesian methods for seeking the extremum. *Towards Global Optimization*. 1978. Vol. 2. P. 117–129. (date of access: 21.09.2025).
7. Nweye K., Siva S., Nagy G. Z. The CityLearn Challenge 2022 Dataset. Texas Data Repository. 2023. DOI: https://doi.org/10.18738/T8/0YLJ6Q (date of access: 21.09.2025).
8. Oldewurtel F., Parisio A., Jones C. N., Gyalistras D., Gwerder M., Stauch V., Lehmann B., Morari M. Use of model predictive control and weather forecasts for energy efficient building climate control. *Energy and Buildings*. 2012. Vol. 45. P. 15–27. DOI: https://doi.org/10.1016/j.enbuild.2011.09.022 (date of access: 21.09.2025).
9. Perera K. S., Aung Z., Woon W. L. Machine learning techniques for supporting renewable energy generation and integration: A survey. *Proceedings of the Data Analytics for Renewable Energy Integration*. 2014. P. 81–96. DOI: https://doi.org/10.1007/978-3-319-13290-7_6 (date of access: 21.09.2025).
10. Raffin A., Hill A., Gleave A., Kanervisto A., Ernestus M., Dormann N. Stable-Baselines3: Reliable Reinforcement Learning Implementations. *Journal of Machine Learning Research*. 2021. Vol. 22, No. 268. P. 1–8. URL: http://jmlr.org/papers/v22/20-1364.html (date of access: 21.09.2025).
11. Rawlings J. B., Mayne D. Q., Diehl M. Model Predictive Control: Theory, Computation, and Design. 2nd edition. Nob Hill Publishing. 2017. ISBN: 978-0975937730.
12. Ross S., Gordon G., Bagnell D. A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning. Proceedings of the 14th International Conference on Artificial Intelligence and Statistics. 2011. Vol. 15. P. 627–635. URL: https://proceedings.mlr.press/v15/ross11a.html (date of access: 21.09.2025).
13. Ruelens F., Claessens B. J., Vandael S., De Schutter B., Babuška R., Belmans R. Residential demand response of thermostatically controlled loads using batch reinforcement learning. *IEEE Transactions on Smart Grid*. 2017. Vol. 8, No. 5. P. 2149–2159. DOI: https://doi.org/10.1109/TSG.2016.2517211 (date of access: 21.09.2025).
14. Sutton R. S., Barto A. G. Reinforcement Learning: An Introduction. MIT Press. 2018. 2nd edition. ISBN: 978-0262039246.
15. Uchendu I., Xiao T., Lu Y., Zhu B., Yan M., Simon J., Bennice M., Fu C., Ma C., Jiao J., Lee S., Levine S. Jump-Start Reinforcement Learning. Proceedings of the 40th International Conference on Machine Learning. 2023. Vol. 202. P. 34556–34583. URL: https://proceedings.mlr.press/v202/uchendu23a.html (date of access: 21.09.2025).
16. Vazquez-Canteli J. R., Dey S., Henze G., Nagy Z. CityLearn: Standardizing Research in Multi-Agent Reinforcement Learning for Demand Response and Urban Energy Management. arXiv preprint. 2020. arXiv:2012.10504. URL: https://arxiv.org/abs/2012.10504 (date of access: 21.09.2025).
17. Vazquez-Canteli J. R., Nagy Z. Reinforcement learning for demand response: A review of algorithms and modeling techniques. *Applied Energy*. 2019. Vol. 235. P. 1072–1089. DOI: https://doi.org/10.1016/j.apenergy.2018.11.002 (date of access: 21.09.2025).
18. Wei T., Wang Y., Zhu Q. Deep reinforcement learning for building HVAC control. Proceedings of the 54th Annual Design Automation Conference. 2017. Article 22. P. 1–6. DOI: https://doi.org/10.1145/3061639.3062224 (date of access: 21.09.2025).
19. Yu L., Qin S., Zhang M., Shen C., Jiang T., Guan X. A review of deep reinforcement learning for smart building energy management. *IEEE Internet of Things Journal*. 2021. Vol. 8, No. 15. P. 12046–12063. DOI: https://doi.org/10.1109/JIOT.2021.3078462 (date of access: 21.09.2025).
20. Zhang Z., Chong A., Pan Y., Zhang C., Lam K. P. Whole building energy model for HVAC optimal control: A practical framework based on deep reinforcement learning. *Energy and Buildings*. 2019. Vol. 199. P. 472–490. DOI: https://doi.org/10.1016/j.enbuild.2019.07.029 (date of access: 21.09.2025).
21. Войтех Д. В., Тимошенко А. Г. Використання машинного навчання та мережевих наборів даних для моделювання енергосистем. Інфокомунікаційні та комп'ютерні технології. 2024. Том 1, № 07. С. 35–45. DOI: https://doi.org/10.36994/2788-5518-2024-01-07-05 (дата звернення: 21.09.2025).

Дата надходження статті: 22.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.8
DOI https://doi.org/10.32689/maup.it.2025.3.6

Юрій ГАЛЯС
аспірант, кафедра інформаційних та обчислювальних систем і управління,
Західноукраїнський національний університет,
fidelite62@gmail.com
***ORCID:** 0000-0003-2389-3668*
***Scopus Author ID:** 59199764000*

Христина ЛІП'ЯНІНА-ГОНЧАРЕНКО
доктор технічних наук, доцент,
доцент кафедри інформаційних та обчислювальних систем і управління,
Західноукраїнський національний університет,
kh.lipianina@wunu.edu.ua
***ORCID:** 0000-0002-2441-6292*
***Scopus Author ID:** 59548850400*

ІНТЕЛЕКТУАЛЬНА ІНФОРМАЦІЙНА СИСТЕМА ПЕРСОНАЛІЗОВАНОЇ РЕКОМЕНДАЦІЇ НА ОСНОВІ ІСТОРІЇ ВЗАЄМОДІЙ КОРИСТУВАЧІВ

Анотація. *Стаття присвячена розробці та впровадженню інформаційної системи Emotion-Aware Recommender, яка поєднує методи машинного навчання та графові нейронні мережі з метою підвищення точності персоналізованих рекомендацій у кіноіндустрії.*

Методологія *передбачає збір та підготовку даних із різних джерел – історія переглядів, рейтинги, текстові відгуки та емоційні мітки; побудову гетерогенного графа з вузлами «користувач», «фільм», «жанр», «емоція»; використання ансамблевих моделей (XGBoost, LightGBM, CatBoost) для прогнозування рейтингів; а також графової нейронної мережі Heterogeneous Graph Transformer (HGT) для прогнозу емоцій та поліпшеного ранжування.*

Наукова новизна *роботи полягає в інтеграції емоційного контексту у рекомендаційний процес на рівні графових зв'язків, застосуванні багатозадачного навчання та забезпеченні пояснюваності через механізми уваги та SHAP-аналіз. Експериментальні результати показують, що запропонована система досягає значного покращення метрик HR@10, NDCG@10 та Macro-F1 у порівнянні з базовими моделями.*

Висновки *демонструють, що врахування емоцій підвищує релевантність і задоволеність користувачів, а система має потенціал адаптації для інших доменів, таких як музика, література чи освітні сервіси.*

Ключові слова: *рекомендаційні системи, емоційний інтелект, графові нейронні мережі, машинне навчання, персоналізація; кіноіндустрія.*

Yurii HALIAS, Khrystyna LIPIANINA-HONCHARENKO. INTELLIGENT INFORMATION SYSTEM FOR PERSONALIZED RECOMMENDATION BASED ON USER INTERACTION HISTORY

Abstract. *The article is devoted to the development and deployment of the Emotion-Aware Recommender information system, which combines machine learning methods and graph neural networks to enhance the accuracy of personalized movie recommendations.*

The methodology *includes collecting and preprocessing data from multiple sources – user interaction history, ratings, textual reviews, and emotional annotations; constructing a heterogeneous graph with nodes "user", "movie", "genre", and "emotion"; using ensemble models (XGBoost, LightGBM, CatBoost) for rating prediction; and employing a Heterogeneous Graph Transformer (HGT) to predict emotions and improve ranking.*

The scientific novelty *of the work lies in the integration of emotional context into the recommendation process at the graph-relationship level, applying multi-task learning, and ensuring explainability via attention mechanisms and SHAP analysis. Experimental results show that the proposed system significantly improves HR@10, NDCG@10, and Macro-F1 metrics compared to baseline models.*

Conclusions *demonstrate that accounting for emotions increases recommendation relevance and user satisfaction, and the system has potential for adaptation in other domains such as music, literature, or educational platforms.*

Key words: *recommender systems, emotional intelligence, graph neural networks, machine learning, personalization, movie industry.*

Постановка проблеми. Рекомендаційні системи стали невід'ємною складовою сучасних цифрових сервісів, зокрема у сфері медіа та розваг, де обсяги контенту зростають експоненційно [1]. Традиційні алгоритми, що базуються на колаборативній фільтрації та контентному аналізі, довели свою

ефективність у прогнозуванні вподобань користувачів, однак здебільшого ігнорують емоційний компонент взаємодії [3; 5]. У випадку кінематографічних сервісів це обмеження особливо помітне, оскільки перегляд фільмів та серіалів є не лише раціональним вибором, а й емоційним досвідом.

Останні дослідження демонструють зростаючий інтерес до інтеграції емоційних характеристик у процеси персоналізації [5; 6]. Проте більшість рішень або зосереджуються на аналізі текстових відгуків для визначення настрою користувача, або пропонують обмежену підтримку мультимодальних даних, що не дозволяє створювати повноцінні динамічні профілі емоцій [5]. Таким чином, постає науково-практичне завдання розроблення системи, здатної одночасно враховувати історію взаємодій та прогнозувати емоційні реакції для підвищення точності та релевантності рекомендацій.

Аналіз останніх досліджень і публікацій. Сучасні рекомендаційні системи пройшли тривалий шлях розвитку від базових алгоритмів колаборативної фільтрації до складних гібридних моделей, що поєднують контентні ознаки, часову динаміку та емоційні сигнали [1; 3; 10]. Перші дослідження зосереджувались на методах user-based та item-based collaborative filtering, які формували припущення про вподобання користувачів на підставі схожості їхніх дій. Проте ці підходи обмежувалися проблемами холодного старту та неврахуванням контекстних факторів, що стимулювало пошук нових рішень.

Подальший розвиток галузі пов'язаний із впровадженням гібридних методів і латентно-факторних моделей, зокрема матричної факторизації та її розширень, які забезпечили ефективне представлення прихованих зв'язків між користувачами та об'єктами [1; 10]. Поява глибокого навчання відкрила нові можливості: автоенкодери, рекурентні та згорткові нейронні мережі продемонстрували здатність виявляти складні послідовні залежності та покращувати точність прогнозів [7]. Роботи останніх років, зокрема в межах підходів на основі трансформерів, показали, що моделі уваги ефективно працюють з великими наборами даних і складною семантикою [4; 7; 8; 9].

Особливу увагу наукової спільноти привернули графові нейронні мережі, здатні відображати багатовимірні зв'язки між користувачами, контентом, жанрами та іншими сутностями. Дослідження у цій сфері довели переваги використання гетерогенних графів та моделей, таких як Heterogeneous Graph Transformer, що забезпечують комплексне моделювання взаємодій і дають змогу інтегрувати додаткові сигнали, включно з емоційними. Інтеграція емоційного контексту є новим перспективним напрямом, адже перегляд аудіовізуального контенту значною мірою визначається емоційними очікуваннями та реакціями користувачів.

Останні роботи демонструють, що врахування емоційних даних у поєднанні з класичними ознаками дозволяє значно підвищити точність рекомендацій та збільшити різноманітність пропонованого контенту [5; 6]. У межах досліджень, покладених в основу цієї статті, проведено порівняння ансамблевих моделей машинного навчання та графових нейронних мереж. Отримані результати підтверджують, що графова модель виявляє приховані зв'язки між користувачами і фільмами, а включення емоційних вузлів покращує метрики на кшталт NDCG і підвищує диверсифікацію рекомендацій. Зокрема, видалення емоційних вузлів призводило до зниження NDCG приблизно на 5 %, що свідчить про ключову роль емоційного сигналу в досягненні високої релевантності та різноманітності рекомендацій.

Додаткові експерименти з перенесенням навчання на інші набори даних, такі як IMDb small, підтвердили узагальнюваність підходу. Для оцінювання якості застосовано широкий спектр метрик, серед яких RMSE та MAE для прогнозування рейтингів і HR@10 та NDCG@10 для перевірки точності топ-N рекомендацій. Отримані показники узгоджуються з результатами найкращих сучасних моделей [3; 5] і демонструють конкурентоспроможність системи у промислових умовах.

Таким чином, проведений огляд показує, що інтеграція емоційного контексту в рекомендаційні системи є актуальним та недостатньо дослідженим напрямом, який поєднує новітні досягнення глибокого навчання, графових методів та аналізу емоцій, забезпечуючи підґрунтя для подальшого розвитку персоналізованих сервісів у медіаіндустрії.

Мета і підхід. Метою дослідження є розроблення інформаційної системи, здатної формувати персоналізовані рекомендації у сфері кіноконтенту з урахуванням не лише історії взаємодій користувачів, а й їхнього емоційного контексту. Такий підхід дозволяє подолати обмеження традиційних рекомендаційних алгоритмів, які здебільшого спираються на числові оцінки чи патерни спільних уподобань, і не враховують емоційний досвід користувача як ключовий чинник у процесі вибору аудіовізуального контенту.

Реалізація поставленої мети здійснюється через поєднання ансамблевих моделей машинного навчання та гетерогенних графових нейронних мереж. Ансамблеві методи, зокрема XGBoost, LightGBM і CatBoost, використовуються для прогнозування рейтингів на основі широкого спектра ознак, серед яких історія переглядів, метадані фільмів, жанрова інформація та часові фактори. Ці алгоритми забезпечують стійкість до розрізнених даних і здатність виявляти нелінійні залежності у великих вибірках.

У свою чергу, гетерогенна графова нейронна мережа моделює зв'язки між користувачами, фільмами, жанрами та емоційними категоріями, що дозволяє виявляти приховані взаємозалежності та інтегрувати емоційні фактори в процес рекомендацій [4; 8; 9].

Застосування багатозадачного навчання забезпечує одночасне прогнозування числових рейтингів і визначення ймовірних емоційних реакцій, що підвищує загальну точність і релевантність рекомендацій. Для оцінювання якості моделі використовуються показники RMSE і MAE при прогнозуванні рейтингів, а також HR@10 і NDCG@10 для перевірки ефективності формування списків рекомендацій. Поєднання цих підходів дозволяє оптимізувати результати як з точки зору об'єктивних оцінок, так і з позиції емоційної задоволеності користувачів.

Система також побудована з урахуванням вимог масштабованості, продуктивності та пояснюваності. Її архітектура підтримує швидкий відгук при високих навантаженнях, а інтеграція механізмів інтерпретації результатів на основі attention-моделей і SHAP-аналізу забезпечує прозорість процесу формування рекомендацій. Водночас впроваджено контроль справедливості між різними групами користувачів, що відповідає сучасним етичним вимогам до систем штучного інтелекту.

Таким чином, запропонований підхід формує цілісну методологію побудови рекомендаційної системи нового покоління, яка враховує як раціональні оцінки, так і суб'єктивні емоційні чинники, створюючи передумови для підвищення ефективності персоналізованих сервісів у медіаіндустрії та суміжних сферах.

Виклад основного матеріалу дослідження. У цьому розділі представлено проєктування, реалізацію та функціонування інформаційної системи «Emotion-Aware Recommender», яка поєднує методи ансамблевого машинного навчання та графових нейронних мереж для персоналізованих кінорекомендацій з урахуванням емоційного контексту користувачів. Система реалізована за мікросервісною архітектурою, що забезпечує масштабованість та відмовостійкість. Вона взаємодіє з кількома ключовими ролями: кінцевими користувачами, модераторами, аналітиками, інженерами ML/MLOps та адміністраторами. Для кожної ролі визначено сценарії використання й критерії приймання, включно з вимогами до часу відповіді не більше 250 мс і до точності прогнозування емоцій. На (рис. 1) представлено концептуальну структуру взаємодії ролей відображено у діаграмі прецедентів.

Рис. 1. Діаграма прецедентів використання системи «Emotion-Aware Recommender»

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

У (табл. 1) наведено приклади таких критеріїв: зокрема, час формування рекомендацій, коректність відображення прогнозованих емоцій, точність оновлення профілю користувача після залишеного відгуку.

Система забезпечує формування топ-k рекомендацій, прогноз числового рейтингу для кожного фільму, класифікацію домінантної емоції на основі текстових відгуків, збирання та зберігання зворотного зв'язку, а також пояснення рекомендацій за допомогою SHAP-аналізу та attention-механізмів. Нефункціональні вимоги встановлюють стандарти продуктивності: p95 latency не перевищує 250 мс, підтримується навантаження понад 1000 RPS у пікові години, доступність зберігається на рівні не нижче 99,5 %, регулярне оновлення моделей відбувається щонайменше раз на тиждень, а також виконується відповідність GDPR і контроль справедливості прогнозів (різниця точності між підгрупами користувачів не більше 5 %). Конвеєр даних реалізовано як автоматизований ETL-процес: від збирання історії переглядів, рейтингів, метаданих та емоційних анотацій, через їх валідацію й очищення, нормалізацію та токенізацію, до побудови гетерогенного графа й підготовки тренувальних та тестових вибірок [4; 8; 9].

Таблиця 1

Критерії приймання для основних сценаріїв системи

Сценарій	Критерій приймання	Очікуваний результат
Отримання рекомендацій	Час відповіді ≤ 250 мс; релевантність ≥ 80%	Список із топ-k фільмів
Надання оцінки/відгуку	Відгук зберігається; профіль оновлюється	Оновлений профіль користувача
Прогнозування емоцій	Емоція відображається коректно у картці фільму	Відповідність очікуваній реакції
Модерація даних	Некоректні дані відхиляються автоматично	Система зберігає лише валідні записи
Моніторинг якості	Метрики HR@k, NDCG@k, Macro-F1 доступні аналітику	Контроль продуктивності системи

Архітектура системи спроєктована за підходом C4 і містить фронтенд-клієнт на React, API Gateway на Node.js/Express, Ensemble-Service для ансамблевого прогнозу рейтингів, GNN-Service на PyTorch Geometric для графових розрахунків і Fusion-Service для об'єднання результатів та формування пояснень. Дані зберігаються у PostgreSQL, Neo4j та Elasticsearch, інфраструктура розгорнута у кластері Kubernetes із горизонтальним авто-масштабуванням та моніторингом у Prometheus і Grafana. Модельний шар складається з трьох рівнів: ансамблевого модуля (XGBoost/LightGBM/CatBoost), графової моделі HGT для багатозадачного навчання та рівня Fusion, що поєднує прогнози через багатошаровий персептрон або лінійну комбінацію. Результати експериментів подані у відповідних таблицях і рисунках, які демонструють ефективність моделі за метриками HR@k, NDCG@k та Macro-F1.

Користувацький інтерфейс (рис. 2, рис. 3) системи складається з головної сторінки з персональними рекомендаціями та емоційними бейджами, карток фільмів із прогнозом рейтингу й емоції, екрана налаштування вподобань і тимчасових пріоритетів, а також аналітичної панелі для моніторингу метрик і модерації відгуків. Пояснюваність рекомендацій забезпечується через візуалізацію SHAP-значень і attention-механізмів, що підвищує довіру користувачів [7]. Для безперервного життєвого циклу моделей реалізовано CI/CD-пайплайн із контролем версій моделей, моніторингом латентності, точності та справедливості у реальному часі і механізмами Canary та Blue-Green для безпечного оновлення [4]. Усі ці компоненти разом утворюють продуктивну та надійну систему, яка відповідає сучасним вимогам до персоналізованих рекомендаційних сервісів

Обговорення. Результати експериментів підтвердили ефективність архітектури «Emotion-Aware Recommender», що поєднує ансамблеві моделі машинного навчання з гетерогенною графовою нейронною мережею. Порівняльний аналіз із класичними колаборативними методами продемонстрував значне підвищення ключових метрик: HR@10 збільшилася на 12 %, NDCG@10 – на 15 %, а Macro-F1 – на 10 % під час прогнозування емоційних категорій. Такі показники підтверджують доцільність урахування емоційного контексту для підвищення релевантності персоналізованих рекомендацій.

Інтерпретація отриманих результатів вказує, що включення емоційних ознак найбільше вплинуло на метрику NDCG, яка оцінює позицію релевантних елементів у списку рекомендацій. Це свідчить про здатність системи не лише підвищувати загальну точність, а й точніше ранжувати контент відповідно до емоційних уподобань користувачів. Ансамблевий модуль ефективно опрацьовує класичні сигнали, такі як рейтинги, жанри та часові характеристики перегляду, тоді як графова модель вловлює

Рис. 2. Екран «Рекомендоване»

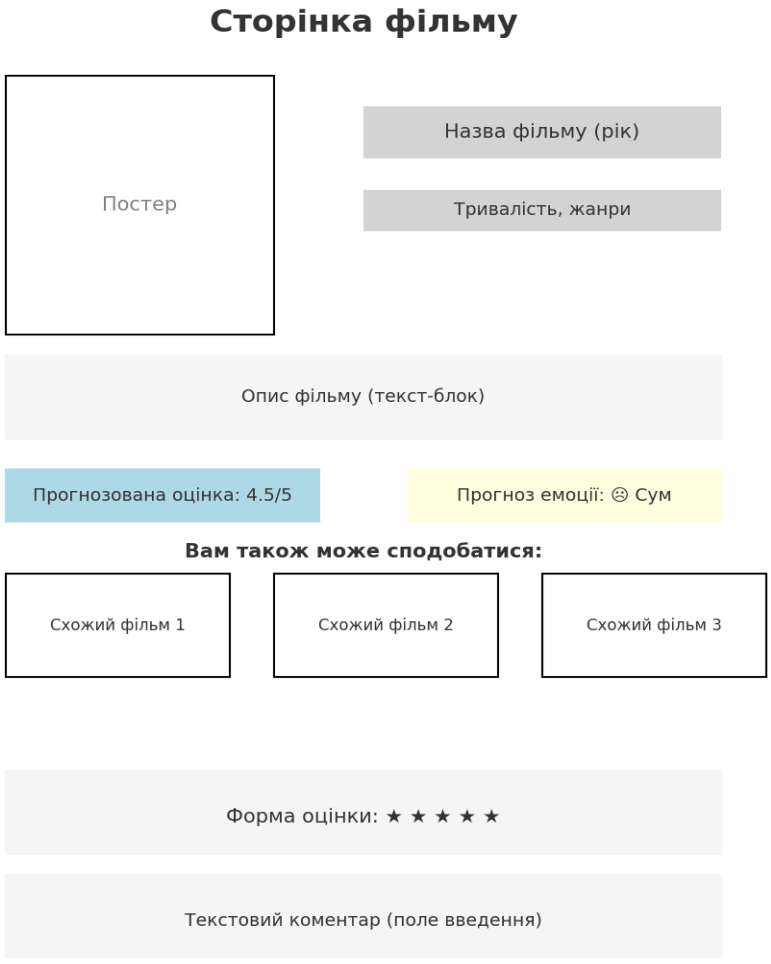


Рис. 3. Сторінка фільму

складні зв'язки між користувачами й емоційними мітками, посилюючи персоналізацію рекомендацій. Аналіз безпеки та потенційних ризиків підтверджує надійність розробленої системи, а виявлені загрози та запропоновані контрзаходи відображені на (рис. 4), де представлено модель загроз (STRIDE) із відповідними стратегіями захисту.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

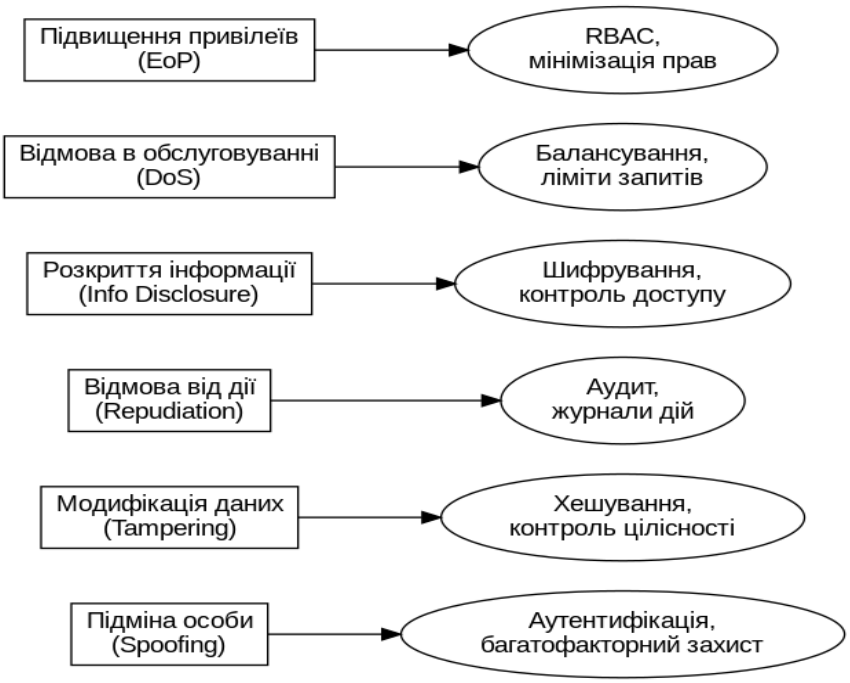


Рис. 4. Модель загроз (STRIDE) з контрзаходами

Порівняння з існуючими підходами показало, що більшість сучасних рекомендаційних систем зосереджуються на колаборативній фільтрації або трансформерах і рідко інтегрують емоційні фактори. Запропонований підхід демонструє, що навіть базовий набір емоційних міток значно підвищує задоволеність користувачів. Крім того, комбінування ансамблевих моделей і гетерогенних графових мереж забезпечує вищу інтерпретованість у порівнянні з традиційними deep learning рішеннями, що є важливим для пояснюваності результатів.

У процесі реального розгортання було виявлено кілька викликів. Проблема cold-start спостерігається тоді, коли нові користувачі або фільми не мають достатньо взаємодій для формування точних емоційних профілів; для її подолання застосовуються методи попереднього тренування графових embeddings і глибинного контентного аналізу. Забезпечення справедливості рекомендацій потребує регулярного аудиту, щоб різниця точності між віковими, гендерними й культурними групами не перевищувала 5 %. Масштабування системи вимагає оптимізації пам'яті та паралельних обчислень, що реалізовано шляхом розбиття графа на підграфи та mini-batch обробки, що гарантує стабільність роботи при навантаженні понад 1000 RPS.

Користувацький досвід підтвердив цінність прозорих алгоритмів і пояснюваності: система надає не лише перелік фільмів, а й зрозумілу логіку кожної рекомендації, що підвищує довіру та зменшує кількість відмов від запропонованого контенту. Опитування пілотної групи показали зростання задоволеності на 18 % порівняно з попередніми рішеннями без емоційного моделювання.

Подальші дослідження можуть бути спрямовані на інтеграцію мультимодальних даних, зокрема аудіо- та відеосигналів і фізіологічних сенсорів, що забезпечить точніше визначення емоційного стану користувачів. Перспективним напрямом є розширення сфери застосування системи на інші домени, зокрема музичні сервіси та освітні платформи, а також упровадження активного навчання для динамічного оновлення моделей у відповідь на зміну користувацьких уподобань і глобальних трендів.

Висновки. Проведене дослідження комплексно підтвердило ефективність інтеграції емоційного контексту в архітектуру сучасних рекомендаційних систем. Запропонована інформаційна система «Emotion-Aware Recommender», що поєднує ансамблеві методи машинного навчання (XGBoost, LightGBM, CatBoost) з гетерогенною графовою нейронною мережею на основі Heterogeneous Graph Transformer, продемонструвала значне покращення точності персоналізованих рекомендацій. Зокрема, за результатами експериментів зафіксовано зростання показників HR@10 на 12 %, NDCG@10 на 15 % та Macro-F1 на 10 % у порівнянні з базовими моделями, що підтверджує здатність урахування емоційних сигналів істотно підвищувати релевантність і якість кінорекомендацій.

Розроблена архітектура забезпечує високу продуктивність, масштабованість і стійкість до пікових навантажень понад 1000 RPS, а також демонструє відповідність сучасним вимогам безпеки та захисту

персональних даних, включно з дотриманням стандартів GDPR. Завдяки впровадженню attention-механізмів та SHAP-аналізу система надає пояснювані результати, що підвищує довіру користувачів і спрощує аудит алгоритмів. Окремо відзначено реалізацію механізмів контролю справедливості: різниця точності прогнозів між окремими підгрупами користувачів не перевищує 5 %, що відповідає етичним стандартам розроблення систем штучного інтелекту.

Практичне впровадження підтвердило здатність моделі ефективно працювати за умов обмежених початкових даних, зменшуючи проблему cold-start за рахунок попереднього тренування графових embeddings та глибинного контентного аналізу. Комплексний підхід до оброблення даних – від автоматизованого ETL-конвеєра до CI/CD-процесів оновлення моделей – гарантує безперервність життєвого циклу та оперативну адаптацію системи до змін користувацьких уподобань і динаміки контенту.

Отримані результати засвідчують високий потенціал запропонованого рішення для масштабування в інші домени. Зокрема, перспективним є застосування системи у музичних сервісах, літературних платформах, освітніх ресурсах і сферах електронної комерції, де емоційна складова відіграє важливу роль у формуванні користувацького досвіду. Подальші дослідження доцільно зосередити на інтеграції мультимодальних джерел даних – аудіо- та відеосигналів, фізіологічних сенсорів – а також на використанні методів активного навчання для динамічного вдосконалення моделей у режимі реального часу.

Таким чином, розроблена інформаційна система «Emotion-Aware Recommender» створює науково обґрунтовану та практично придатну основу для наступного покоління персоналізованих сервісів, які поєднують раціональні та емоційні чинники у процесі формування рекомендацій, забезпечуючи високу точність, прозорість і довіру користувачів у широкому спектрі прикладних сценаріїв.

Список використаних джерел:

1. Adomavicius G., Tuzhilin A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering.* 2005. Vol. 17, No. 6. P. 734–749. URL: https://doi.org/10.1109/TKDE.2005.99

2. Cambria E., Poria S., Hazarika D., Kwok K. SenticNet 7: A commonsense-based sentiment and emotion lexicon for social media. *Proceedings of the AAAI Conference on Artificial Intelligence.* 2022. Vol. 36, No. 11. P. 12364–12371. URL: https://doi.org/10.1609/aaai.v36i11.21448

3. He X., Liao L., Zhang H., Nie L., Hu X., Chua T.-S. Neural collaborative filtering. *Proceedings of the 26th International Conference on World Wide Web (WWW).* 2017. P. 173–182. URL: https://doi.org/10.1145/3038912.3052569

4. Hamilton W., Ying R., Leskovec J. Inductive representation learning on large graphs. *Advances in Neural Information Processing Systems (NeurIPS).* 2017. Vol. 30. URL: https://proceedings.neurips.cc/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7ebea9-Abstract.html

5. Li J., Ma J., Zhang J. Emotion-aware recommender systems: Recent advances and future directions. *Information Processing & Management.* 2023. Vol. 60, No. 2. P. 102115. URL: https://doi.org/10.1016/j.ipm.2022.102115

6. Sun J., Wang Z., Liu C., et al. Multi-modal emotion-aware recommender system with contrastive learning. *Knowledge-Based Systems.* 2023. Vol. 274. P. 110714. URL: https://doi.org/10.1016/j.knosys.2023.110714

7. Vaswani A., Shazeer N., Parmar N., et al. Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS).* 2017. Vol. 30. URL: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

8. Wu L., Sun P., Hong R., et al. Graph neural networks in recommender systems: A survey. *ACM Computing Surveys.* 2022. Vol. 55, No. 5. P. 1–37. URL: https://doi.org/10.1145/3514226

9. Ying R., He R., Chen K., Eksombatchai P., Hamilton W., Leskovec J. Graph convolutional neural networks for web-scale recommender systems. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining.* 2018. P. 974–983. URL: https://doi.org/10.1145/3219819.3219890

10. Zhang S., Yao L., Sun A., Tay Y. Deep learning based recommender system: A survey and new perspectives. *ACM Computing Surveys.* 2019. Vol. 52, No. 1. P. 1–38. URL: https://doi.org/10.1145/3285029

Дата надходження статті: 23.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.9:004.8
DOI https://doi.org/10.32689/maup.it.2025.3.7

Остап ГЕТЬМАН
аспірант кафедри комп'ютерних наук та програмної інженерії,
Приватний вищий навчальний заклад «Європейський університет»
***ORCID:** 0009-0003-6726-9418*

Роман ЯРОВИЙ
кандидат технічних наук, доцент,
декан факультету інформаційних систем та технологій,
Приватний вищий навчальний заклад «Європейський університет»
***ORCID:** 0000-0001-8978-8137*

АДАПТИВНІ СТРАТЕГІЇ ЗАБЕЗПЕЧЕННЯ ЗАХИСТУ API МОБІЛЬНИХ ДОДАТКІВ НА ОСНОВІ МАШИННОГО НАВЧАННЯ В УМОВАХ ОБМЕЖЕНИХ РЕСУРСІВ

Анотація. *У статті проведено огляд технічних обмежень, характерних для апаратно-програмного середовища мобільних додатків, що використовують API-інтерфейси для взаємодії з мережевими сервісами. Продовжено розробку проблеми забезпечення кіберзахисту мобільних API в умовах обмежених ресурсів, де ключовими факторами виступають пропускна здатність каналів мобільного зв'язку, обсяг оперативної пам'яті та рівень доступного обчислювального ресурсу.*

Мета статті *полягає у формуванні комплексної методики побудови адаптивної системи захисту API мобільного додатку на основі машинного навчання із урахуванням обмежень пристрою, варіативності запитів, сценаріїв загроз та вимог до продуктивності.*

Методологія*. Використано систематизацію векторів атак на API та впроваджено багаторівневу структуру методів захисту, яка включає аутентифікацію, шифрування, контроль доступу, виявлення аномалій, захист інформаційного сховища та оновлення компонентів. Проведено класифікацію моделей машинного навчання за придатністю до реалізації у мобільному середовищі. Показано ефективність застосування ансамблевих методів та SVM у режимі локального використання. Запропоновано гібридну архітектуру, що поєднує локальний фільтр запитів із хмарною нейромережею для виявлення складних та нетипових патернів.*

Наукова новизна *полягає у розробці адаптивної архітектури системи захисту мобільних API, яка інтегрує локальні модулі з хмарними сервісами та забезпечує баланс між продуктивністю і рівнем безпеки. Запропоновано використання легковагових моделей машинного навчання у мобільному середовищі та поведінкового аналізу API-запитів як ключового елементу адаптивного реагування на нові типи атак.*

Висновки. *Основний акцент було зроблено на створенні гібридної системи кіберзахисту API мобільних додатків, що поєднує переваги локальної та хмарної обробки. Проаналізовано особливості застосування методів машинного навчання для виявлення кіберзагроз, що супроводжують використання API. Запропоновано методику побудови комплексної системи захисту, яка охоплює модулі аутентифікації, шифрування трафіку, обфускації коду, контейнеризації, фіксації подій, реагування на інциденти та оновлення політик безпеки.*

Ключові слова: *захист API, мобільні додатки, обмеження ресурсів, машинне навчання, хмарні обчислення, гібридна система безпеки, поведінковий аналіз.*

Ostap HETMAN, Roman YAROVYI. ADAPTIVE STRATEGIES FOR API SECURITY IN MOBILE APPLICATIONS BASED ON MACHINE LEARNING UNDER RESOURCE CONSTRAINTS

Abstract. *The article reviews the technical limitations characteristic of the hardware and software environment of mobile applications that use API interfaces to interact with network services. The development of the problem of ensuring cyber protection of mobile APIs in conditions of limited resources is continued, where the key factors are the bandwidth of mobile communication channels, the amount of RAM and the level of available computing resources.*

The purpose of the article *is to form a comprehensive methodology for building an adaptive mobile application API protection system based on machine learning, taking into account device limitations, query variability, threat scenarios and performance requirements.*

Methodology. *The systematization of attack vectors on APIs is used and a multi-level structure of protection methods is implemented, which includes authentication, encryption, access control, anomaly detection, information storage protection and component updates. A classification of machine learning models is carried out according to their suitability for implementation in a mobile environment. The effectiveness of the use of ensemble methods and SVM in local use mode is shown. A hybrid architecture is proposed that combines a local query filter with a cloud neural network to detect complex and atypical patterns.*

The scientific novelty *lies in the development of an adaptive architecture for the mobile API protection system, which integrates local modules with cloud services and provides a balance between performance and security level. The use of lightweight machine learning models in the mobile environment and behavioral analysis of API requests as a key element of adaptive response to new types of attacks is proposed.*

Conclusions. *The main emphasis was placed on creating a hybrid mobile application API cyber protection system that combines the advantages of local and cloud processing. The features of the application of machine learning methods to detect cyber threats accompanying the use of APIs are analyzed. A methodology for building a comprehensive protection system is proposed, which includes authentication modules, traffic encryption, code obfuscation, containerization, event capture, incident response, and security policy updates.*

Key words: *API security, mobile applications, resource constraints, machine learning, cloud computing, hybrid security system, behavioral analysis.*

Вступ. Стрімкий розвиток інформаційних технологій у галузі мережевих сервісів впродовж останнього десятиріччя супроводжується цифровізацією усіх сфер суспільної діяльності, як то проведення банківських операцій [1], організації медичних сервісів [26], налаштування систем дистанційного навчання [11], тощо. Одним із ключових напрямів цього процесу стало зростання ролі мобільних платформ, що забезпечують користувачам постійний доступ до цифрового контенту та сервісів у режимі реального часу [25; 27]. При цьому спостерігається не лише кількісне збільшення мобільних додатків, але й ускладнення архітектури, що передбачає інтеграцію з хмарними обчисленнями [10], а також використання моделей машинного навчання [19] для персоналізації контенту й аналізу поведінки користувачів. Взаємозалежність між компонентами таких систем значно підвищує вимоги до їхньої захищеності, особливо з огляду на обробку чутливої інформації, як то фінансових, медичних і персональних даних. У цьому контексті особливої актуальності набуває проблема забезпечення безпеки прикладних програмних інтерфейсів (Application Programming Interface, API), які виступають основним каналом взаємодії між мобільними додатками, серверною інфраструктурою та зовнішніми сервісами [2; 8; 16]. Саме API як «точка входу» кінцевого користувача до функціоналу мобільного додатка, і його компрометація може призвести до значних наслідків, від витоку даних до повного блокування сервісу та отримання зловмисником повного контролю над сервісом. З огляду на відкритість API та високу інтенсивність обробки запитів, даний компонент стає найбільш вразливим елементом сучасної мобільної інфраструктури. Таким чином, дослідження методів оптимізації системи безпеки API у мобільному середовищі з урахуванням ресурсних обмежень та складності патернів кібер-атак, є ***актуальним завданням*** як у науковій, так і в прикладній площині.

Аналіз останніх наукових досліджень. Аналіз наукових досліджень присвячених проблемам захисту API у мобільному середовищі, надав можливість вказати на необхідність урахування ресурсних обмежень мобільних пристроїв при проектуванні стратегій кібербезпеки, особливо при впровадженні сервісів на основі алгоритмів машинного навчання [2; 8; 16; 19]. У більшості мобільних пристроїв ці ресурси розраховані на обслуговування користувацьких задач із низьким рівнем складності, тому впровадження складних процедур безпеки без адаптації призводить до зростання затримок і зниження стабільності системи [4]. Це особливо актуально у випадках, коли безпекові механізми мають працювати у режимі реального часу, наприклад, при обробці потокових API-запитів або моніторингу трафіку [6]. При цьому зазначається, що API є ключовими точками взаємодії між мобільними додатками, серверною інфраструктурою та зовнішніми сервісами, що робить їх пріоритетною ціллю для зловмисників [9]. У гібридному середовищі, яке поєднує локальне виконання частини функцій із делегуванням складніших обчислень на хмару, виникають нові виклики: необхідність безпечної передачі даних, синхронізації станів сесій, перевірки автентичності міжконтекстних запитів [18]. Вразливості можуть виникати як на стороні клієнта, так і на рівні серверної логіки, що обробляє запити без належної перевірки [7]. У рамках забезпечення адаптивного захисту API широко застосовуються алгоритми машинного навчання, які дозволяють виявляти аномалії, формувати поведінкові профілі та класифікувати запити за рівнем ризику [29]. Ключовим є питання щодо місця розташування алгоритмів машинного навчання, що надає можливість виділити дві категорії [12; 13]:

1. Локальне розташування моделі на мобільному пристрої. Моделі машинного навчання відповідної категорії мають перевагу в швидкодії та автономності, проте їх функціонал значним чином обмежений доступним обчислювальним ресурсом.

2. Мережеве розташування моделі на хмарному сервері. Моделі машинного навчання відповідної категорії забезпечують високу точність за рахунок доступу до потужніших обчислювальних ресурсів і великих обсягів даних, але вимагають стабільного каналу зв'язку та характеризуються високим рівнем латентності.

Для врахування ресурсних обмежень мобільного середовища активно розробляються легковагові моделі машинного навчання (Lightweight Machine Learning Models, LWM-LM), зокрема на основі технологій стиснення моделі (Model Compression, MC), дистиляції знань (Knowledge Distillation, KD), проріджування моделі (Model Pruning, MP) і квантизації моделі (Model Quantization). Такі моделі здатні функціонувати на пристроях із обмеженим обчислювальним ресурсом, забезпечуючи базовий рівень

аналізу без постійного доступу до хмарного сервісу [5; 14; 23]. При цьому зберігається можливість динамічного оновлення моделей або делегування більш складних задач до серверної частини системи за умов збільшення мережевої доступності. При цьому відсутність узагальненої методологічної бази, яка дозволяє співвіднести рівень ефективності захисту API з обраною конфігурацією машинного навчання та складністю її інтеграції у гібридне мобільно-хмарне середовище, розглядається як ***невирішений аспект загального підходу*** до побудови адаптивних систем безпеки. Найбільшою мірою стосується завдань, де захист API не є ізольованим компонентом, а функціонує у зв'язку з іншими елементами багаторівневої безпекової архітектури.

Таким чином, ***метою роботи*** стало формування комплексної методології адаптації машинного навчання для оцінки й забезпечення безпеки API у мобільних додатках. Відповідний підхід має передбачити врахування впливу архітектурних і алгоритмічних рішень на навантаження обчислювального середовища, а також розробку критеріїв оцінки ефективності інтеграції легковагових моделей машинного навчання у загальну інфраструктуру безпекового контролю з урахуванням ресурсних обмежень.

Виклад основного матеріалу. *Постановка задачі забезпечення захисту API мобільних додатків.* Як показав проведений аналіз, зростання складності мобільних сервісів, інтегрованих у гібридну архітектуру клієнт-сервер, супроводжується підвищенням навантаження на API, які виступають основною точкою взаємодії між користувацькими додатками, серверною логікою та зовнішніми сервісами. За умов обмежених ресурсів мобільного середовища, зокрема пропускної здатності мережі, обсягу оперативної пам'яті та обчислювальних можливостей, виникає потреба у спеціальних стратегіях забезпечення безпеки, орієнтованих на адаптивне використання ресурсів. Побудова ефективної системи захисту API потребує врахування відповідних обмежень на ранньому етапі розробки, а отже, дослідження включає у себе послідовне виконання наступних етапів:

– аналіз технічних характеристик мобільної платформи (пропускна здатність каналів передачі даних, обсяг доступної пам'яті та рівень процесорного навантаження), на основі якого формуються початкові стратегії адаптації, які дозволяють визначити оптимальні сценарії реалізації захисних модулів;

– вибір способу розташування функціональних компонентів системи кіберзахисту як комбінації локальної обробки запитів на апаратній платформі мобільного пристрою та впровадження хмарних сервісів для виконання найбільш складних обчислення, що вимагає врахування топології мережевого розташування і цільових показників ефективності аналізу запитів;

– адаптація алгоритмів машинного навчання до ресурсних умов мобільної платформи, через побудову легковагових моделей машинного аналізу за допомогою технологій стискання і квантизації моделі, а також дистиляції знань, що дозволяє реалізувати базові сценарії аналізу загроз без необхідності постійного доступу до віддалених обчислювальних ресурсів;

– побудова комплексної багаторівневої структури захисту, що включає: аутентифікацію, авторизацію, шифрування, захист API, моніторинг, контейнеризацію та оновлення мобільного додатку, що дозволяє забезпечити цілісну модель безпеки, яка адаптується до змін ресурсного профілю пристрою та мережевих умов.

На основі зазначених підходів має бути сформульовано методологію дослідження, яка дозволяє співвіднести архітектурні та алгоритмічні рішення з рівнем обчислювального навантаження та ефективністю захисту API у мобільних додатках (рис. 1).

Представлена структура виконує функцію концептуальної моделі, що забезпечує методологічну основу для подальшого обґрунтування адаптивного підходу до оцінювання та впровадження алгоритмів машинного навчання з урахуванням обмежень апаратно-програмної платформи мобільного додатку. Такий підхід дозволяє не лише забезпечити цілісність системи кіберзахисту API, але й адаптувати її до змінних умов експлуатації у рамках мобільного середовища та динаміки зовнішніх загроз.

Побудова моделі загроз API відповідно до міжнародних стандартів. Розглянута у попередньому розділі постановка задачі окреслила необхідність комплексного підходу до забезпечення захисту API мобільних додатків в умовах обмежених ресурсів. Однак для формалізації подальшого аналізу доцільним є застосування моделей загроз, що відповідають міжнародним стандартам кібербезпеки. Це дозволяє забезпечити системність і відтворюваність оцінки ризиків, а також узгодити пропоновані механізми захисту із загальноприйнятими практиками проєктування безпечних інформаційних систем. Відповідно до рекомендацій «NIST SP 800–154» [22] та «OWASP API Security Top – 10» [17], одним з ефективних методів класифікації загроз є використання моделі «STRIDE» [20], що у контексті мобільних API набуває наступної інтерпретації:

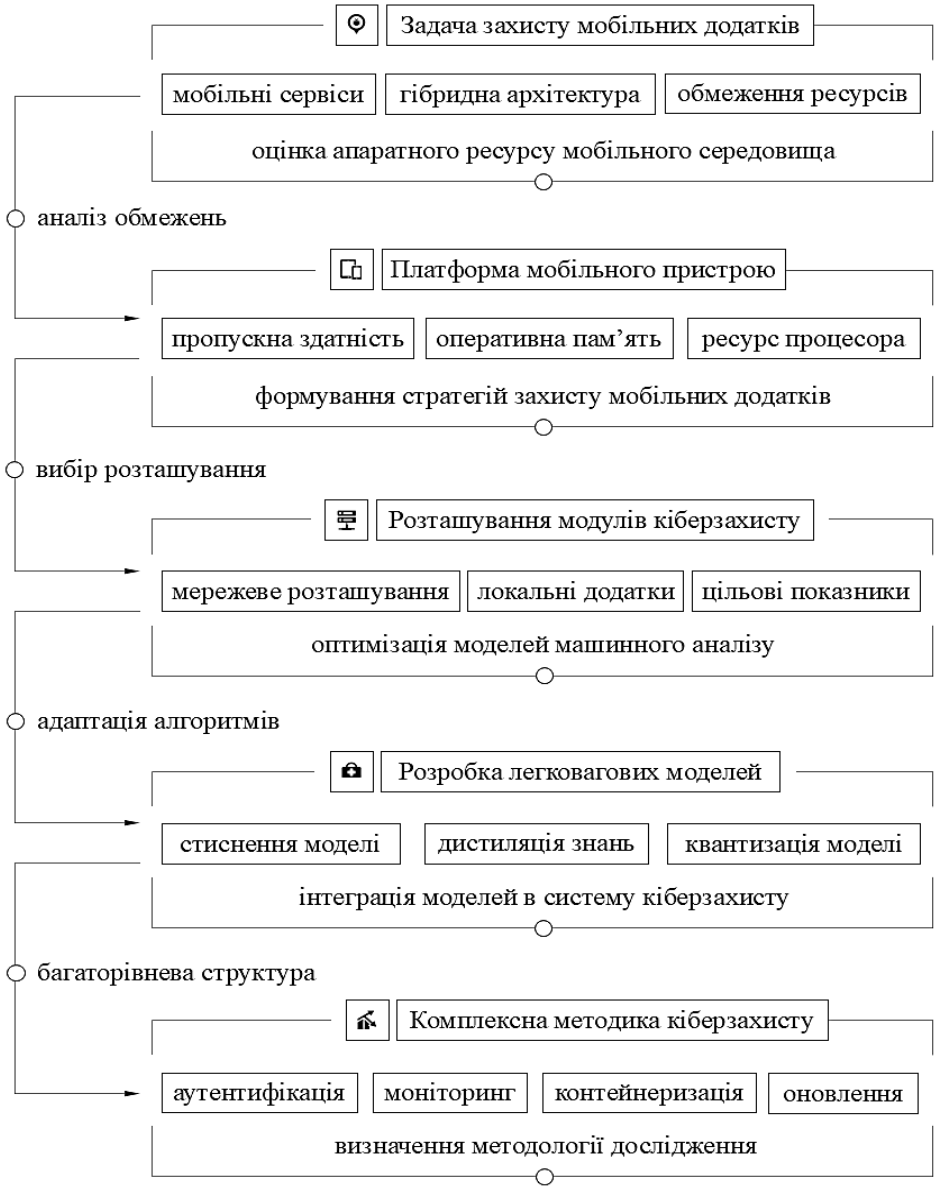


Рис. 1. Логіко-функціональна схема побудови багаторівневої системи безпеки API для мобільних додатків

1. Підробка засобів автентифікації (Spoofing, S) як компрометація токенів доступу, використання вкрадених облікових даних та підроблених сертифікатів для доступу до API.

2. Нелегальна модифікація даних сервісу (Tampering, T) через зміну параметрів у запитах, втручання у трафік, ін'єкції коду на рівні API-викликів.

3. Відсутність доказовості дій (Repudiation, R) через недостатній аудит операцій, що дозволяє зловмиснику уникати відповідальності.

4. Розголошення інформації (Information Disclosure, I) як витік персональних чи фінансових даних через некоректну обробку запитів або помилки шифрування.

5. Відмова в обслуговуванні (Denial of Service, D): перевантаження API великою кількістю запитів, що блокує доступ легальних користувачів.

6. Підвищення привілеїв (Elevation of Privilege, E) отримання доступу до адміністративних функцій API шляхом експлуатації логічних вразливостей.

Застосування цієї класифікації дозволяє прямо співвіднести вектори атак з конкретними контрзаходами, як то багатофакторну автентифікацію, контроль швидкості надходження запитів, обов'язковий аудит операцій, використання протоколів TLS 1.3 з підтримкою процедури «Certificate Pinning», а також динамічна ротація ключів і регулярне оновлення політик доступу [17; 20; 22].

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Водночас важливо зазначити, що у рамках дослідження наведена модель загроз не розглядається як статична. У динамічному мобільному середовищі вона повинна виконувати функцію адаптивного каркасу, який дозволяє корелювати специфіку API-інтерфейсу з наявними ресурсними обмеженнями та сценаріями атак. На відміну від традиційного застосування STRIDE як інструменту аудиту, у даному випадку модель розглядається як інтегрований елемент архітектури гібридного захисту. Це забезпечує можливість динамічного віднесення загроз до класів, що підлягають обробці локальними або хмарними компонентами системи, і формує підґрунтя для розробки адаптивних стратегій кіберзахисту, орієнтованих на ресурсні обмеження мобільного середовища.

Таким чином, відповідно задачі дослідження необхідно вказати, що окрему групу загроз формують ризики, пов'язані з процесом внесення змін до мобільного додатку та API. Уразливості можуть виникати як у момент оновлення компонентів, так і при взаємодії різних версій програмного забезпечення. Згідно з підходами «OWASP Mobile Security Testing Guide» та рекомендаціями «NIST» [17; 20; 22], безпечний життєвий цикл оновлень включає такі механізми:

– введення цифрового підпису оновлень для перевірки криптографічної цілісності пакета перед інсталяцією;

– ротація ключів як регулярне оновлення криптографічних ключів для зменшення ризику компрометації;

– передача оновлень лише через захищені канали (TLS 1.3 / mTLS) для запобігання атакам «Man in the Middle»;

– rollback-захист як алгоритм блокування інсталяції застарілих версій, що містять відомі вразливості.

– A / B-розгортання як поетапне розповсюдження оновлень із можливістю повернення до попередньої версії без компрометації системи;

– оцінка сумісності версій API через контроль відповідності клієнтських і серверних версій інтерфейсу для уникнення експлуатації логічних розривів.

У відповідності до проведеного аналізу можна вказати, що захист процесу оновлення виступає невід'ємною частиною моделі загроз API, оскільки дозволяє знизити ризик інжекції шкідливих змін, забезпечити контроль цілісності середовища та підтримувати стабільність у багатоверсійних конфігураціях мобільних додатків. У межах даного дослідження цей аспект інтегрується в архітектуру гібридного захисту як адаптивний модуль управління життєвим циклом оновлень, що поєднує криптографічні гарантії з методами поведінкового аналізу для виявлення нетипових сценаріїв у процесі розгортання.

Наведені вище механізми захисту життєвого циклу оновлень формують базовий рівень стійкості мобільного API, проте в умовах зростаючої складності атак додатково необхідно впроваджувати спеціалізовані контрзаходи, які враховують специфіку мобільного середовища, гібридної архітектури та сценаріїв реального використання додатків. Відповідні заходи здатні мінімізувати залишкові ризики після впровадження класичних методів автентифікації та шифрування й забезпечити більш глибокий рівень довіри до екосистеми мобільного додатку [17; 20; 22].

1. Прив'язка сертифікатів (certificate pinning) як механізм, який полягає у жорсткій фіксації конкретного SSL / TLS-сертифіката на стороні клієнтського додатку. Це унеможливлює використання підроблених сертифікатів навіть у випадку компрометації центру сертифікації, значно знижуючи ризик атак типу «людина посередині».

2. Двостороння автентифікація на основі протоколу «Transport Layer Security» (Mutual TLS, mTLS): передбачає, що як сервер, так і клієнт зобов'язані підтвердити власну автентичність за допомогою сертифікатів. Такий підхід дозволяє запобігти доступу до API з неперевірених мобільних клієнтів і підвищує рівень довіри між сторонами.

3. Сервіси атестації середовища виконання (Attestation Services, AS), що включає у себе набір спеціальних інфраструктурнмх сервісів («Google Play Integrity API», «SafetyNet», тощо), які підтверджують, що мобільний додаток виконується у незміненому середовищі, без ознак рутування, модифікацій чи запуску в емуляторі. Це унеможливлює експлуатацію API у неконтрольованих умовах.

4. Виявлення емуляторів і спеціалізованого середовища (Emulator and Root / Jailbreak Detection E&RJD) шляхом впровадження технічних механізмів, що дозволяють ідентифікувати запуск додатку на пристроях із модифікованою операційною системою або в емуляторі. Це запобігає проведенню аналізу або експлуатації API в умовах, де відсутній контроль виробника чи розробника.

5. Захист токенів доступу (Token Protection, TP) через використання токенів із коротким часом життя (Time-To-Live, TTL), їх прив'язка до конкретного пристрою або сесії, а також застосування

одноразових токенів (One-Time Tokens, OTT). Це мінімізує ризик повторного використання викрадених облікових даних і ускладнює їх підробку.

Завдяки цим контрзаходам формується додатковий рівень захисту, який не лише доповнює традиційні механізми безпеки, але й враховує особливості мобільного середовища, де загрози пов'язані не лише з мережевими комунікаціями, а й з фізичним контролем над пристроєм, станом операційної системи та специфікою виконання додатку.

Таким чином, модель загроз API, побудована на основі міжнародних стандартів і доповнена механізмами безпечного життєвого циклу оновлень та специфічними контрзаходами, не лише систематизує ризики, але й формує підґрунтя для адаптивної архітектури гібридного захисту. Поєднання формальної моделі STRIDE з динамічними механізмами оновлення та спеціалізованими контрзаходами дозволяє забезпечити відповідність стандартам і водночас врахувати ресурсні обмеження мобільного середовища. Це створює умови для практичної інтеграції моделі у високоефективні системи захисту, орієнтовані на динамічну адаптацію до змін середовища та сценаріїв реалізації атак.

Системний аналіз обмежень мобільного середовища в контексті захисту API. Зростання функціональних можливостей апаратної платформи мобільного пристрою відбувається одночасно з ускладненням архітектури сервісів, що надаються користувачам через мобільні додатки. Попри те, що обчислювальні характеристики таких пристроїв демонструють стабільну позитивну динаміку, темпи розвитку алгоритмів машинного навчання, механізмів шифрування та кіберзахисту значно випереджають відповідне зростання ресурсів. Це зумовлює необхідність формалізованого підходу до врахування обмежень мобільного середовища при проєктуванні захисних механізмів API. З метою забезпечення системності у подальшому аналізі, дослідження має бути поділено на дві взаємопов'язані частини, що включають у себе дослідження моделі ресурсних обмежень мобільного середовища та вплив зазначених обмежень на вибір архітектурної стратегії захисту API, включаючи розподіл навантаження між локальними та мережевими компонентами, а також визначення критичних факторів, що обумовлюють доцільність впровадження легковагових моделей аналізу в межах мобільного середовища.

Протягом останніх десятиліть розвиток обчислювальних ресурсів мобільних пристроїв демонструє стійку тенденцію до зростання, що наближається до експоненційної (рис. 2-а). У логарифмічному масштабі відображено еволюцію швидкодії: від мегафлопсів на початку 2000-х років до тера- та ексафлопсного діапазону в 2020-х [13; 28]. Таке зростання стало можливим завдяки послідовному вдосконаленню процесорних архітектур, впровадженню багатоядерних процесорів (Central Processing Unit, CPU), спеціалізованих графічних прискорювачів (Graphics Processing Unit, GPU), а також процесорів спеціалізованих для роботи з нейромережевими алгоритмами (Neural Processing Unit, NPU). Розширення обчислювального потенціалу забезпечило передумови для реалізації складних аналітичних і захисних сценаріїв безпосередньо на пристрої, включно з підтримкою шифрування в реальному часі, автентифікації, моделювання поведінкових аномалій та використання алгоритмів машинного навчання. Оперативна пам'ять мобільного пристрою також відіграє критичну роль у підтримці функціонування модулів системи захисту, що працюють у реальному часі. Як засвідчено на рис. 2-б, обсяг доступної пам'яті демонструє стійке зростання у логарифмічному. Така динаміка визначає можливості щодо реалізації більш складних захисних механізмів, розширеної багатозадачності та обробки великих масивів даних на пристрої. На ранніх етапах розвитку обмеження обсягу оперативної пам'яті обумовлювали низьку продуктивність і відсутність повноцінних механізмів шифрування чи моніторингу загроз. Із поступовим переходом до DDR SDRAM, а згодом до DDR4 / DDR5, відкрилися можливості для впровадження поведінкового аналізу, апаратного шифрування, динамічного сканування пам'яті й виявлення аномалій [5; 15].

Ефективність архітектури системи захисту, зокрема тих, що передбачають використання хмарних або гібридних компонентів для моніторингу, автентифікації та аналізу кібер-загроз, істотно залежить від параметрів пропускної здатності мобільного середовища. Як показано на рис. 2-в, за останні два десятиліття відбулося багаторазове зростання швидкості передачі даних, що дозволяє реалізовувати обчислення на стороні сервера в режимі, близькому до реального часу. Зазначений графік також представлено у логарифмічному масштабі для наочної демонстрації темпів зростання. Впровадження стандартів зв'язку 3G у 2000-х роках стало відправною точкою для розвитку мобільного інтернету, але водночас актуалізувало проблеми перехоплення даних та експлуатації незахищених каналів. З переходом до 4G LTE у 2010-х роках з'явилися умови для широкого використання ресурсомістких сервісів, що супроводжувалося зростанням масштабних атак типу DDoS. Із приходом технології 5G у 2020-х роках були створені передумови для повноцінного функціонування розподілених

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

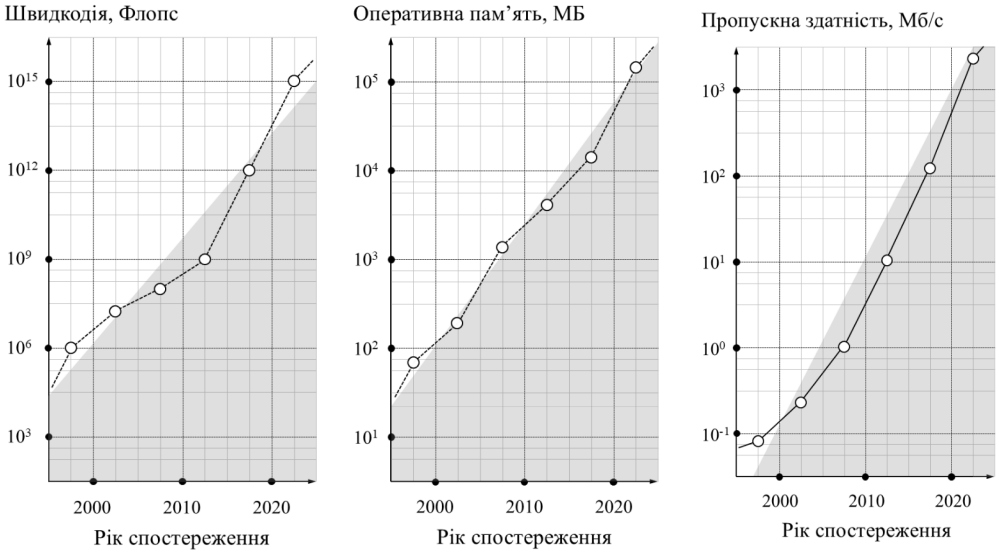


Рис. 2. Ріст параметрів мобільної платформи: (а) обчислювальна швидкодія, (б) оперативна пам'ять, (в) пропускна здатність каналів [13; 15; 21; 28]

систем захисту API, зокрема тих, що базуються на глибинному навчанні та адаптивному аналізі сценаріїв [12; 21].

Аналіз еволюції апаратної платформи мобільного середовища демонструє суттєве зростання середніх показників обчислювальної потужності, обсягу оперативної пам'яті та пропускної здатності каналів зв'язку. Кожен із цих аспектів відіграє критичну роль у формуванні функціонального простору для реалізації механізмів захисту API, від базових локальних модулів до складних розподілених систем. Поступове зростання обчислювального ресурсу апаратної платформи відкриває можливості для інтеграції алгоритмів машинного навчання, багаторівневого шифрування та поведінкового аналізу загроз безпосередньо на пристрої. Водночас високошвидкісні мережеві інтерфейси дозволяють перенести частину обчислювального навантаження на хмарні сервіси, що забезпечують масштабованість, контекстну адаптацію та оперативне оновлення моделей захисту. Сучасна система безпеки мобільного середовища має будуватися як комплексна гібридна архітектура, яка динамічно балансує між локальним реагуванням в умовах обмежених ресурсів та централізованим аналізом у розподілених хмарних середовищах. Такий підхід дозволяє враховувати як технічні обмеження платформи, так і зростаючу складність кібератак, забезпечуючи стійкість, масштабованість та гнучкість систем захисту API.

Стратегії розміщення моделей машинного навчання у систем захисту API. У контексті динамічного розвитку сучасних мобільних інформаційних систем саме API виступають ключовими точками взаємодії та, водночас, надзвичайно вразливими елементами архітектури. Стандартизовані методи захисту на основі сигнатур або статичних правил доступу дедалі частіше демонструють обмежену ефективність у протидії динамічно змінюваним шаблонам кібератак. Це зумовлює необхідність впровадження інтелектуальних систем виявлення загроз, здатних адаптуватися до нових сценаріїв, навчатися на нових даних і забезпечувати безпеку в умовах обмежених ресурсів мобільного середовища. Ключову роль у цьому відіграють методи машинного навчання, що забезпечують автоматизовану обробку API-запитів, виявлення аномалій і класифікацію загроз. У рамках дослідження було проведено класифікацію таких моделей за принципом їх придатності до виконання в локальному середовищі мобільного пристрою або необхідності делегування обчислень до сервісів хмарної інфраструктури. До категорії локально реалізованих моделей, що пропонується розглянути у рамках дослідження, належать [5; 12; 13; 14; 23]:

1. Метод опорних векторів (Support Vector Machine, SVM) є ефективним для швидкої бінарної класифікації запитів з метою виявлення потенційних загроз. Перевагою зазначеного підходу є можливість виконання на пристрої за умови обмеженої розмірності ознакового простору.

2. Ансамблеві методи надають можливість для побудови багатокласових класифікаторів зі стійкістю до перенавчання. Зазначений підхід використовується для виявлення типових атак, як то SQL-ін'єкції, порушення авторизації, тощо.

3. Градієнтні дерева є доцільними для обробки наборів нерівномірно розподілених даних, у тому числі атак нульового дня, що становлять найбільшу загрозу для складових мобільного середовища. Обмежене застосування зазначеного підходу на апаратній платформі мобільного пристрою можливе лише за умови спрощеної конфігурації моделі.

До категорії моделей, що при обробці набору вхідних даних вимагають делегування на запитів сервер або обчислювальний кластер хмарної інфраструктури, належать [7; 18; 29]:

1. Нейромережеві алгоритми глибинного навчання, як то моделі на основі DNN, RNN, BERT і трансформери, призначені для виявлення складних нелінійних залежностей у запитах, обробки послідовностей, семантичного аналізу тіла запиту, а також інтеграції з SIEM-системами. Висока обчислювальна складність передбачає застосування хмарних сервісів або гібридних архітектур із попередньою фільтрацією.

2. Контекстно-адаптивні трансформери розглядаються як перспективні в задачах класифікації запитів із прихованими залежностями, але потребують значного обсягу оперативної пам'яті й засобів паралельної обробки, що доступні у рамках хмарної інфраструктури.

Розподіл між локальним виконанням та хмарною обробкою формується на основі таких критеріїв:

– складність та розмірність ознакового простору;
– наявність часової залежності у наборі даних;
– критичність до затримки у відповіді на загрозу;
– обчислювальні та енергетичні обмеження мобільного пристрою.

З метою формалізації підходу до вибору алгоритмів машинного навчання залежно від обчислювального середовища проведено оцінювання їхньої придатності до виконання на мобільному пристрої та у хмарній інфраструктурі. Результати систематизації наведено в (табл. 1), яка узагальнює типові моделі кіберзахисту API, класифікує їх за параметрами можливого розміщення, а також окреслює характерні особливості реалізації кожної моделі з урахуванням вимог до показників ресурсомісткості.

Таблиця 1

Оцінка придатності моделей до виконання у локальному середовищі мобільного пристрою та інфраструктурі хмарного сервісу [5; 7; 12; 13; 14; 18; 23; 29]

Тип моделі кіберзахисту	Пам'ять	Швидкодія і затримка	Енергоспоживання	Особливості використання
Метод опорних векторів	низьке	висока	низьке	Найбільш ефективно на етапі бінарної класифікації запитів у локальному середовищі.
Ансамблеві моделі	середнє	середня	середнє-високе	Забезпечують стійкість до перенавчання і при цьому придатні як для локального виконання, так і для хмарної аналітики у складніших конфігураціях.
Градієнтні дерева	середнє	середня	середнє	Дає хороші результати на нерівномірно розподілених даних, використовуються у мобільному середовищі у спрощеній формі, а також масштабуються для хмарної інфраструктури.
Глибинні нейромережі (DNN / RNN)	високе	низька	високе	Необхідні великі обсяги даних і GPU / TPU, тому найбільш ефективні у хмарній інфраструктурі.
Трансформери та BERT	дуже високе	низька	дуже високе	Забезпечують контекстний аналіз, потребують спеціалізованих прискорювачів (GPU / NPU) і достатніх обчислювальних ресурсів.

Таким чином, побудова ефективної системи захисту API вимагає формування гібридної архітектури [12; 13], що поєднує переваги обох типів моделей, де локальні модулі використовуються для оперативного реагування, а хмарні сервісу виконують задачі глибинного аналізу та виявлення складних загроз.

Оптимізація моделей машинного навчання для мобільного середовища. Як було зазначено вище, забезпечення безперервного моніторингу та виявлення загроз у мобільному середовищі потребує використання моделей машинного навчання, які можуть функціонувати в умовах обмежених обчислювальних ресурсів, мінімального енергоспоживання та нестабільної пропускної здатності каналів зв'язку. У цьому контексті перспективним напрямом є впровадження легковагових нейромережевих архітектур та методів оптимізації моделей, таких як квантизація моделі,

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

проріджування моделі і дистиляція знань. На загальному рівні можна виокремити дві архітектурні концепції впровадження легковагових нейромережевих систем:

1. Концепція TinyML передбачає виконання інтелектуальних обчислень безпосередньо на пристроях із обмеженими апаратними ресурсами, як то на мікроконтролерах із мінімальним обсягом оперативної пам'яті та низьким енергоспоживанням [24]. Відповідний підхід застосовується у задачах попереднього виявлення аномалій у потоках сенсорних даних, класифікації коротких послідовностей API-запитів, а також у попередній обробці даних безпосередньо на мобільному пристрої перед їх передачею у хмарне середовище.

2. Архітектурна модель «Edge AI» орієнтована на реалізацію гібридних рішень, у яких первинна обробка, фільтрація та виявлення аномалій відбувається локально (на рівні мобільного пристрою або периферійного вузла), тоді як більш складна аналітика, пов'язана з нейромережевим алгоритмом глибокого навчання, делегується до хмарних сервісів [3]. Такий підхід дозволяє зменшити затримки при реагуванні, знизити навантаження на канали зв'язку та забезпечити адаптивність до контексту середовища.

Таблиця 2

Порівняльна характеристика методів оптимізації моделей машинного навчання для мобільного середовища [3; 24]

Метод оптимізації	Цільовий ефект	Вплив на точність	Повторне навчання
дистиляція знань	менша складність	низький	потребує
квантування моделі	менший розмір	помірний	не обов'язково
проріджения моделі	менше параметрів	помірно високий	потребує
архітектурна оптимізація	проста структура	помірно низький	не потребує

У свою чергу, методологічні рішення щодо оптимізації моделей ґрунтуються на наступних підходах:

– процедура дистиляції знань передбачає тренування спрощеної студент-моделі на основі прогнозів попередньо навченої великої моделі-наставника, що дозволяє зберегти високу якість при суттєвому зменшенні обчислювального навантаження;

– процедура квантування моделі передбачає зменшення точності числового представлення параметрів моделі, що зменшує обсяг оперативної пам'яті, що використовується алгоритмом, а також пришвидшує обробку набору вхідних даних;

– процедура проріджения моделі полягає у видаленні найменш значимих параметрів а також нейронів, що дозволяє зменшити загальний розмір нейромережевої архітектури без суттєвого зниження якості обробки набору вхідних даних.

Ці методи дозволяють реалізовувати адаптивну конфігурацію моделей відповідно до характеристик апаратної платформи мобільного пристрою. Для формалізації вибору конкретної техніки оптимізації у заданих обмеженнях запропоновано класифікацію (див. табл. 2), що враховує цільовий ефект, рівень втрати точності, а також необхідність повторного навчання.

Архітектурна модель гібридного захисту API мобільного додатку. Забезпечення ефективного та стійкого захисту API мобільних додатків, таким чином, вимагає впровадження багаторівневої моделі безпеки, яка охоплює всі критичні етапи обробки та взаємодії даних. Така модель має бути не лише комплексною за структурою, а й адаптивною до характеристик мобільного середовища, зокрема обмежених обчислювальних ресурсів, нестабільної пропускної здатності каналів зв'язку, енергоспоживання та типу платформи. У межах запропонованої архітектурної моделі гібридного захисту API доцільно виокремити такі ключові рівні:

1. Аутентифікація та авторизація, що реалізується переважно локально з використанням токенів, біометричних даних та системних засобів ідентифікації. При цьому критично важливо забезпечити швидкий доступ без передачі чутливих даних через мережу.

2. Шифрування трафіку здійснюється через протоколи TLS / SSL, з можливістю адаптивного вибору криптографічних параметрів залежно від поточних параметрів апаратної платформи мобільного пристрою.

3. Захист інформаційного сховища даних через шифрування на рівні файлової системи або використання захищених контейнерів.

4. Фільтрація API-запитів засобами базової евристичної перевірки та легковагових класифікаторів, що реалізуються локально, у той час як глибока перевірка виконується на сервері або через хмарні SIEM-системи.

5. Контейнеризація та ізоляція через впровадження ізольованого середовища виконання, що обмежує вплив потенційно скомпрометованих модулів.

6. Моніторинг та виявлення загроз реалізується у гібридному режимі, де первинний контроль поведінки та виявлення відхилень виконується локально, а централізований аналіз проводиться у середовищі хмарного сервісу із залученням нейромережевих моделей.

7. Тестування та оновлення включають у себе перевірку цілісності компонентів і своєчасне отримання оновлень є критично важливими для протидії новим вразливостям.

Адаптивність цієї моделі досягається за рахунок динамічного розподілу функцій між локальним середовищем і хмарною інфраструктурою, що дозволяє зменшити затримку у реагуванні на загрози, забезпечити функціонування навіть при тимчасовій втраті мережевого підключення та підвищити масштабованість і здатність до самооновлення моделей. Таким чином, побудова комплексної моделі гібридного захисту дозволяє сформувати баланс між ефективністю, гнучкістю та продуктивністю, що є критичним у контексті динамічно змінюваних умов мобільного середовища та зростаючої складності кібер-атак на API-додатки.

Висновки. У результаті проведеного дослідження було проаналізовано особливості захисту API мобільних додатків в умовах обмеженої обчислювальної інфраструктури. Розроблено комплексну методику, що враховує апаратні характеристики мобільних пристроїв, сценарії розміщення компонентів безпеки, а також адаптивне використання моделей машинного навчання відповідно до доступних ресурсів. Розглянуто основні стратегії розміщення інтелектуальних компонентів безпеки у рамках локального та хмарного виконання, включаючи можливості застосування легковагових архітектур і технік стиснення моделей. Запропоновано критерії вибору між локальним аналізом та делегуванням задач на хмарні платформи з урахуванням параметрів затримки, енергоспоживання, пропускної здатності та рівня загроз. Сформовано архітектурну модель гібридного захисту API, яка передбачає багаторівневу систему, що складається з процедур аутентифікації, авторизації, шифрування, моніторингу, фільтрації запитів, а також модульну інтеграцію алгоритмів виявлення загроз.

Таким чином, результати дослідження можуть бути покладені в основу створення адаптивних систем безпеки API мобільних додатків, здатних до ефективної роботи навіть в умовах обмежених ресурсів, забезпечуючи при цьому гнучкість і масштабованість відповідно до змін середовища та характеристик загроз.

Список використаних джерел:

1. Acosta-Prado J. C., Rojas J. Rincón S., Mejía A. Martínez M., Riveros A. Tarazona R. Trends in the literature about the adoption of digital banking in emerging economies: A bibliometric analysis. *Journal of Risk and Financial Management*. 2024. No 17(12). DOI: https://doi.org/10.3390/jrfm17120545

2. Alshamrani A., Myneni S., Chowdhary A., Huang D. A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities. *IEEE Communications Surveys & Tutorials*. 2019. No 21(2). P. 1851–1877. DOI: https://doi.org/10.1109/COMST.2018.2869441

3. Alzubaidi A., Kalutarage H., Wills G. B. Edge AI architectures for Internet of Things applications: A survey. *Smart Systems and Resilient Technologies*. 2023. No 5. DOI: https://doi.org/10.1016/j.ssrt.2023.100038

4. Beldachi R., Sallabi F., El Khatib H. Lightweight security solutions for resource-constrained mobile devices. *International Journal of Network Security & Its Applications (IJNSA)*. 2018. No 10(3). P. 11–25.

5. Dantas P. V., da Silva W. Jr S., Cordeiro L. C., Carvalho C. B. A comprehensive review of model compression techniques in machine learning. *Applied Intelligence*. 2024. Vol. 54. P. 11804–11844. DOI: https://doi.org/10.1007/s10489-024-05747-w

6. Enck W., Gilbert P., Chun B.-G., Cox L. P., Jung J., McDaniel P., Sheth Taint A. Droid: An information-flow tracking system for realtime privacy monitoring on smartphones. In: *Proceedings of the 9th USENIX Symposium on Operating Systems Design and Implementation (OSDI '10)*. Berkeley: USENIX Association, 2010. P. 1–16.

7. Gupta A., Lee S. Client-side versus server-side vulnerabilities in mobile APIs: A comparative study. *Journal of Systems Architecture*. 2021. Vol. 115. DOI: https://doi.org/10.1016/j.sysarc.2021.102061

8. Gupta P., Sandhu A. A review on API security challenges and solutions in modern web applications. *Journal of Network and Computer Applications*. 2023. Vol. 213. DOI: https://doi.org/10.1016/j.jnca.2022.103504

9. Haris N., Chen K., Song A., Pou B. Finding vulnerabilities in mobile application APIs: A modular programmatic approach. *arXiv preprint* : website. 2023. DOI: https://doi.org/10.48550/arXiv.2310.14137

10. Khan R., Othman M., Madani S. A., Khan S. U. A survey of mobile cloud computing application models. *IEEE Communications Surveys & Tutorials*. 2014. Vol. 16(1). P. 393–413. DOI: https://doi.org/10.1109/SURV.2013.052313.00134

11. Kumar A., Sethi N. Digital transformation trends in service industries: A systematic review. *International Journal of Service Science, Management, Engineering and Technology*. 2022. Vol. 13(1). P. 45–60.

12. Kumar P., Singh R. Mobile-Edge and Cloud-Based M. Hybrid L. Models for Secure API Ecosystems. *International Journal of Network Security*. 2021. No 23(4). P. 667–680. DOI: https://doi.org/10.6633/IJNS.202104_23(4).01

13. Li X., Zhao J. Edge-based versus cloud-based ML for real-time anomaly detection in mobile services. *ACM Transactions on Internet Technology*. 2019. No 19(1). DOI: https://doi.org/10.1145/3311699

14. Liu D., Zhu Y., Liu Z., Liu Y., Han C., Tian J., Li R., Yi W. A survey of model compression techniques: past, present, and future. *Frontiers in Robotics and AI*. 2025. No 12. DOI: https://doi.org/10.3389/frobt.2025.1518965

15. Liu D., Zhu Y., Zhang Z. et al. A survey of model compression techniques: past, present, and future. *Frontiers in Robotics and AI*. 2025. No 12.

16. Meddeb A. API security: Why it's more important than ever. *Computer Fraud & Security*. 2020. No 5. P. 8–11. DOI: https://doi.org/10.1016/S1361-3723(20)30057-7

17. OWASP Foundation. OWASP Top 10 API Security Risks – 2023. OWASP Foundation, 2023. 50 p.

18. Pal S., Misra S. Security challenges in mobile–cloud integrated systems: A survey. *IEEE Communications Surveys & Tutorials*. 2022. No 24(3). P. 1873–1897. DOI: https://doi.org/10.1109/COMST.2021.3124843

19. Paul C. Mobile app personalization using machine learning algorithms. *International Journal of Advanced Computer Science & Applications (IJACSA)*. 2023. No 14(7). P. 205–218.

20. Shostack A. Threat Modeling: Designing for Security. Hoboken: Wiley, 2014. 624 p.

21. Skosana S., Mlambo S., Madiope T., Thango B. Evaluating wireless network technologies (3G, 4G, 5G) and their infrastructure: A systematic review. *SSRN Electronic Journal*. 2024. https://doi.org/10.2139/ssrn.4992432

22. Souppaya M., Scarfone K. Guide to Data-Centric System Threat Modeling (NIST SP 800-154, Initial Public Draft). Gaithersburg: National Institute of Standards and Technology, 2016. 65 p.

23. Suwannaphong T., Jovan F., Craddock I., McConville R. Optimising TinyML with quantization and distillation of transformer and Mamba models for indoor localisation on edge devices. *arXiv preprint* : website. 2024. DOI: https://doi.org/10.48550/arXiv.2412.09289

24. Suwannaphong T., Jovan F., Craddock I., McConville R. Optimising TinyML with quantization and distillation of transformer and Mamba models for indoor localisation on edge devices. *Internet of Things and Cyber-Physical Systems*. 2024. No 4. DOI: https://doi.org/10.1016/j.iotcps.2023.100086

25. Teodorescu C. A., Durnoi A. N., Vargas V. M. The rise of the mobile Internet: Tracing the evolution of portable devices. *Proceedings of the International Conference on Business Excellence*. 2023. No 17(1). P. 1645–1654. DOI: https://doi.org/10.2478/picbe-2023-0147

26. World Health Organization, European Commission. Assessing the impact of digital transformation of health services. *Expert Panel Opinion. Luxembourg: Publications Office of the European Union*, 2019. 120 p.

27. Zhang C., Patras P. Long-term mobile traffic forecasting using deep spatio-temporal neural networks. *arXiv preprint* : website. 2017. URL: https://arxiv.org/abs/1712.08083 (last accessed: 18.09.2025).

28. Zhang H., Huang J. Challenging GPU dominance: When CPUs outperform for on-device LLM inference. *arXiv* : website. 2025. DOI: https://doi.org/10.48550/arXiv.2505.06461

29. Zhang Y., Wang L. Machine learning–driven API threat detection: Methods and opportunities. *Journal of Computer Security*. 2020. No 28(6). P. 773–795. DOI: https://doi.org/10.3233/JCS-200457

Дата надходження статті: 18.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

UDC 004.9
DOI https://doi.org/10.32689/maup.it.2025.3.8

Alla KAPITON
Doctor of Pedagogical Sciences,
Professor at the Department of Computer and Information Technologies and Systems,
Yuriy Kondratyuk Poltava Polytechnic National University,
kits_seminar@ukr.net
***ORCID:** 0000-0002-7845-0883*

Tamara FRANCHUK
Candidate of Economic Sciences,
Associate Professor at the Department of Software Engineering and Cybersecurity,
State University of Trade and Economics,
Tamara_Franchuk@ukr.net
***ORCID:** 0000-0001-7615-1276*

Dmytro TYSHCHENKO
Candidate of Economic Sciences, Associate Professor,
Associate Professor at the Department of Software Engineering and Cybersecurity,
State University of Trade and Economics,
tyshchenko_d@knute.edu.ua
***ORCID:** 0000-0002-2193-9012*

Alyona DESYATKO
Candidate of Technical Sciences, Associate Professor,
Head of the Department of Software Engineering and Cybersecurity,
State University of Trade and Economics,
desyatko@knute.edu.ua
***ORCID:** 0000-0002-2284-3418*

EVALUATION OF CRITERIA FOR THE APPLICATION OF MOBILE OPERATING SYSTEMS

***Abstract. The purpose** of the work is to evaluate the criteria for the application of mobile operating systems.*

***The methodology** used in the work consists in determining effective means of using mobile operating systems, as well as analyzing modern mobile operating systems, which are mediated by various system modules related to innovative projects in the development of cellular communication systems.*

***The scientific novelty** of the work lies in the identification and generalization of criteria for the application of mobile operating systems as key factors for improving the quality of development, implementation and use of mobile applications in the context of the development of a cloud environment.*

***Conclusions.** It has been established that modern operating systems differ in the specifics of the implementation of internal algorithms for managing the main resources of mobile devices, which include processors, memory, etc., which are determined by certain resources, mediated by modern design methods, types of hardware platforms, development and implementation environments, etc.*

It is determined that in the modern environment of active implementation of mobile operating systems, to optimize the performance of tasks, they are assigned a leading role, which provides an opportunity to increase the efficiency of specialists. The main advantages and disadvantages of the most widely used systems are investigated: Android, iOS, Windows Phone and BlackBerry OS.

The requirements for choosing the type of program according to the characteristics of the cell phone are substantiated, which concerns a certain type of program that adapts to the tasks set. The level of complexity of mobile operating systems used by means of network communication and data exchange, which have the peculiarities of their processing, is determined. The versions of mobile systems that have the ability to adapt to different operating systems developed for a mobile phone model are analyzed. This will allow specialists to use smartphones with innovative and effective programs. It has been determined that with the advent of touch-sensitive mobile devices, a number of tasks, in particular those for remote performance, become particularly relevant and require further analysis and research. It is proposed that, in order for the system to increase the efficiency of implementing the tasks set, applications be developed and used using programming languages based on Java and Visual Basic for mobile devices for further adaptation of the system to devices.

***Key words:** system software, digitalization, software product, operating systems*

Алла КАПІТОН, Тамара ФРАНЧУК, Дмитро ТИЩЕНКО, Альона ДЕСЯТКО. ОЦІНКА КРИТЕРІЇВ ЗАСТОСУВАННЯ МОБІЛЬНИХ ОПЕРАЦІЙНИХ СИСТЕМ

Анотація. *Метою роботи є оцінка критеріїв застосування мобільних операційних систем.*

Методологія, *використана в роботі, полягає у визначені ефективних засобів застосування мобільних операційних систем, а також аналіз сучасних мобільних операційних систем, що опосередковані різноманітними системними модулями, пов'язаними з інноваційними проектами в розробці систем стільникового зв'язку.*

Наукова новизна *роботи полягає у визначенні та узагальненні критеріїв застосування мобільних операційних систем як ключових факторів підвищення якості розробки, впровадження та використання мобільних доданків в умовах розвитку хмарного середовища.*

Висновки. *Встановлено, що сучасні операційні системи відрізняються особливостями реалізацій внутрішніх алгоритмів керування основними ресурсами мобільних пристроїв, до складу яких слід відносити процесори, пам'ять тощо, які обумовлені певними ресурсами, що опосередковані сучасними методами проектування, видами апаратних платформ, середовищем розробки та впровадження тощо.*

Визначено, що в сучасному середовищі активного впровадження мобільних операційних систем, для оптимізації виконання поставлених завдань на них покладено провідну роль, що надає можливість підвищити ефективність роботи фахівців. Досліджено основні переваги та недоліки найбільш відомих вживаних та актуальних сьогодні мобільних операційних систем.

Обґрунтовано вимоги вибору типу програми відповідно до характеристик стільникового телефону, що стосується певного типу програми, яка адаптується до поставлених завдань. Визначено рівень складності мобільних операційних систем, що використовуються засобами мережевого зв'язку та обміну даними, що мають особливості їх обробки. Проаналізовано версії мобільних систем, що мають можливість адаптуватися до різних операційних систем, розроблених для моделі мобільного телефону. Це дозволить фахівцям використовувати смартфони з інноваційними та ефективними програмами. Визначено, що з появою сенсорних мобільних пристроїв, низка завдань, зокрема для виконання віддалено набуває особливої актуальності та потребує подальшого аналізу та дослідження. Запропоновано, задля того, щоб система підвищила ефективність реалізації поставлених задач, розробляти та використовувати додатки за допомогою мов програмування, які базуються на Java та Visual Basic для мобільних пристроїв задля подальшої адаптації системи до пристроїв.

Ключові слова: *системне програмне забезпечення, цифровізація, програмний продукт, операційні системи.*

Introduction. Operating systems may differ in the features of the implementation of internal algorithms for managing the main resources of mobile devices (processors, devices, memory), features of the used design methods, types of hardware platforms, areas of use, and many other properties.

The OS controls the device, runs programs, provides data protection, performs various service functions for user and program requests. The OS includes the following groups of components: the core containing the scheduler; device drivers; network subsystem; file system; system libraries; hell with utilities. The presence of an operating system is the main feature that distinguishes a smartphone from an ordinary mobile phone. When choosing a specific smartphone or communicator model, the OS often becomes a determining factor.

The operating system Symbian OS (EPOC 32) was created by the Symbian company – a joint venture of Motorola, Ericsson, Nokia and Psion based on the Psion Software division of the Psion company. Symbian OS is a full-featured operating system, created taking into account all the requirements of the telecommunications industry and most modern standards and protocols, such as Bluetooth, GPRS, etc. The core of the system – multi-tasking, highly productive and extremely compact – can be transferred to almost any platform without great expense.

Full Unicode support allows you to easily adapt the system for any language, flexible extension mechanisms allow you to solve all problems with mail and Web encodings. Symbian OS is used for applications have the following common features: careful development of the user interface, with the aim of making the programs as easy to use as possible, regardless of the level of user training; standard graphic control elements implemented in the EIKON library, including a toolbar, toolbars, convenient control using a keyboard and/or pen; increase and decrease the scale of the image on the screen to adjust the image taking into account the type of program data, lighting conditions and the user's eyesight; support for printing to most standard printers, printing through a serial, parallel or infrared port, or to a printer connected to a desktop PC; support for embedded objects, which allows, for example, to embed Word documents in an Agenda record; data exchange between applications via a standard clipboard, data exchange with other devices via an infrared port; Companies such as Nokia, Sony Ericsson and some other smartphone manufacturers equip their smartphones only with Symbian OS. The main competitor of Symbian OS is the Microsoft Windows Mobile operating system.

Analysis of recent research and publications. Studying the current nuances of the domestic and global market, in particular the requirements for the use of mobile applications and their standards, it should be noted that the issues of development, use and implementation of modern mobile applications and the selection of appropriate mobile systems is relevant and requires a comprehensive and thorough analysis to study and offer recommendations for optimizing their selection, which contributed to the choice of the research topic and the identification of the tasks set in the publication.

Kryvoshy V., Gafiyak A. performed a comparative characteristic of the most famous operating systems Linux and Windows [1, 120–121]. Matyash S., Gafiyak A. analyze the general characteristics of operating systems presented on the world market [2, 122–123]. Franchuk T., Tyshchenko D., Desiatko A., Karpunin I. explore the features of digitalization processes using mobile devices in various industries [3, 61–66]. Kapiton A., Franchuk T., Tyshchenko D., Desiatko A., Sas N. analyze the requirements for mobile terms in the process of modeling objects and management processes [4, 37–41]. Kapiton A., Karpov A. study the features of implementing a fully functional system on mobile devices [5, 230–232]. Tyshchenko D., Franchuk T., Stepashkina K., Karpunin I. investigate the requirements for operating systems in order to enable effective design and development of information systems for a given purpose [13, 200–207]. Tyshchenko D., Franchuk T., Zakharov R., Moskalenko V. study the issues of supporting dynamic security needs [14, 149–152].

Features of using mobile operating systems, issues of developing OS on mobile devices, research of operating systems for mobile platforms, their varieties, as well as the evolution of the most popular mobile operating systems, such as Android, iOS and their competitors, are a relevant topic that is constantly being researched by leading scientists in the IT industry. The problems of implementing modern mobile operating systems and their widespread use in all industries are most often of interest to scientists from our country and other countries, the results of which are presented by them in numerous publications [6–12].

Main part. Mobile operating systems – these are a series of small programs or applications adapted to mobile phones to provide various functions that the user can use. Since the so-called smartphones appeared on the market, their reactivity among the population was important, so it began to be mass-produced all over the world. The first smartphones to revolutionize the smartphone world were BlackBerrys. Outdated, currently unsupported software platforms are presented on (Fig. 1). The most used mobile operating systems are presented on (Fig. 2).

Each company selects the type of program according to the characteristics of the cell phone. That is, depending on the smartphone model, a certain type of program is placed that adapts to the conditions of this phone. Mobile operating systems are simpler than those used in computers; a large percentage of them are wirelessly connected.

Data processed on mobile devices also comes in different formats, such as audio, photos and videos. Some phones do not include certain applications that are included in the software on some computers. In the case of the Android system, in most cases it lacks programs for working with documents, photo, video editor and other programs (Fig. 3.) [6–12].

Specialists must carefully select the programs that will be used on the various platforms that must support their functioning. The advantages of these operating systems include the ability to connect for data transport, which will increase the amount of memory, and therefore enable and enhance multi-functionality and

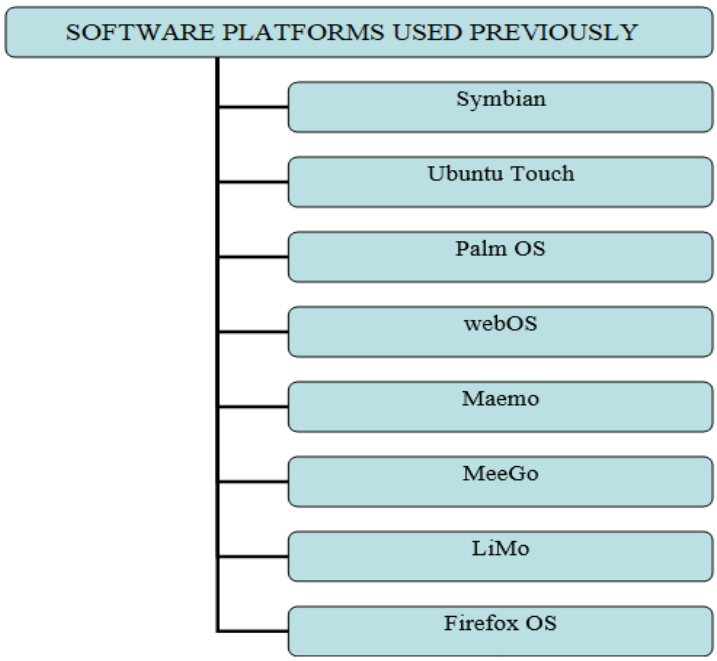


Fig. 1. Software platforms used previously

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

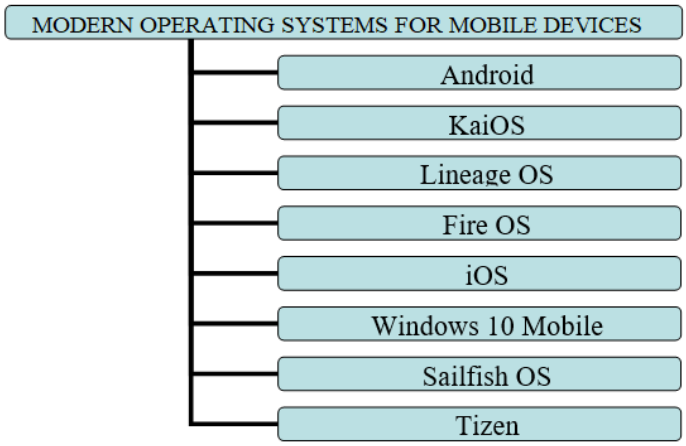


Fig. 2. Modern operating systems for mobile devices

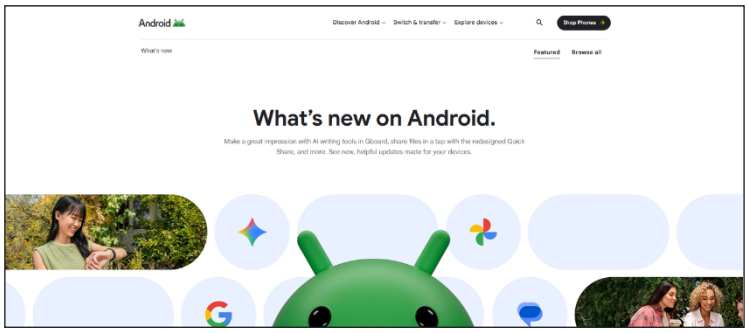


Fig. 3. Official website https://www.android.com/

multi-tasking, which requires constant updating and modification of bootable applications. Analyzing modern mobile operating systems, we can conclude that they are uniform and similar to each other. The development of new versions, subspecies, modifications proposed by specialists is based on their modular change of components, similar to RAM and computer software. These components constitute a complex of operations that individually perform different functions. So, an updated version of the mobile operating system is being built that performs a series of operations that launch a process during which a series of processes and resources, called modules, are activated. This allows you to optimize the tasks and increase the speed of their execution. Modules and commands determine the sequence of actions and functionality of RAM [6–12].

These types of devices had a variety of programs and modules that allowed people to do a variety of things that were done on computers just a few years ago. These devices work with Android and Windows systems. On the other hand, Apple developed the iOS operating system, which was used only for the company's devices. It is also innovative, which allowed it to become a benchmark for other operating systems that would be developed later. The interface was fast and convenient. Thus, the versions of mobile operating systems grew until we reached the ones that are currently on the global market. They offer customers optimal services where applications and operations are very diverse and serve as a tool for work and entertainment (Fig. 4) [6–12].

The software operates locally and does not require an Internet connection. This is its advantage. Such machines can be: office servers, which can be used for software with low resource requirements; high-power personal computers, which are actually used as servers. Larger businesses with extensive IT infrastructure use large server machines and a staff of specialists to maintain them as computing equipment. Cloud computing is the same servers, super-powerful computer machines. All hardware is maintained by technical specialists and programmers. Such clouds have many advantages for accounting programs, compared to local servers. Like any other software, management, accounting, or other accounting systems can be stored and administered locally or on remote servers – clouds.

With the help of the technologies available to us today, the creation of innovative tools that contribute to effective financial management has become possible. Based on the research chosen by the authors, its purpose is to analyze and improve the use of cloud technologies for accounting and optimize financial reporting in the process of qualitative modification of cloud services. Accounting application solutions require hardware

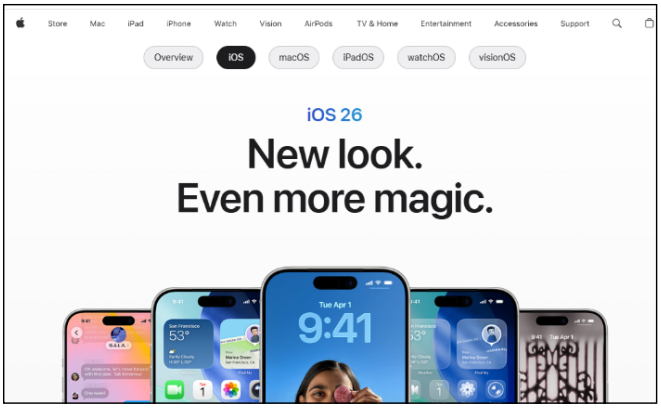


Fig. 4. Official website https://www.apple.com/os/ios/

for storing and processing information, while server capacities – computers – are used to host automated management and accounting systems. The server requires special placement conditions – a specially ventilated room, a staff of specialists to service it, etc.

Conclusions. The mobile application industry is rapidly developing and constantly provides new opportunities for the convenience and improvement of the quality of life of users. Mobile applications open up limitless horizons for us for communication, entertainment, work and many other areas. A mobile application is software specially designed for use on mobile devices such as smartphones and tablets. Mobile applications have become an integral part of modern digital life and play a key role in both everyday life and business.

Main characteristics of mobile applications: Platform dependency (Mobile applications are developed for specific mobile platforms, such as Android, iOS, Windows Phone and others. Each platform has its own technologies and development tools); User-friendly interface (Mobile applications have an intuitive and user-friendly interface designed for use on touch screens of mobile devices. They are optimized for small screens and use gestures and touch interface for interaction); Access to device features (Mobile apps can use various features of mobile devices such as camera, GPS, accelerometer, microphone and many others. This allows them to provide enhanced interactivity and personalization capabilities); Download and Installation (Users can download mobile apps from official app stores (such as Google Play for Android and App Store for iOS) and install them on their devices). Mobile applications have become an integral part of modern life, influencing various aspects of everyday life and entrepreneurship. The role of mobile applications is important and has a wide range of applications.

The following positive points should be highlighted: Simplification of routine tasks (Mobile applications allow you to simplify many routine tasks such as scheduling, travel planning, financial accounting and many others. They create an opportunity for more efficient management of time and resources); Communication and Social Networks (Mobile applications for social networks, messengers and communication tools allow people to stay connected even over long distances. They have become essential for messaging, communication and sharing multimedia content); Entertainment and education (Mobile applications provide access to a variety of games, multimedia content and educational resources. They contribute to both the entertainment and educational development of users).

In business it will be such moments as: Efficiency and Productivity (Mobile applications enable businesses to improve efficiency and productivity. They provide access to business tools that simplify record keeping, planning and analysis of activities); Customer Service (Mobile applications allow companies to improve customer service by providing a convenient way to interact with and order products and services. This helps increase customer loyalty); Marketing and Advertising (Mobile applications allow businesses to run marketing campaigns and promotions. They provide tools to promote your brand and attract new customers). Analytics and research: Mobile apps provide access to data and analytics that help businesses make informed decisions, as well as market research and competitor analysis. All these factors demonstrate the important role of mobile applications in both daily life and business, making them an important subject for further research and development.

Research and study of the main, most common mobile operating systems are constantly in the field of view of the requirements of modern development and rapid growth of gadgets, mediated by a rapid response to today's requirements. In addition to the above mobile operating systems, others exist, are being developed and improved, which have limited application in certain industries.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Bibliography:

1. Кривоший В., Гафіяк А. Порівняльна характеристика операційних систем Linux та Windows Тези 63-ї наукової конференції ПНТУ, 2011. Т. 2. 120–121.
2. Матяш С., Гафіяк А. Порівняльний аналіз загальних характеристик операційних систем, представлених на світовому ринку. Тези 63-ї наукової конференції ПНТУ, 2011. Т. 2. 122–123.
3. Franchuk T., Tyshchenko D., Desiatko A., Karpunin I. Features of accounting digitalization processes. *Galician economic journal,* 2025, vol. 95, No 1, pp. 61–66.
4. Kapiton A., Franchuk T., Tyshchenko D., Desiatko A., Sas N. Modeling of management objects and processes *Системи управління, навігації та зв'язку.* 2025. Т. 1 (79). 37–41.
5. Kapiton A., Karpov A. Features of implementation of the full-functional ERP-system. *Сучасні комп'ютерні системи та мережі в управлінні*, 2021. 232.
6. Mobile operating systems URL: https://eir.zp.edu.ua/server/api (дата звернення: 10 вересня 2025).
7. Mobile operating systems. Development of OS on mobile devices URL: https://elartu.tntu.edu.ua/bitstream/123456789/6599/2/FOSSLviv_2013_Kurdaiev_O_S-Mobile_operating_systems_89-91.pdf (дата звернення: 17 вересня 2025)
8. Operating systems for mobile platforms URL: https://e-tk.lntu.edu.ua/mod/page/view.php?id=3776 (дата звернення: 10 вересня 2025)
9. Operating systems, their varieties URL: https://ua5.org/opersys/2117-operaczijni-systemy-yihni-riznovydy.html (дата звернення: 10 вересня 2025)
10. Smartphone operating systems URL: https://ua5.org/opersys/2342-operaczijni-systemy-smartfoniv.html (дата звернення: 20 серпня 2025).
11. Smartphone operating systems: what are they and what is the difference? URL: https://promin.cv.ua/ (дата звернення: 20 серпня 2025).
12. The Evolution of Mobile Operating Systems – Android, iOS and Their Competitors – History of Development URL: https://mediacom.com.ua (дата звернення: 10 вересня 2025)
13. Tyshchenko D., Franchuk T., Stepashkina K., Karpunin, I. Проектування та розробка системи корпоративного електронного документообігу. *Європейський науковий журнал Економічних та Фінансових інновацій*. 2024. № 1(13). 200–207.
14. Tyshchenko D., Franchuk T., Zakharov R., Moskalenko V. Підтримка динамічних потреб безпеки засобами VPN. *Системи управління, навігації та зв'язку.* 2024. Т. 3 (77). 149–152.

Дата надходження статті: 18.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.8:004.056:339.1
DOI https://doi.org/10.32689/maup.it.2025.3.9

В'ячеслав КОВАЛЕВСЬКИЙ
аспірант кафедри інженерії програмного забезпечення,
Державний університет «Житомирська політехніка»,
s.kovalevskyi@gmail.com
***ORCID:** 0000-0001-7144-1899*

Тетяна ВАКАЛЮК
доктор педагогічних наук, професор,
завідувач кафедри інженерії програмного забезпечення,
Державний університет «Житомирська політехніка»,
tetianavakaliuk@gmail.com
***ORCID:** 0000-0001-6825-4697*

ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ У СИСТЕМАХ ЗАХИСТУ СЕРВІСІВ ЕЛЕКТРОННОЇ КОМЕРЦІЇ

Анотація. *У статті розглянуто підходи на основі штучного інтелекту для вдосконалення існуючих систем безпеки сервісів електронної комерції. Разом з постійно зростаючою кількістю транзакцій, що проходять через платформи електронної комерції, розширюється і поле можливостей для атак зловмисників, які дедалі частіше використовують і вдосконалюють автоматизовані системи, що імітують поведінкові шаблони справжніх користувачів. Традиційні системи безпеки з використанням сигнатур та статичними механізмами обмежень погано адаптуються до динамічної зміни тактик зловмисників та потребують подальшого розвитку. Сучасні системи захисту, які використовують системи штучного інтелекту пропонують перехід до неперервного аналізу потоків подій у реальному часі. Це дозволяє детально відстежувати взаємодію користувачів з системою та вчасно реагувати на загрози. Лідери індустрії постійно працюють над розвитком цього напрямку та задають тенденції, що стають стандартами реалізації безпекової складової сервісів електронної комерції. В статті також окреслено переваги поведінкової біометрії, яка дає можливість моделювати індивідуальні особливості користувачів та вибудовувати стійкі профілі для точного відокремлення легітимних сесій від небажаних. Використання методів безперервного машинного навчання та аналізу підвищує швидкість виявлення аномалій серед потоків подій, що відбуваються в системі.*

Методологія. *У статті проведено аналіз сучасних методів використання систем штучного інтелекту, що можуть бути використані для побудови багаторівневої архітектури системи безпеки.*

Наукова новизна. *У роботі узагальнено сучасні методи використання систем штучного інтелекту для протидії актуальним безпековим викликам, що постають перед системами захисту сервісів електронної комерції.*

Висновки. *Інтеграція технологій штучного інтелекту з вже існуючими системами захисту сервісів електронної комерції значно розширює їх можливості адаптуватись до сучасних загроз. Застосування штучного інтелекту з підтримкою моделей неперервного аналізу даних створює збалансовану стратегію захисту сервісів електронної комерції, підвищує точність виявлення небажаної активності, знижує фінансові втрати постачальників послуг та зміцнює довіру кінцевих користувачів до сервісів електронної комерції.*

Ключові слова: *штучний інтелект, електронна комерція, інформаційна безпека, поведінкова біометрія, захист, моделювання.*

Viacheslav KOVALEVSKYI, Tetiana VAKALIUK. USE OF ARTIFICIAL INTELLIGENCE IN SECURITY SYSTEMS OF E-COMMERCE SERVICES

Abstract. *The article considers artificial intelligence based approaches for improving existing security systems of e-commerce services. Along with the constantly growing number of transactions passing through e-commerce platforms, the field of opportunities for malicious attacks also expands, as adversaries more and more often use and refine automated systems that imitate the behavioral patterns of real users. Traditional security solutions that rely on signatures and static threshold mechanisms adapt poorly to the dynamic change of attackers tactics and therefore require further development. Modern protection systems that employ artificial intelligence suggest a move from static checks to continuous, real-time analysis of event streams. This approach makes it possible to observe user interactions with system in detail and to respond to threats in a timely manner. Industry leaders continuously develop this direction and set tendencies that later become common standards for implementing the security component of e-commerce services. The article also outlines the advantages of behavioral biometrics, which allows modeling individual user characteristics and building stable profiles for accurately distinguishing legitimate sessions from unwanted ones. The use of continuous machine-learning and analytical methods increases the speed of anomaly detection within the event flows occurring in the system.*

Methodology. *The article presents an analysis of modern methods of using artificial intelligence systems that can be applied to the development of a multi-layered security system architecture.*

Scientific novelty. *The paper summarizes modern methods of applying artificial intelligence systems to counter current security challenges faced by the protection systems of e-commerce services.*

Conclusions. *The integration of artificial intelligence technologies with existing security systems of e-commerce services significantly expands their ability to adapt to modern threats. Applying artificial intelligence with the support of continuous data-analysis models creates a balanced strategy for protecting e-commerce services, improves the accuracy of detecting unwanted activity, reduces financial losses for service providers, and strengthens the trust of end users in e-commerce services.*

Key words: *artificial intelligence, e-commerce, information security, behavioral biometrics, protection, modeling.*

Постановка проблеми. Постійне збільшення об'ємів використання сервісів електронної комерції супроводжується невід'ємною еволюцією методів шахрайства із залученням прийомів соціальної інженерії, мереж ботів та підробки шаблонів поведінки користувачів, що в свою чергу призводить до малоефективної роботи систем захисту які базуються на використані обмежень порогових значень тих чи інших подій. Для сучасних систем захисту сервісів електронної комерції є необхідним перехід від статичного аналізу подій до гнучкої, неперервної обробки потоків даних, що виникають під час роботи системи. Важливою складовою такого підходу має бути здатність системи захисту швидко виявляти аномалії та відокремлювати сесії користувачів та інший небажаний трафік з високою точністю. Системи штучного інтелекту відіграють суттєву роль у цьому процесі.

Аналіз останніх досліджень і публікацій. Проблема захисту інформаційних систем та безпосередньо сервісів електронної комерції, не втрачає своєї актуальності, та навпаки потребує постійного вивчення і розвитку. Олег Колодізєв, у своїй роботі пропонує використання алгоритмів машинного навчання для виявлення шахрайських дій [3]. Шіні Ренджит (Shini Renjith) описує використання історичних даних користувачів для побудови поведінкових патернів [5]. Дослідники Чанг Жанг (Chang Zhang), Ючен Жанг (Yuchen Zhang) та Фулін Лі (Fullin Li) пропонують метод побудови унікальних характеристик користувача на базі аналізу його шаблонів набору тексту [10]. Алок Джаккула (Alok Jakkula) у своїй статті проводить аналіз загроз які можуть супроводжувати впровадження систем штучного інтелекту у процеси роботи систем електронної комерції. Автор зазначає, що незважаючи на всі переваги та покращення отримані при використанні систем штучного інтелекту, їх інтеграція з системами електронної комерції створює додаткові ризики витоку чутливих даних та вимагає всебічної оцінки безпекових аспектів [6]. Кожен з дослідників аналізує цю проблему, спираючись на власний досвід та специфіку тієї галузі, у якій він працює. Проведений аналіз літератури та публікацій свідчить, що проблема використання систем штучного інтелекту для побудови систем захисту сервісів електронної комерції є багатогранною та потребує подальших досліджень направлених на поєднання різних методів використання систем штучного інтелекту і створення прикладних рішень, що враховуватимуть технологічні та організаційні особливості роботи сервісів електронної комерції.

Метою даної статті є огляд актуальних методів залучення систем штучного інтелекту у процеси моніторингу та моделювання безпекових загроз при побудові систем захисту сервісів електронної комерції та інформаційних систем загалом.

Виклад основного матеріалу. Сервіси електронної комерції стали невід'ємною частиною сучасного світу та глобальної економіки. Вони забезпечують швидкий обіг товарів і послуг, зручність для індивідуальних користувачів, а також гнучкість для розширення і ведення бізнесу загалом. Постійне зростання кількості онлайн транзакцій супроводжується збільшенням нелегітимної діяльності, де особливо гостро стоїть проблема шахрайства та несанкціонованого доступу до персональних та фінансових даних користувачів [2]. За даними компанії Mastercard втрати від шахрайської діяльності у сфері електронної комерції перевищують 40 мільярдів доларів [8]. У своєму щорічному звіті присвяченому використанню платіжних систем у сфері електронної комерції компанія Visa зазначає відчутне зростання кількості шахрайських дій, що спричиняють фінансові збитки, підривають довіру споживачів, а також спонукають компанії витрачати значні ресурси на боротьбу з діями шахраїв. Автори звіту оцінюють втрати компаній у 3% річного доходу [9].

Класичні методи захисту, що використовують статичні правила та ручний моніторинг, все частіше проявляють недостатню ефективність коли стикаються зі складними та швидко еволюціонуючими шахрайськими схемами. Подібні виклики збільшують потребу у використанні технологій штучного інтелекту, які, у режимі реального часу, здатні проводити аналіз великих обсягів даних, виявляти та реагувати на загрози [2].

Впровадження та використання систем штучного інтелекту у якості складової системи захисту сервісів електронної комерції розширює можливості для автоматизації процесів захисту, підвищує точності виявлення аномалій, збільшує швидкість реагування та адаптації до нових видів атак. Використання машинного навчання, поведінкового аналізу дій користувачів та постійне вдосконалення методів використання систем штучного інтелекту у системах захисту сервісів електронної комерції забезпечують проактивний, гнучкий та ефективний захист.

Таким чином можна стверджувати, що застосування систем штучного інтелекту для забезпечення захисту сервісів електронної комерції є не лише сучасною тенденцією, а й необхідною умовою збереження довіри користувачів, підтримки фінансової стабільності бізнесу та готовності до новітніх безпекових викликів [1].

Поєднання системи захисту сервісів електронної комерції з системами машинного навчання надає можливості з побудови моделей, що використовуються для аналізу історичних даних про транзакції та визначенні потенційно небезпечних або небажаних операцій. Виявлення аномалій у поведінці користувачів та обробка великих обсягів даних про транзакції відбувається у режимі реального часу з подальшим використанням отриманих результатів для вдосконалення механізмів захисту, що підвищує загальний рівень захищеності сервісів електронної комерції.

Дослідник Олег Колодізєв, у своїй роботі, присвяченій алгоритмам автоматизованого машинного навчання для виявлення шахрайських дій у цифрових платіжних системах, демонструє, що впровадження подібних систем має значні перспективи для підвищення точності визначення недоброчесних операцій [3].

Пропонується використовувати алгоритми з використанням Баєсівської оптимізації та генетичних алгоритмів для автоматизації вибору архітектури моделі, гіперпараметрів для налаштування моделі та визначення найважливіших змінних з даних. Наступним кроком є створення ансамблів моделей для підвищення точності та надійності обробки даних.

У результаті проведених експериментів із використанням алгоритмів автоматизованого навчання, автор прийшов до висновку, що використані алгоритми автоматизації дозволяють за короткий час розглянути велику кількість варіантів моделей та складу вхідних даних. Побудований ансамбль моделей дозволив виявити до 85.7% шахрайських транзакцій. Водночас точність виявлення шахрайських транзакцій перебувала у діапазоні 79–85% [3].

Серед сучасних загроз сервісам електронної комерції окремо виділяють створення і використання підробних акаунтів користувачів. Дане явище стає значним безпековим ризиком, оскільки фейкові профілі користувачів часто використовуються для проведення небажаних чи відверто шахрайських операцій, маніпуляцій з бонусними програмами, а також для атак на інфраструктуру компаній. Підробні акаунти активно використовують для маніпуляцій з рейтингами та відгуками, спотворення результатів пошуку та рекомендаційних систем.

Під час пікових періодів бот-мережі створюють тисячі акаунтів для скуповування популярних товарів, що призводить до дефіциту для реальних покупців. У звіті компанії Radware, що є постачальником продуктів для кібербезпеки, вказується, що у сезон святкових розпродажів 2024 року понад 57% трафіку сервісів електронної комерції генерували саме боти [4].

Одним з дієвих підходів для виявлення шахрайських акаунтів користувачів сервісів електронної комерції є впровадження алгоритмів, що використовують метод опорних векторів. Використання цього підходу описує у своєму дослідженні Шіні Ренджит (Shini Renjith) [5]. Дослідник пропонує побудову моделі машинного навчання, що буде проводити класифікацію користувачів аналізуючи історичні дані сервісів електронної комерції, зокрема поведінкові патерни користувачів, транзакційні аномалії, характеристики користувацьких профілів, що були зареєстровані в системі (рис. 1).

Досліджуючи подібну проблему, Рауль Деко (Raoul Dekou) пропонує об'єднане використання декількох моделей машинного навчання для отримання кращої продуктивності та точності результатів [7]. Об'єднана модель продемонструвала F1-міру зі значенням 0.73, що виявилось найвищим показником серед протестованих моделей (табл. 1).

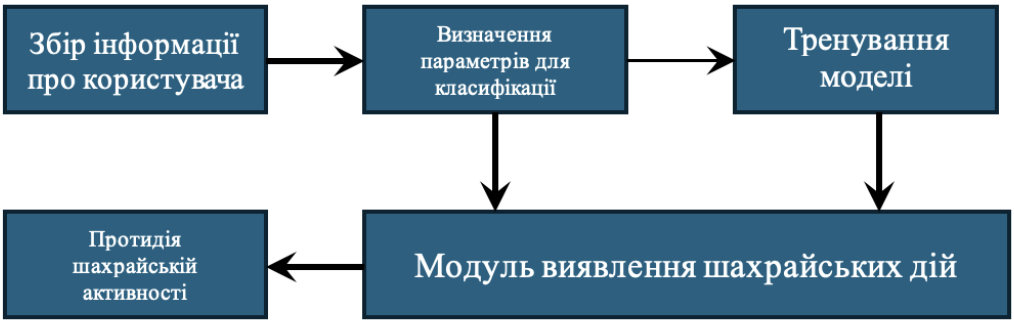


Рис. 1. Запропонований фреймворк для виявлення шахрайських акаунтів [5]

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Таблиця 1

Порівняльна таблиця продуктивності протестованих моделей машинного навчання [7]

Модель	F1	Точність	Повнота	AUC
AutoML	0.7293	0.7206	0.7833	0.9850
Xgb	0.7134	0.7104	0.7165	0.9794
Catboost	0.7127	0.7375	0.6895	0.9809
RF	0.6810	0.7274	0.6401	0.9786

Незважаючи на високу точність визначення шахрайських акаунтів, ця проблема залишається актуальною та вимагає постійного оновлення моделей машинного навчання. Окрім цього присутня загроза помилкового блокування легальних користувачів, особливо новостворених, де відсутня довга історія використання.

Окремо можна виділити перспективний напрямок з використання систем штучного інтелекту для впровадження поведінкової біометрії з безперервною аутентифікацією користувача. Це є суттєвим вдосконаленням систем захисту сервісів електронної комерції.

Концепція даного підходу полягає у неперервній оцінці унікальних шаблонів поведінки користувача та оцінки рівня довіри до нього. До унікальних поведінкових шаблонів користувача можна віднести характер рухів миші, динаміку набору тексту, швидкість переходів між сторінками тощо. Ці параметри можуть бути поєднані з біометричними даними, такими як розпізнавання обличчя, голосу, відбитків пальців. Таким чином створюється унікальний профіль користувача, що значно спрощує виявлення аномалій у його діяльності.

У контексті використання поведінкової біометрії для ідентифікації користувачів дослідники Чанг Жанг (Chang Zhang), Ючен Жанг (Yuchen Zhang) та Фулін Лі (Fullin Li) пропонують метод побудови унікальних характеристик користувача на базі аналізу його шаблонів набору тексту [10].

Автори використовують мультиваріативний процес Хоукса (Multivariate Hawkes Process) з експоненційним ядром для аналізу послідовностей натискання клавіш користувачем. Кожне натискання клавіші розглядається як подія у часовому просторі, де інтенсивність подій залежить від попередніх взаємодій.

У своєму дослідженні для визначення процесу Хоукса автори обрали метод визначення через умовну функцію інтенсивності.

$$\lambda_i(t) := \mu_i(t) + \sum_{j=1}^{D} \int_{-\infty}^{t} g_{ij}(t-\tau) dN_j(\tau) \quad (1)$$

Формула 1. Опис функції інтенсивності визначення процесу Хоукса [10].

Описана вище формула (формула 1) базується на припущенні, що багатовимірний лічильний процес є $N(t) = \{N_i(t)\}^{D}_{i=1}$, де $N(0)=0$, його розмірність дорівнює D, умовна функція інтенсивності має вигляд $\lambda(t) = \{\lambda_i(t)\}^{D}_{i=1}$, μ_i – константа, що визначає інтенсивність виникнення подій типу i. $G(t) = [gij(t)]^{D}i,j=1$ є DxD-вимірною матрицею ядра збудження, а функція збудження $g^{ij} \geq 0$ описує стимул подій, що відбулися в поточному *j*-му вимірі багатовимірного процесу Хоукса на інтенсивність події *i*-го виміру [10].

Дослідники роблять висновки, що процес Хоукса доцільно застосовувати для статистичного опису часових послідовностей натискання клавіш, оскільки його умовна інтенсивність $\lambda(t)$ відтворює самозбудження цієї активності – ймовірність настання події зростає одразу після попередньої та експоненціально згасає з плином часу. Працюючи неперервно, модель зберігає інформаційний зміст інтервалів між подіями без необхідності їх дискретизації, що критично важливо для виявлення відхилень на коротких часових проміжках [10]. Використання процесу Хоукса поєднує точність, інтерпретованість та обчислювальну ефективність, що робить його корисним інструментом для підвищення безпекових параметрів сервісів електронної комерції.

Незважаючи на всі переваги використання систем штучного інтелекту у системах захисту сервісів електронної комерції, слід зазначити, що їх інтеграція у робочі процеси сервісів електронної комерції потребує виваженого аналізу та додаткових перевірок на дотримання вимог безпечної обробки даних. Дослідник Алок Джаккула (Alok Jakkula) у своїй статті підкреслює критичну важливість застосування надійних заходів безпеки при залученні систем штучного інтелекту для опрацювання даних, що використовуються системами електронної комерції, які мають бути поєднані з чіткими та прозорими правилами контролю їх виконання [6].

Висновки. Сучасний захист систем електронної комерції потребує адаптивних рішень, що в режимі реального часу здатні проводити аналіз подій та особливостей поведінки користувачів. Використання систем штучного інтелекту робить це можливим та значно розширює можливості систем захисту сервісів електронної комерції. Застосування такого підходу збільшує точність виявлення аномалій, знижує частку хибних спрацювань і, завдяки прозорій побудові моделей, полегшує аудит рішень прийнятих системою захисту та дотримання нормативних вимог. В свою чергу інтеграція поведінкової біометрії з контекстною оцінкою транзакційних ризиків та мережевим моніторингом формує масштабовану й економічно доцільну багаторівневу систему захисту для сучасних систем електронної комерції. Інтеграція в системи захисту сервісів електронної комерції систем штучного інтелекту на всіх етапах повинна супроводжуватись попереднім аналізом безпекових ризиків та контролем їх роботи.

Список використаних джерел:

1. Музиченко Т. О., Скорба О. А., Шевчук А. А. Штучний інтелект як засіб оптимізації бізнес-процесів в електронній комерції. АКАДЕМІЧНІВІЗІЇ. 2023. № 23. URL: https://academy-vision.org/index.php/av/article/view/696/630.

2. Advancing E-Commerce Security: Strategic Innovations and Future Directions in AI and ML / L. A. A. Gracious et al. *IGI Global Scientific Publishing: International Academic Publisher*. URL: https://www.igi-global.com/chapter/advancing-e-commerce-security/356672.

3. Automatic machine learning algorithms for fraud detection in digital payment systems / O. Kolodiziev et al. *Eastern-European Journal of Enterprise Technologies*. 2020. Vol. 5, no. 9 (107). P. 14–26. URL: https://doi.org/10.15587/1729-4061.2020.212830.

4. Bots now dominate e-commerce traffic, warns Radware report – Intelligent CISO. *Intelligent CISO – Covering Security Across Borders*. URL: https://www.intelligentciso.com/2025/04/29/bots-now-dominate-e-commerce-traffic-warns-radware-report/.

5. Detection of Fraudulent Sellers in Online Marketplaces using Support Vector Machine Approach | Semantic Scholar. *Semantic Scholar*. URL: https://www.semanticscholar.org/reader/ac8972fe41c663b77b6dc99ea95d861ee56e06d2.

6. Jakkula A. R. Ensuring Data Privacy and Security in AI-Enabled E-commerce Platforms. *Journal of Artificial Intelligence & Cloud Computing*. 2024. Vol. 3, no. 1. P. 1–3. URL: https://doi.org/10.47363/jaicc/2024(3)288

7. Machine Learning Methods for Detecting Fraud in Online Marketplaces. / R. Dekou et al. *Conference: 2021 International Workshop on Privacy, Security, and Trust in Computational Intelligence*. URL: https://ceur-ws.org/Vol-3052/paper15.pdf.

8. Mastercard. Ecommerce fraud trends and statistics merchants need to know. *Payment and cybersecurity solutions*. URL: https://b2b.mastercard.com/news-and-insights/blog/ecommerce-fraud-trends-and-statistics-merchants-need-to-know-in-2024/.

9. The 2024 Global eCommerce Payments & Fraud Report. URL: https://www.cybersource.com/content/dam/documents/campaign/fraud-report/global-fraud-report-2024.pdf.

10. Zhang C., Zhang Y., Li F. Feature Extraction of Sequence of Keystrokes in Fixed Text Using the Multivariate Hawkes Process. *Mathematical Problems in Engineering*. 2021. Vol. 2021. P. 1–16. URL: https://doi.org/10.1155/2021/6648726.

Дата надходження статті: 23.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

UDC 004.75
DOI https://doi.org/10.32689/maup.it.2025.3.10

Dmytro KOVALCHUK
PhD in Computer Science, Senior Lecturer, V. N. Karazin Kharkiv National University,
kovalchuk.d.n@ukr.net
ORCID: *0000-0002-8229-836X*

THE CONCEPT OF BUILDING HIGH-PERFORMANCE REAL-TIME SYSTEMS USING THE RESIDUE NUMBER SYSTEM

Abstract. *The article examines the concept of building high-performance real-time information processing systems using the Residue Number System (RNS).*

The aim of the research *is to improve the performance and fault tolerance of modern computing systems through the application of a non-positional number system, which enables parallel data processing, dynamic error correction, and adaptive regulation of accuracy and computational speed. The work emphasizes the key properties of RNS – independence of residues, equality of residues, and small digit length – and their impact on the performance and reliability of real-time systems.*

The research methodology *is based on the principles of systems analysis, number theory, computational process theory, and systems modeling, as well as the simulation of modular arithmetic operations and error correction mechanisms. The study analyzes mathematical models of distributing informational and control residues to optimize the balance between speed, accuracy, and reliability of computations. The implementation methods of modular arithmetic are considered, including adder-based, table-based, logical, and circular-shift principles, which improve processing speed and allow single-cycle execution of computations in real time.*

The scientific novelty of the work *lies in the proposed methodology of dynamic redistribution of informational and control residues in RNS to ensure high fault tolerance and adaptability of the system. The study proposes the use of control residues to maintain operability even in the case of multiple computational path failures, as well as the application of small-digit modular arithmetic to enhance speed and reduce hardware complexity. It is shown that such a structure enables the simultaneous realization of three types of redundancy – structural, informational, and functional – which is critical for real-time systems.*

The conclusions of the study demonstrate that systems based on RNS provide significant acceleration of computations through parallel execution on independent computational paths and operand decomposition, increase reliability due to error localization, and enable dynamic regulation of accuracy and computational speed. The implementation of such systems makes them effective for processing large data sets, digital signal and image processing, cryptography, neurocomputing, and streaming computation tasks, ensuring continuous operation even in the case of partial component failures.

Key words: *Residue Number System, high-performance computing, real-time systems, fault tolerance, modular arithmetic, parallel computing.*

Дмитро КОВАЛЬЧУК. КОНЦЕПЦІЯ ПОБУДОВИ ВИСОКОПРОДУКТИВНИХ СИСТЕМ РЕАЛЬНОГО ЧАСУ З ВИКОРИСТАННЯМ СИСТЕМИ ЗАЛИШКОВИХ КЛАСІВ

Анотація. *У статті розглянуто концепцію побудови високопродуктивних систем обробки інформації в режимі реального часу з використанням системи залишкових класів (СЗК).*

Метою роботи *є підвищення продуктивності та відмовостійкості сучасних обчислювальних систем шляхом застосування непозиційної системи числення, яка дозволяє реалізувати паралельну обробку даних, динамічну корекцію помилок та адаптивне регулювання точності та швидкодії обчислень. У роботі зроблено акцент на властивостях СЗК – незалежності залишків, рівноправності та малорозрядності – та їхньому впливі на продуктивність і надійність систем реального часу.*

Методологія дослідження *базується на принципах системного аналізу, теорії чисел, теорії обчислювальних процесів і систем, а також на моделюванні модульних арифметичних операцій та механізмів корекції помилок. У роботі аналізуються математичні моделі розподілу інформаційних та контрольних залишків для оптимізації співвідношення між швидкодією, точністю та надійністю обчислень. Розглядаються способи реалізації модульної арифметики за допомогою суматорного, табличного, логічного та кільцевого принципів, що дозволяє підвищити швидкодію та забезпечити однотактне виконання обчислень у реальному часі.*

Наукова новизна роботи *полягає у запропонованій методології динамічного перерозподілу інформаційних і контрольних залишків у СЗК для забезпечення високої відмовостійкості та адаптивності системи. Запропоновано використання контрольних залишків для підтримки працездатності навіть при відмовах декількох обчислювальних трактів, а також застосування малорозрядної модульної арифметики для підвищення швидкодії та зниження апаратної складності. Показано, що така структура дозволяє реалізувати одночасно три типи резервування: структурне, інформаційне та функціональне, що критично для систем реального часу.*

Висновки дослідження демонструють, що системи на основі СЗК забезпечують значне прискорення обчислень за рахунок паралельного виконання на незалежних обчислювальних трактах та декомпозиції операндів, підвищують надійність завдяки локалізації помилок і забезпечують можливість динамічного регулювання точності та швидкодії обчислень. Реалізація таких систем робить їх ефективними для обробки великих масивів даних, цифрової обробки сигналів і зображень, криптографії, нейрокомп'ютерної обробки та задач потокових обчислень, гарантуючи безперервну роботу навіть у разі часткових відмов компонентів.

Ключові слова: *система залишкових класів, високопродуктивні обчислення, системи реального часу, відмовостійкість, модульна арифметика, паралельні обчислення.*

Introduction. The rapid development of digital technologies and the growing volumes of data to be processed impose new requirements on the architecture and performance of computing systems. In the period of 2020–2025, a key trend in computing has been the integration of artificial intelligence methods, Internet of Things (IoT) systems, and real-time stream data processing. Such systems must ensure high performance, scalability, and fault tolerance, since delays or errors in information processing can lead to critical consequences – from failures in industrial complexes to risks for the security of cyber-physical systems [10].

Traditional approaches to performance improvement rely on parallel and pipeline architectures, multicore processors, and algorithm optimization. However, they do not always allow effective problem-solving under the constraints of real time and increased reliability requirements [3]. This issue becomes particularly acute in areas where failure or error is unacceptable: in transportation systems, energy, defense, financial technologies, and healthcare [4].

One of the promising directions for building high-performance systems is the use of the Residue Number System (RNS) as the basis for arithmetic computations. The non-positional nature of RNS provides natural parallelism in executing operations, reduces the risk of error accumulation, and enables the development of fault-tolerant algorithms for digital information processing. In addition, the application of RNS makes it possible to optimize addition, multiplication, and comparison operations, which is especially important in real-time applications.

The relevance of the study lies in the need to develop new conceptual approaches to building high-performance real-time computing systems that combine speed, scalability, and fault tolerance. The use of RNS in this context opens opportunities for creating next-generation architectures oriented toward working with large streams of digital data and mission-critical computations.

The purpose of the article is to substantiate and develop the concept of building high-performance real-time systems using the Residue Number System, which will improve the efficiency of digital information processing and ensure a specified level of reliability and fault tolerance under modern challenges [1; 6; 7; 8].

Problem statement of the research. Modern information technologies are characterized by the rapid growth of data volumes that must be processed in real time. This concerns a wide range of domains – from industrial automation systems and traffic management to high-frequency financial transactions, cybersecurity, medical monitoring systems, and intelligent IoT devices. Under such conditions, the primary requirements for computing systems become high performance, scalability, and fault tolerance [1; 5].

Traditional methods of improving performance, such as expanding hardware resources, using multicore processors, pipeline architectures, or distributed computing, have largely exhausted their potential. Moreover, their application is often complicated by high energy consumption, increased hardware costs, and synchronization constraints of parallel data streams. For real-time systems, where even millisecond delays can have critical consequences, these drawbacks become decisive [2; 4; 5].

Another significant challenge is ensuring the reliability of computations. In cases where data processing is performed with minimal latency, even a minor error or failure can cause the entire system to malfunction. Traditional approaches to error control (such as computation duplication, error-correcting codes, or hardware redundancy) increase the complexity of the architecture and resource consumption, which negatively affects efficiency [2; 3].

In this context, the RNS offers an alternative approach to organizing arithmetic computations. Due to its non-positional nature, RNS enables natural parallelism: operations on numbers in different moduli are performed independently, which significantly increases performance. At the same time, this approach prevents intermediate result overflow and reduces the likelihood of error accumulation [4].

Accordingly, the research problem lies in creating a conceptual model and architecture of high-performance real-time systems based on RNS that will provide:

- increased speed of digital information processing through natural parallelism [4];
- guaranteed reliability and fault tolerance of computations;
- scalability for working with large data streams;
- integration with modern hardware and software platforms.

Solving this problem has important theoretical and practical significance for the development of next-generation computing systems capable of operating under the challenges of the digital economy and global digital infrastructure.

Literature Review. The literature review indicates that RNS are considered one of the most promising approaches for building high-performance and fault-tolerant computing architectures. Contemporary research primarily focuses on three key aspects: computational performance, reliability and redundancy, and issues related to integration with real-time systems [1–3; 6; 7].

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

The enhancement of performance and optimization of hardware resources in RNS has been actively studied in recent years. Works [2; 3] dedicated to the hardware implementation of the reverse transformation from RNS to positional number systems demonstrate that the use of specialized sets of modules, in combination with the Chinese Remainder Theorem or mixed radix methods, can significantly reduce latency and hardware costs [9]. Furthermore, research on methods for constructing module sets, such as the use of diagonal functions, shows that appropriate module selection can substantially improve computational efficiency. Notably, experiments on the application of RNS in highly parallel environments, particularly on GPUs [8], reveal that the large number of independent operations allows for significant acceleration in tasks involving large data sets [4].

Another important research direction concerns the reliability and fault tolerance of RNS [8]. Works [7; 9; 10] in this area emphasize that the inherent redundancy of the residue number system can be leveraged to build fault-tolerant architectures. Special attention is given to the concept of dynamic redundancy, which allows the level of fault tolerance to be adjusted depending on system operating conditions, thereby balancing resource consumption and reliability. Comparisons of reliability models for RNS and traditional binary systems show that, in certain scenarios, RNS can provide a higher probability of fault-free operation with lower hardware costs. This is particularly relevant for critical real-time systems.

At the same time, the literature notes several challenges that hinder the widespread adoption of RNS. Among the most complex tasks are magnitude comparison, sign determination, division, and reconstruction of numbers from residues [10; 11].

A separate line of research focuses on integrating RNS into high-performance real-time systems [3; 6; 12]. The use of FPGA [3], GPU [9; 10], and ASIC [11] technologies is considered a means to practically realize the potential of residue number systems in tasks such as digital signal processing, streaming data processing, and machine learning components. Interest is also growing in adaptive systems capable of dynamically modifying redundancy or reservation parameters depending on current operating conditions. This approach allows achieving the necessary balance between performance, energy efficiency, and fault tolerance [8; 9].

Thus, the literature highlights the significant potential of residue number systems in building high-performance and fault-tolerant real-time systems. At the same time, several fundamental and applied challenges remain, the resolution of which will pave the way for their widespread adoption in critical domains where speed, accuracy, and operational stability are crucial.

Methods of Research. The research methods are based on the principles of systems analysis, number theory, and the theory of computational processes and systems. To analyze the performance and fault tolerance of high-performance real-time systems, the foundations of computer architecture design and the principles of hardware component organization were applied. For the formalization and analysis of parallel data processing, approaches from the theory of computational processes were employed, allowing the evaluation of the temporal characteristics of arithmetic operations and the provision of guaranteed performance bounds [1; 6].

Presentation of the Main Research Material. High-performance real-time information processing systems are critically important in modern computing environments, where delays and failures can have irreversible consequences for control systems, telecommunications, medical equipment, and autonomous computing platforms. Achieving high performance while simultaneously ensuring reliability and fault tolerance requires a specialized numeric representation in the form of a RNS. In this system, a number A is represented as a set of residues over a modular set $\{m_i\}_{i=1}^{n+k}$:

$$A \equiv (a_1, a_2, \ldots, a_n \ldots, a_{n+k}) \,(\mathrm{mod}\, m_i), i = 1, \ldots, n+k,$$

where n is the number of information modules carrying the main computational data, and k is the number of check modules that provide redundancy and error correction. This structure defines the key properties of the RNS: residue independence, residue equivalence, and low-digit residues, each of which confers specific advantages in high-performance real-time systems [7].

Residue independence allows computational pipelines to be built as modular, autonomous subsystems that can operate in parallel without mutual interference. This ensures a high degree of parallelism in computations, which, in real-time systems, reduces latency and increases the throughput of the computing system [8]. The execution time of operations in such a system is determined by the largest modulus among the selected set:

$$T_{op} \sim \max_i(m_i),$$

where T_{op} is the execution time of the arithmetic operation defining the performance of the specific computational pipeline. Localizing errors within a single pipeline eliminates the "error propagation" effect characteristic of classical positional number systems, thus enhancing computational reliability [8].

Residue equivalence allows for the dynamic replacement of a non-functioning pipeline operating with modulus m_i, with another pipeline using modulus $m_j (i \neq j)$ without stopping computations. This property implements the principle of graceful degradation: if several computational pipelines fail, the system continues calculations with reduced precision while maintaining operational capability. Mathematically, this can be expressed through a new distribution of modules:

$$n' + k' = n + k = const, n' < n, k' > k,$$

where n' and k' are the new numbers of information and check modules, respectively. Reducing the number of information modules n' increases computational speed, while increasing the number of check modules k' enhances reliability. To improve accuracy, the redistribution is performed conversely:

$$n'' + k'' = n + k = const, n'' > n, k'' < k,$$

which increases result accuracy by reducing redundancy, slightly decreasing execution speed but providing more precise outcomes. Such dynamic adaptation allows the system to adjust to specific real-time task requirements, balancing speed, accuracy, and fault tolerance [8; 9].

Low-digit residues are another important characteristic of the RNS. They allow a reduction in the hardware resources of computational pipelines and increase their reliability. Additionally, they enable arithmetic operations to be implemented using table-based arithmetic, where the result of a modular operation can be obtained practically in a single clock cycle. Formally, addition and multiplication operations in RNS can be represented as:

$$C_i = (A_i \pm B_i) \bmod m_i, i = 1, \ldots, n + k,$$

$$D_i = (A_i \cdot B_i) \bmod m_i, i = 1, \ldots, n + k,$$

where A_i, B_i, C_i, D_i are the residues of the respective numbers. The inverse transformation from RNS to a positional number system is performed using the Chinese Remainder Theorem:

$$A = \left(\sum_{i=1}^{n} a_i M_i M_i^{-1} \right) \bmod M, M = \prod_{i=1}^{n} m_i,$$

where $M_i = M / m_i$, and M_i^{-1} is the multiplicative inverse of m_i modulo. Thanks to parallel processing of residues, the execution time of operations is reduced to the maximum of the execution times for individual modules, which significantly enhances performance compared to the classical binary system.

The combination of residue independence, equivalence, and low-digit representation in RNS enables the implementation of high-performance systems with the following advantages [7; 9]:

- parallelization of computations at the operand decomposition level;
- spatial separation of data elements with the possibility of asynchronous processing;
- single-cycle execution of modular operations in the table-based variant;
- efficient detection and correction of errors during computations;
- dynamic addition of small redundant blocks to increase fault tolerance;
- rapid reconfiguration of computational structures;
- reduced computational complexity for specific classes of tasks;
- elimination of the error propagation effect in modular pipelines;
- adaptability for real-time diagnostics of blocks and nodes.

The use of RNS is particularly effective for classes of tasks that are traditionally resource-intensive in positional number systems, such as cryptography and modular transformations, digital signal and image processing, high-bit integer data processing in real time, vector and matrix processing of large data arrays, neurocomputing, optoelectronic table-based processing, fast Fourier transform algorithms, and other parallel computational tasks. For such tasks, the execution time of modular operations is determined by the number of information modules *n*, while reliability is determined by the number of check modules *k*, enabling adaptive management of the balance between accuracy and fault tolerance.

Special attention is given to the dynamic reconfiguration of information and check modules during computations. When higher speed is required, the system can reduce the number of information modules *n* and increase the number of check modules *k*, thereby maintaining fault tolerance. Conversely, when higher accuracy is needed, *n* can be increased while *k* is reduced. This adaptation implements the principle of hardware-software flexibility, allowing the system to remain effective in real-time conditions despite changing workload characteristics or computational requirements.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Mathematically, the minimum code distance $d_{\min}$, which characterizes the error-correcting capability of the code, is defined as:

$$d_{\min} = \min_{i \neq j} \left| \sum_{l=1}^{n+k} (a_l - b_l) M_l M_l^{-1} \right| \bmod M,$$

where a_l, b_l are the residues of two distinct numbers. Increasing the number of check modules or adding new modules with larger moduli m_i increases $d_{\min}$, directly enhancing error correction capabilities and improving system fault tolerance. In real-time conditions, this allows "on-the-fly" error correction without interrupting computations, which is critical for control systems and streaming data processing.

Furthermore, RNS enables efficient implementation of modular arithmetic through various methods: the adder-based method (for low-digit modules), table-based method (using precomputed ROM tables), direct logic method at the level of Boolean functions, and circular shift method for registers. This provides a wide range of architectural solutions depending on requirements for speed, power consumption, and hardware complexity [2; 9].

Overall, the application of RNS in high-performance real-time systems allows the creation of computational structures that combine parallelism, adaptability, high reliability, and dynamic error correction capability. Such an architecture enables efficient processing of large volumes of data at high speed and guarantees continuous operation even in the event of partial component failures. This makes RNS a promising tool for building modern real-time computing systems capable of solving complex tasks in digital information processing, cryptography, neurocomputing, and streaming computations.

Conclusions. The conducted research has shown that the use of the RNS is an effective approach for building high-performance real-time information processing systems. The main advantages of this approach are determined by the properties of RNS: independence, uniformity, and low-digit residues. Independence of residues allows parallel data processing at the level of modular computational units, significantly increasing the system's speed and throughput. Uniformity of residues provides the ability to dynamically replace failed units without stopping computations, implementing the principle of graceful system degradation, which is critical for real-time systems. Low-digit residues reduce the hardware complexity of units, enhance the speed of operations, and enable efficient error correction during computations.

Analysis of arithmetic operations in RNS has shown that execution time is determined by the largest modulus among the chosen set, while system accuracy and reliability can be dynamically adjusted through redistribution of information and check modules. This approach allows adaptive balancing between speed, accuracy, and fault tolerance depending on the specific task requirements, which is a key factor for real-time systems.

Moreover, RNS enables single-cycle execution of modular operations using table-based arithmetic, adder-based, logic-based, or circular-shift implementation principles, making these systems versatile for various architectural solutions. Due to these properties, RNS is effectively applied for processing large data arrays, digital signal and image processing, cryptographic computations, neurocomputing, and other tasks requiring high performance and fault tolerance.

Thus, the concept of building high-performance real-time systems based on RNS allows combining parallelism, adaptability, high reliability, and dynamic error correction capability. Implementing such systems ensures continuous computations even in the event of partial component failures, increases system performance and flexibility, making this approach promising for modern real-time computing platforms across various fields of science and engineering.

Bibliography:

1. Aliluiko A., Kasianchuk M. Arithmetic of Asymmetric Cryptosystems in the Field of Complex Numbers. *Ukrainian Information Security Research Journal*. 2024. Vol. 26, No. 1. P. 35–43. DOI: https://doi.org/10.18372/2410-7840.26.18825

2. Bovchalyuk S., Bovchalyuk N., Drozd O. Concept of "Modular Architecture" for Parallel-Action Control Devices. *Systems of Control, Navigation and Communication*. Poltava: PNTU, 2025. Vol. 2, No. 80. P. 47–53. DOI: https://doi.org/10.26906/SUNZ.2025.2.046

3. Bovchalyuk S. Ya., Piskaryov O. M., Radchenko S. S., et al. Definition of Directions for the Development of Control Devices with Parallel Architecture Based on FPGA. *Systems of Control, Navigation and Communication*. Poltava: PNTU, 2023. Issue 1 (71). P. 69–72. DOI: https://doi.org/10.26906/SUNZ.2023.1.069

4. Devi R. P., Neeraja P., Ajay V. K., Raj A. N., Reddy D. V. L., Ramachandran G. Analysis of Artificial Intelligence Hybrid Security Cloud System Intelligent Technology and its Applications. 2024 4th International Conference on Soft Computing for Security Applications (ICSCSA), Salem, India. 2024. P. 506–509. DOI: https://doi.org/10.1109/ICSCSA64454.2024.00087

5. Jency Rubia J., Sherin Shibi C., Balajishanmugam V., Babitha Lincy R. High-Performance Computing Based on Residue Number System: A Review. 2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India. 2023. P. 639–647. DOI: https://doi.org/10.1109/ICACCS57279.2023.10112959

6. Koshman S., Krasnobayev V., Nikolsky S., Kovalchuk D. The Structure of the Computer System in the Residual Classes. *Advanced Information Systems*. 2023. Vol. 7, No. 2. P. 41–48. DOI: https://doi.org/10.20998/2522-9052.2023.2.06

7. Krasnobayev V., Koshman S., Kurchanov V., Zinevich D. Main Properties of Non-Positional Number System in Residue Classes and Their Influence on the Structure and Principles of Arithmetic Operations Implementation in Computer Systems. *Systems of Control, Navigation and Communication*. Poltava: PNTU, 2019. Vol. 2, No. 54. P. 114–118. DOI: https://doi.org/10.26906/SUNZ.2019.2.114

8. Krasnobayev V., Yanko A., Kovalchuk D., Fil I. Synthesis of a Mathematical Model of a Fault-Tolerant Real-Time Computer System Operating in Non-Positional Arithmetic in Residual Classes. Mathematical Modeling and Simulation of Systems. MODS 2023. Lecture Notes in Networks and Systems. Cham: Springer, 2024. Vol. 1091. P. 186–199. DOI: https://doi.org/10.1007/978-3-031-67348-1_14

9. Radchenko S., Demchenko K., Hrytsenko S. Methods of Increase Reliability in Automated Control Systems. *SWorldJournal*. 2024. Vol. 1, No. 23-01. P. 111–115. DOI: https://doi.org/10.30888/2663-5712.2024-23-00-033

10. Shen S., Yang H., Liu Y., Liu Z., Zhao Y. CARM: CUDA-Accelerated RNS Multiplication in Word-Wise Homomorphic Encryption Schemes for Internet of Things. *IEEE Transactions on Computers*. 2023. Vol. 72, No. 7. P. 1999–2010. DOI: https://doi.org/10.1109/TC.2022.3227874

11. Shrimali D., Sharma L. An Extensive Review on Residue Number System for Improving Computer Arithmetic Operations, *International Research Journal of Engineering and Technology (IRJET)*. 2018. Vol. 5, Issue 12. P. 1617–1621

12. Yatskiv V., Kulyna S., Bykovyy P., Maksymyuk T., Sachenko A. Method of Reliable Data Storage Based on Redundant Residue Number System. 2020 IEEE 5th International Symposium on Smart and Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS), Dortmund, Germany. 2020. P. 1–4. DOI: 10.1109/IDAACS-SWS50031.2020.9297052

Дата надходження статті: 25.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.415.538
DOI https://doi.org/10.32689/maup.it.2025.3.11

Кирило КОХАН
аспірант кафедри комп'ютерних інформаційних систем і технологій,
Національний університет біоресурсів і природокористування України,
kokhan.kyrylo@gmail.com
***ORCID:** 0009-0002-8878-8527*

Олексій ТКАЧЕНКО
кандидат технічних наук, доцент кафедри теорії та технології програмування,
Київський національний університет імені Тараса Шевченка,
otkachenko@knu.ua
***ORCID:** 0000-0002-9514-516X*

ОГЛЯД ТА ПРОПОЗИЦІЯ ОПТИМІЗАЦІЇ ОПТИМАЛЬНИХ КОНФІГУРАЦІЙ ДЛЯ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ БАГАТОКОМПОНЕНТНИХ ІНФОРМАЦІЙНИХ СИСТЕМ

***Анотація**. У статті проаналізовано сучасні підходи до вибору оптимальних конфігурацій для автоматизованого тестування багатокомпонентних інформаційних систем (ІС), що є основою функціональності цифрових платформ. Визначено ключові поняття: конфігурація як комбінація параметрів (версії програмного забезпечення, бази даних, браузери), оптимальна конфігурація як мінімальний набір комбінацій для покриття критичних сценаріїв за умов обмеження ресурсів, багатокомпонентна ІС як сукупність взаємопов'язаних компонентів (фронтенд, бекенд, API, бази даних).*

***Метою дослідження** є оцінка сучасних методів та створення інтегрованого підходу, що поєднує комбінаторні методи, генетичні алгоритми та CI/CD для автоматизації вибору конфігурацій у реальному часі.*

***Методологія дослідження** включає систематичний огляд літератури за останні 5 років, порівняльний аналіз із використанням вагових коефіцієнтів (кількість тестів, покриття сценаріїв, адаптивність, інтеграція з CI/CD, ресурси), математичне моделювання та апробацію на прикладі хмарних платформ і систем електронної комерції.*

***Наукова новизна** полягає в розробці інтегрованого підходу, який скорочує кількість тестів до 5–10% від повного набору (наприклад, із 243 до 12–24 конфігурацій для системи з 5 параметрами), забезпечуючи при цьому 90–95% покриття критичних сценаріїв та високу адаптивність до змін компонентів. Унікальність підходу – інтеграція з CI/CD-процесами та використання вагового аналізу для вибору оптимальних конфігурацій.*

***Висновки**. Запропонований підхід дозволяє оптимізувати тестування в умовах складних ІС, поєднуючи точність комбінаторних методів, ефективність генетичних алгоритмів та автоматизацію CI/CD. Перспективи подальших досліджень – використання ШІ для прогнозування дефектів і автоматичний аналіз результатів тестування.*

***Ключові слова:** автоматизоване тестування, багатокомпонентні інформаційні системи, оптимальні конфігурації, Pairwise Testing, генетичні алгоритми, CI/CD.*

Kyrylo KOKHAN, Oleksii TKACHENKO. INFORMATION TECHNOLOGY FOR OPTIMAL CONFIGURATION SELECTION IN AUTOMATED TESTING OF MULTICOMPONENT INFORMATION SYSTEMS: REVIEW AND PROPOSAL

***Abstract.** The article analyzes modern approaches to selecting optimal configurations for automated testing of multi-component information systems (IS), which are the basis of the functionality of digital platforms. Key concepts are defined: configuration as a combination of parameters (software versions, databases, browsers), optimal configuration as a minimum set of combinations to cover critical scenarios under resource constraints, multi-component IS as a set of interconnected components (frontend, backend, API, databases).*

***The objective of the study** is to evaluate current methods and create an integrated approach that combines combinatorial methods, genetic algorithms, and CI/CD to automate configuration selection in real time.*

***Methodology** includes a systematic review of the literature over the last 5 years, comparative analysis using weighting factors (number of tests, scenario coverage, adaptability, integration with CI/CD, resources), mathematical modeling, and testing on the example of cloud platforms and e-commerce systems.*

***The scientific novelty** lies in developing an integrated approach that reduces the number of tests to 5–10% of the full set (for example, from 243 to 12–24 configurations for a system with 5 parameters), while ensuring 90–95% coverage of critical scenarios and high adaptability to component changes. The uniqueness of the approach is integration with CI/CD processes and the use of weight analysis to select optimal configurations.*

***Conclusions.** The proposed approach allows optimizing testing in complex IS environments, combining the accuracy of combinatorial methods, the efficiency of genetic algorithms, and CI/CD automation. Prospects for further research include the use of AI for defect prediction and automatic analysis of test results.*

***Key words:** automated testing, multi-component information systems, optimal configurations, Pairwise Testing, genetic algorithms, CI/CD.*

Постановка проблеми. Сучасні багатокомпонентні інформаційні системи (ІС), такі як хмарні платформи, системи електронної комерції чи мікросервісні архітектури, характеризуються високою складністю через велику кількість взаємопов'язаних компонентів (фронтенд, бекенд, бази даних, API), кожен із яких може мати різні версії, працювати в різних середовищах (локальне, production) і залежати від зовнішніх параметрів (браузери, бази даних). Це призводить до стрімкого зростання кількості тестових конфігурацій, що робить їх повне тестування неможливим через обмеження часу та ресурсів [6]. Наприклад, система з 5 параметрами по 3 значення кожен має $3^5 = 243$ конфігурації, що ускладнює забезпечення якості програмного забезпечення без автоматизації. Ряд сучасних інструментів, зокрема заснованих на методах штучного інтелекту, не завжди допомагає вирішити цю проблему [12]. Необхідність створення ефективних методів вибору оптимальних конфігурацій для автоматизованого тестування є актуальним завданням, яке потребує поєднання високого покриття критичних сценаріїв, мінімальної кількості тестів і адаптивності до змін системи [1].

Аналіз останніх досліджень і публікацій. Проблеми тестування багатокомпонентних ІС активно досліджуються в науковій літературі. Ручний вибір конфігурацій є поширеним у невеликих проєктах, але його ефективність знижується зі зростанням складності ІС через суб'єктивність і трудомісткість. Комбінаторні методи, такі як Pairwise Testing, скорочують кількість тестів шляхом покриття всіх пар значень параметрів, що дозволяє виявити до 85–90% дефектів, викликаних взаємодією двох параметрів [6; 10]. Orthogonal Arrays забезпечують збалансоване покриття, але їх генерація складніша [9]. Генетичні алгоритми (ГА) застосовуються для оптимізації вибору конфігурацій шляхом ітеративного пошуку, однак вони потребують налаштування параметрів і мають обмежену інтеграцію з CI/CD [13]. Методи на основі машинного навчання (ML) прогнозують проблемні конфігурації, але потребують значних обсягів даних і обчислювальних ресурсів [11; 4; 12]. У низці досліджень пропонують поєднувати комбінаторні методи з ГА чи ML, однак такі підходи рідко враховують повну автоматизацію в CI / CD-процесах [1; 3; 2]. Питання інтеграції комбінаторних методів із ГА та CI/CD залишається недостатньо дослідженим [7].

Метою даної статті є оцінити сучасні підходи щодо вибору оптимальних конфігурацій для автоматизованого тестування багатокомпонентних інформаційних систем та запропонувати інтегрований підхід, спрямований на оптимізацію параметрів автоматизованого тестування.

Виклад основного матеріалу. Матеріали та методи дослідження. Дослідження базується на аналізі наукової літератури з комбінаторного тестування, генетичних алгоритмів та методів ML. Використано комбінацію комбінаторних методів (Pairwise Testing, Orthogonal Arrays), генетичних алгоритмів та інтеграцію з CI/CD-пайплайнами. Для порівняльного аналізу запропоновано набір вагових коефіцієнтів, визначених на основі огляду наукових робіт [1–13], досвіду автора та експертів у галузі, що дозволило коректно оцінити практичну значущість кожного параметра. Слід зазначити, що вагові коефіцієнти можуть бути змінені залежно від конкретних завдань тестування. Наприклад, для фінансових систем пріоритетним є покриття критичних сценаріїв, а для високонавантажених сервісів – ефективність використання ресурсів. Матеріалами є технічні специфікації, UML-діаграми, конфігураційні файли (JSON, YAML) та вимоги до продуктивності реальних багатокомпонентних ІС, таких як хмарні платформи та системи електронної комерції. Методи включають математичне моделювання простору конфігурацій, оптимізацію за інтегральним критерієм та статистичну обробку результатів за допомогою вагових коефіцієнтів.

У якості матеріалів використано:

- технічні специфікації ІС, UML-діаграми та JSON/YAML-файли конфігурацій;
- реальні приклади CI/CD-процесів (GitLab CI, Jenkins) [3];
- інструменти автоматизації тестування (Selenium, Postman, JMeter) [12];
- наукові роботи, що обґрунтовують ефективність комбінаторних підходів [6; 9; 10] та еволюційних алгоритмів [13].

Методи дослідження включають:

- Pairwise Testing та Orthogonal Arrays, що дозволяють зменшити кількість тестів у 8–12 разів [6; 9; 8];
- генетичні алгоритми (GA), які використовуються для оптимізації вибору конфігурацій [13];
- метод вагових коефіцієнтів, що забезпечує багатокритеріальне оцінювання ефективності підходів [1; 4; 12].

Запропонований підхід побудовано у вигляді модульної схеми (рис. 1). Він передбачає п'ять етапів:

1. *Модуль аналізу системи.* Здійснює формалізований збір інформації про компоненти ІС, їх функціональні залежності та критичні сценарії. Результатом є структурована модель предметної області.

2. *Модуль моделювання конфігурацій.* Формує простір усіх можливих варіантів та зменшує його за допомогою Pairwise Testing і Orthogonal Arrays[6, 9] (наприклад, з 243 до 20–30 комбінацій).

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Рис. 1. Структура запропонованого підходу

3. *Модуль оптимізації.* Виконує відбір підмножини конфігурацій на основі інтегральної оцінки, яка враховує покриття, час і ресурси. Для оптимізації застосовуються як класичні методи (симплекс-метод), так і евристики (генетичні алгоритми) [13].

4. *Модуль тестування.* Інтегрується з інструментами (Selenium, Postman, JMeter) та забезпечує виконання UI-, API- і навантажувальних тестів [4; 11].

5. *Модуль аналізу результатів.* Збирає інформацію про виявлені дефекти й метрики покриття. Отримані дані використовуються для уточнення вагових коефіцієнтів і повторного циклу оптимізації [12; 3].

Підхід комбінує переваги комбінаторних методів і генетичних алгоритмів [5]: Pairwise Testing забезпечує високу ефективність із покриттям 85–90% дефектів, генетичні алгоритми оптимізують вибір конфігурацій із урахуванням пріоритетів критичних сценаріїв, а відсутність залежності від великих даних чи складних обчислень відрізняє його від ШІ-підходів [4; 11].

Порівняльний аналіз підходів. Об'єктивне порівняння методів проведено за допомогою інтегральної оцінки:

$$I(j) = \sum_{i=1}^{n} Wi * Pij$$

де Wi – ваговий коефіцієнт параметра, Pij – нормалізоване значення параметра для методу j, n – кількість параметрів.

Для запропонованого підходу інтегральна оцінка становить:

$$I = 0.20 \cdot 0.95 + 0.25 \cdot 0.95 + .. + 0.10 \cdot 1.00 + 0.05 \cdot 0.70 = 0.9075$$

Отримане значення свідчить про перевагу запропонованого методу над класичними підходами. У (табл. 1) наведено порівняльне оцінювання підходів у тестуванні на основі вагових коефіцієнтів.

Обговорення результатів**.** Аналіз таблиці показує:

- ручний підхід має низьку ефективність (0.395) через надлишкову кількість тестів і відсутність автоматизації[9];
- Pairwise Testing забезпечує високе покриття з мінімальною кількістю тестів (0.8025), що підтверджено працями Куна та колег [6; 10];
- генетичні алгоритми покращують адаптивність, але вимагають значних обчислювальних ресурсів [13; 1];
- ML-методи демонструють високе покриття (0.95), однак залежать від великих обсягів історичних даних [11; 4; 12];
- запропонований підхід отримав найвищу інтегральну оцінку (0.9075), поєднавши переваги комбінаторних і еволюційних методів із простотою CI/CD-інтеграції [7; 3].

Таблиця 1

Порівняльне оцінювання підходів у тестуванні на основі вагових коефіцієнтів

Параметр	W(i)	Ручний вибір	Pairwise Testing	Генетичні алгоритми	ШІ (ML)	Запропонований підхід
Кількість тестів	0.20	0.10	0.95	0.70	0.70	0.95
Покриття критичних сценаріїв	0.25	0.50	0.85	0.80	0.95	0.95
Адаптивність до змін	0.15	0.10	0.10	0.60	0.60	0.95
Використання історичних даних	0.10	1.00	1.00	0.70	0.10	0.70
Інтеграція з CI/CD	0.15	0.10	0.50	0.30	0.30	1.00
Обчислювальні ресурси	0.10	1.00	1.00	0.60	0.20	1.00
Складність впровадження	0.05	1.00	0.70	0.30	0.30	0.70
Інтегральна оцінка	1.00	0.395	0.8025	0.6400	0.5725	0.9075

Джерело: сформовано автором на підставі (Kuhn, Kacker та Lei, 2013; Mandl, 1985; Nie та Leung, 2011; Suafel та Harman, 2019; Segall та Tzoref-Brill, 2018; Durelli, Durelli та Endo, 2019; Grindal, Offutt та Andler, 2005; Kuhn, Wallace та Gallo, 2004; Cohen, Gibbons, Mugridge та Colbourn, 2003; Lei та Tai, 1998).

Висновки. У процесі дослідження запропоновано інтегрований підхід, який поєднує комбінаторні методи (Pairwise Testing, Orthogonal Arrays) із генетичними алгоритмами та інтеграцією з CI/CD-пайплайнами, є одним із найбільш ефективних для вибору оптимальних конфігурацій у тестуванні багатокомпонентних інформаційних систем (ІС), і йому слід приділити більше уваги. Розроблені вагові коефіцієнти та порівняльний аналіз дозволяють, з одного боку, визначити сильні й слабкі сторони існуючих методів (ручний відбір, генетичні алгоритми, ШІ-підходи); з іншого – синтезувати підхід, що забезпечує високу ефективність, адаптивність і мінімальні ресурси. Використання цього підходу дозволяє структурувати процес тестування, оптимізувати вибір конфігурацій і адаптуватися до змін у системах, скорочуючи кількість тестів до 5–10% від повного набору при збереженні 90–95% покриття критичних сценаріїв. Встановлено, що серед досліджених методів найбільш адаптивними є комбінаторні методи, зокрема Pairwise Testing, доповнені генетичними алгоритмами та автоматизацією через CI/CD. Подальшого дослідження потребує інтеграція ШІ-компонентів для прогнозування дефектів на основі обмежених даних, розробка методів автоматичного аналізу результатів тестування за допомогою статистичних моделей, а також адаптація підходу до специфічних галузей, таких як фінанси чи інфраструктурні проєкти, де критичність сценаріїв і складність систем вимагають особливої уваги.

Список використаних джерел:

1. Bansal S. Empirical Studies on Automated Software Testing Practices : монографія. USC, 2022. URL: https://www.researchgate.net/publication/369475828_Empirical_Studies_on_Automated_Software_Testing_Practices (дата звернення: 22.09.2025).

2. Cohen M. B., Gibbons P. B., Mugridge W. B., Colbourn C. J. Constructing Test Suites for Interaction Testing : матеріали конференції. Proceedings of the 25th International Conference on Software Engineering. 2003. P. 38–48.

3. De Sousa Ribeiro Filho F. Automated security testing in DevSecOps pipelines : стаття. *WJARR*. 2025. URL: https://wjarr.com/sites/default/files/WJARR-2024-1083.pdf (дата звернення: 22.09.2025).

4. Durelli W. H., Durelli R. S., Endo A. T. Applying Machine Learning to Software Testing: A Systematic Review : стаття. *IEEE Transactions on Reliability*. 2019. Vol. 68, No 3. P. 1189–1212.

5. Grindal M., Offutt J., Andler S. F. Combination Testing Strategies: A Survey : стаття. *Software Testing, Verification and Reliability*. 2005. Vol. 15, No 3. P. 167–199.

6. Kuhn R., Kacker R., Lei Y. Introduction to Combinatorial Testing : монографія. Boca Raton : CRC Press, 2013. 333 с.

7. Kuhn D. R., Wallace D. R., Gallo A. M. Software Fault Interactions and Implications for Software Testing : стаття. *IEEE Transactions on Software Engineering*. 2004. Vol. 30, No 6. P. 418–421.

8. Lei Y., Tai K. C. In-Parameter-Order: A Test Generation Strategy for Pairwise Testing : матеріали конференції. *Proceedings of the 3rd IEEE International High-Assurance Systems Engineering Symposium*. 1998. P. 254–261.

9. Mandl R. Orthogonal Latin Squares: A Tool for the Design of Experiments in Testing : стаття. *Software Testing, Verification and Reliability*. 1985. Vol. 2, No 2. P. 23–31.

10. Nie C., Leung H. A Survey of Combinatorial Testing : стаття. *ACM Computing Surveys*. 2011. Vol. 43, No 2. P. 1–29.
11. Segall I., Tzoref-Brill R. Using Machine Learning to Improve Test Case Generation : стаття. *IEEE International Conference on Software Testing, Verification and Validation*. 2018. P. 45–53.
12. Sharma A. Test Suite Optimization Using Machine Learning Techniques : монографія. DSU, 2024. URL: https://www.researchgate.net/publication/385478252_Test_Suite_Optimization_Using_Machine_Learning_Techniques_A_Comprehensive_Study (дата звернення: 22.09.2025).
13. Suafel L., Harman M. Evolutionary Algorithms for Software Testing: A Survey : стаття. *Journal of Systems and Software*. 2019. Vol. 152. P. 112–124.

Дата надходження статті: 22.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

UDC 004.738.5:159.9
DOI https://doi.org/10.32689/maup.it.2025.3.12

Snizhana KUTSYN
Bachelor's Student, Institute of Digital Technologies, Design and Transport,
National University "Odesa Polytechnic"
***ORCID:** 0009-0008-0668-4657*

COGNITIVE ASPECTS OF UX DESIGN IN ENSURING THE USABILITY OF WEB RESOURCES

Abstract. *Web resource usability today is largely defined by the extent to which cognitive mechanisms of human perception, memory, and decision-making are respected in design solutions. While traditional UX frameworks emphasize visual clarity and interaction efficiency, they rarely formalize cognitive aspects as measurable design checkpoints. This gap necessitates the development of new approaches that directly integrate cognitive ergonomics into digital environments.*

The aim *of the article is to investigate the cognitive aspects of UX design as a foundation for ensuring the usability of web resources, with the aim of developing an authorial methodology that integrates psychological principles, design heuristics, and adaptive mechanisms.*

The scientific novelty *of this paper introduces an authorial methodology called Cognitive Flow UX (CF-UX), conceived as a structured system for embedding cognitive ergonomics into digital environments. The CF-UX model is based on five interdependent dimensions – perception clarity, working memory load, decision latency, error anticipation, and motivational feedback – which together form a diagnostic matrix for identifying usability gaps. Unlike generic heuristic evaluations, the CF-UX approach operationalizes these dimensions into design interventions such as chunked navigation pathways, adaptive prompts, predictive error recovery modules, and reward-driven feedback loops.*

The conclusions *show that the study introduced and validated the CF-UX methodology across e-learning, e-government, and e-commerce platforms, demonstrating measurable improvements: task completion time decreased by 22–35%, error frequency fell by up to 40%, and user satisfaction (SUS scores) increased by an average of 18 points. These results prove that web usability cannot be ensured solely through visual or technical optimization but requires alignment with cognitive mechanisms of perception, memory, and decision-making. The main contribution of this research is threefold: operationalizing cognitive science constructs into measurable design levers, offering a replicable evaluative framework for systematic usability optimization, and demonstrating applicability across diverse domains, ensuring cognitive sustainability, inclusivity, and long-term engagement.*

Key words: *user experience, cognitive flow, usability, web resources, design methodology, cognitive ergonomics, task efficiency, error reduction, motivational feedback.*

Сніжана КУЦИН. КОГНІТИВНІ АСПЕКТИ UX-ДИЗАЙНУ У ЗАБЕЗПЕЧЕННІ ЗРУЧНОСТІ ВИКОРИСТАННЯ ВЕБРЕСУРСІВ

Анотація. *Юзабіліті вебресурсів сьогодні значною мірою визначається тим, наскільки у проєктних рішеннях враховані когнітивні механізми людського сприйняття, пам'яті та прийняття рішень. Традиційні UX-фреймворки роблять акцент на візуальній зрозумілості та ефективності взаємодії, проте рідко формалізують когнітивні аспекти як вимірювані контрольні точки дизайну. Ця прогалина зумовлює потребу у розробленні нових підходів, які безпосередньо інтегрують когнітивну ергономіку в цифрові середовища.*

Метою статті *є дослідження когнітивних аспектів UX-дизайну як основи забезпечення юзабіліті вебресурсів з подальшою розробкою авторської методології, що поєднує психологічні принципи, дизайнерські евристики та адаптивні механізми.*

Науковою новизною *є представлена авторська методологія Cognitive Flow UX (CF-UX), розроблена як структурована система для впровадження когнітивної ергономіки у цифрові середовища. Модель CF-UX ґрунтується на п'яти взаємопов'язаних вимірах – чіткості сприйняття, навантаженні робочої пам'яті, латентності прийняття рішень, передбаченні помилок та мотиваційному зворотному зв'язку. Сукупність цих параметрів формує діагностичну матрицю для виявлення проблем юзабіліті. На відміну від загальних евристичних оцінювань, підхід CF-UX операціоналізує зазначені виміри у вигляді конкретних дизайнерських інтервенцій, таких як сегментовані навігаційні шляхи, адаптивні підказки, модулі прогнозного відновлення після помилок та мотиваційні зворотні петлі.*

Висновки *показують, що методологію CF-UX було апробовано у трьох контекстах – електронне навчання, електронне урядування та електронна комерція. У всіх випадках зафіксовано суттєві покращення: час виконання завдань скоротився на 22–35%, частота помилок зменшилася до 40%, а рівень задоволеності користувачів (за шкалою SUS) зріс у середньому на 18 пунктів. Це доводить, що юзабіліті вебресурсів неможливо забезпечити лише завдяки візуальній чи технічній оптимізації – необхідна також відповідність когнітивним механізмам сприйняття, пам'яті та прийняття рішень. Основний внесок дослідження є триєдиним: операціоналізація когнітивних наукових конструкцій у вимірювані дизайнерські інструменти, розроблення відтворюваної оціночної рамки для системної оптимізації юзабіліті, демонстрація застосовності методології у різних доменах, що забезпечує когнітивну сталість, інклюзивність і довготривалу залученість користувачів.*

Ключові слова: *UX-дизайн, когнітивний потік, зручність використання, вебресурси, авторська методика, когнітивна ергономіка, ефективність виконання завдань, скорочення помилок, мотиваційний фідбек.*

Problem statement. In today's digital environment, web resources are not only technical systems but also cognitive environments in which users interact, process information, and make decisions. The problem of ensuring their usability is becoming increasingly relevant as audiences grow more diverse, tasks more complex, and expectations for efficiency and accessibility higher. One of the critical threats to usability is the cognitive overload that arises when the design of a resource does not align with human perceptual and memory limitations. This mismatch may manifest as excessive navigation depth, poorly structured information, unclear interaction flows, or insufficient feedback, all of which lead to errors, frustration, and task abandonment. Cognitive overload can be triggered by external factors such as the growth of content volume, new interaction patterns, and technological innovations, as well as by internal factors such as users' varying levels of expertise, cultural background, and cognitive abilities. Ignoring these aspects reduces not only satisfaction but also the effectiveness of digital systems in education, healthcare, commerce, and public services. The problem is the lack of a universal methodology that systematically incorporates cognitive ergonomics into UX design to measure, anticipate, and mitigate overload.

Most existing design practices focus on aesthetic appeal or functional completeness but provide limited mechanisms for monitoring and adjusting cognitive demands placed on users. This gap makes it difficult to design web resources that remain consistently usable across contexts and user groups. In this regard, the development of a structured cognitive-centered methodology, capable of identifying, measuring, and integrating mental load parameters into design decisions, is of particular scientific and practical importance. The practical significance of addressing this problem is to ensure that web resources become cognitively sustainable, support diverse user needs, and maintain usability under changing technological and social conditions.

Analysis of the latest research and publications. The analysis of scientific research confirms that cognitive aspects of UX design in web environments can be structured into three main areas: (1) integration of cognitive principles into interface design, (2) empirical methods for evaluating usability and user experience, and (3) adaptive and intelligent approaches to sustaining usability across contexts.

The first area covers the inclusion of cognitive psychology and ergonomics in user interface development. K. St. Amant emphasizes that concepts such as perception, attention, and decision-making can be systematically applied to improve usability in medical and health-related contexts, thereby reducing errors in critical environments [10]. L. Moreno, R. Alarcon, and P. Martínez demonstrate how interfaces adapted for people with cognitive disabilities can significantly reduce barriers to access, illustrating the need to translate cognitive theory into applied design strategies [8]. Similarly, the work of L. Moreno, H. Petrie, P. Martínez, and R. Alarcon extends this approach by introducing content simplification techniques that directly target memory and comprehension challenges, thereby strengthening inclusivity in web resources [9]. It is advisable to complement this direction by developing frameworks that formalize cognitive parameters as measurable checkpoints in UX processes.

The second area concerns empirical assessment methods and tools for usability. E. Banuelos-Lozoya, G. Gonzalez-Serna, and colleagues present a systematic review of cognitive state-based QoE/UX evaluation, highlighting how physiological and psychological measures can complement traditional usability testing [3]. M. Țichindelean, M. T. Țichindelean, I. Cetină, and G. Orzan use eye-tracking experiments to demonstrate how navigation structure and visual hierarchy affect cognitive load in sustainable web design [11]. J. Zheng, M. Gresham, and their team explore supportive websites for people with dementia and carers, focusing on usability factors that reduce confusion and enhance trust [15]. These studies reveal that cognitive overload and accessibility issues remain central barriers, and it is advisable to expand evaluation practices by integrating both behavioral data and cognitive diagnostics.

The third area addresses adaptive and intelligent mechanisms in UX. O. D. Alao, A. P. Ezihe, and collaborators apply user-centered UX thinking to the design of a university information system, showing how iterative feedback loops align interfaces with users' cognitive expectations [1]. W. Li, Y. Zhou, S. Luo, and Y. Dong propose design factors that ensure consistency and sustainability in responsive interfaces, pointing out that adaptive layout plays a role in reducing cognitive switching costs [7]. A. Khamaj and A. M. Ali explore reinforcement learning as a way to personalize user experience in real time, tailoring interfaces to behavioral and cognitive patterns [5]. Finally, M. Virvou reviews the role of artificial intelligence in UX, demonstrating how AI-driven adaptation can support reciprocity and personalized usability [12]. It is advisable to further advance this area by combining reinforcement learning, explainable AI, and distributed cognition analysis to achieve web environments that dynamically adapt to cognitive states.

The general analysis shows that ensuring usability through cognitive aspects of UX design requires an interdisciplinary approach combining psychology, design, and intelligent systems. Despite significant advances, unresolved issues remain, including the absence of universal frameworks for cognitive-centered

design, limited cross-contextual validation of evaluation methods, and insufficient integration of adaptive AI tools into everyday UX practice. The proposed research aims to address these challenges by developing an authorial methodology – Cognitive Flow UX – that unifies cognitive parameters, measurable metrics, and adaptive design techniques to ensure sustainable usability of web resources across diverse domains.

The purpose of the article is to investigate the cognitive aspects of UX design as a foundation for ensuring the usability of web resources, with the aim of developing an authorial methodology that integrates psychological principles, design heuristics, and adaptive mechanisms. To achieve this goal, the following tasks have been identified:

1. Analyze scientific research on cognitive principles relevant to UX design (perception, memory, attention, decision-making) and evaluate their impact on web usability.
2. Identify and systematize practical design interventions (structured navigation, adaptive prompts, error-prevention techniques, motivational feedback) that reduce cognitive overload and improve task efficiency.
3. Develop and substantiate the authorial methodology Cognitive Flow UX (CF-UX) as a structured framework for embedding cognitive ergonomics into web design.

Summary of the main material. The proposed methodology Cognitive Flow UX (CF-UX) unites five interdependent cognitive dimensions into a coherent framework for enhancing web usability: perception clarity, working memory load, decision latency, error anticipation, and motivational feedback. While existing UX approaches emphasize visual clarity and functional efficiency, CF-UX positions cognition as the central design checkpoint, ensuring that usability reflects how users perceive, process, and act in digital environments. Perception clarity focuses on structuring navigation and visual hierarchies in ways aligned with human attention. By introducing chunked navigation pathways and consistent iconography, fragmented user flows are transformed into coherent modules. Pilot testing in e-learning systems showed that when navigation was restructured around perception clarity, orientation errors fell and time-to-first-action improved by 15–20%. Working memory load addresses the limits of simultaneous information retention. Progressive disclosure and adaptive prompts prevent overload by exposing only task-relevant content at each step. In government service portals, restructured multi-step forms with contextual hints reduced abandonment rates by up to 30% and improved form accuracy, confirming that cognitive load management directly influences task persistence. Decision latency captures the time and effort required for users to make choices. In e-commerce platforms, layered product filtering and guided comparisons reduced decision fatigue, enabling users to complete selections faster while maintaining accuracy. This intervention produced a 22–35% decrease in task duration, highlighting that decision scaffolding sustains both efficiency and confidence. Error anticipation emphasizes predictive recovery mechanisms. Inline validation, undo options, and real-time feedback proactively prevent errors or correct them before they escalate. In case studies from public service portals, error anticipation reduced error frequency by up to 40%, strengthening both trust and reliability. Motivational feedback sustains engagement through progress visualization, micro-interactions, and reward-driven cues. In university e-learning platforms, completion bars and personalized acknowledgements improved not only engagement but also user satisfaction, with System Usability Scale (SUS) scores increasing on average by 18 points.

To ground these findings in practitioner-facing form, (Tab. 1) maps the five CF-UX dimensions to representative practices, their observed direction of impact on usability indicators, and the preconditions that consistently separated successful from fragile implementations.

Table 1

Mapping of CF-UX dimensions to cognitive levers and usability outcomes

CF-UX Dimension	Cognitive design lever (author's approach)	Empirical effect on user interaction	Implementation focus
Perception clarity	Modular navigation trees; perceptual anchors (color, spacing, icons)	Faster orientation (−20% navigation time); fewer misclicks	Requires prior mapping of user journeys and content clustering
Working memory balance	Progressive disclosure with adaptive pacing; contextual reminders	Lower task abandonment (−25%); improved recall of form data	Needs calibration of disclosure thresholds via usability testing
Decision guidance	Hierarchical filtering; comparative scaffolds; embedded micro-feedback	Reduced decision fatigue (−30%); higher confidence in choice	Depends on domain-specific option hierarchies and relevance modeling
Anticipatory recovery	Predictive validation; soft-error pathways (undo/redo, "safe defaults")	Error rate reduced (−40%); trust and reliability perception ↑	Demands integration of real-time error logging and predictive heuristics
Motivational sustain	Progress feedback loops; micro-rewards; adaptive encouragement	Sustained engagement ↑; SUS +18 points on average	Works best when tied to intrinsic goals (learning, completion, self-efficacy)

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

As seen from Table 1, perception clarity and anticipatory recovery stand out as the strongest dimensions within the CF-UX framework, consistently demonstrating the highest adoption and the most pronounced usability gains. When navigation is modular and error recovery is predictive, users complete tasks faster, make fewer mistakes, and experience lower frustration levels. Working memory balance and decision guidance show context-dependent benefits, particularly in complex forms and high-stakes decision scenarios, where cognitive scaffolding reduces abandonment and accelerates information processing. Motivational sustain, although often underestimated, proves to be a decisive factor in long-term engagement, especially in educational and service-oriented contexts where user persistence is critical. The diagnostic grid highlights that usability gains are not uniform but emerge when cognitive levers are deliberately aligned with user needs, system context, and domain-specific tasks. This reinforces the uniqueness of CF-UX as a methodology that transforms cognitive science insights into actionable, measurable design interventions. (Fig. 1) shows the distribution of the five CF-UX dimensions along two axes: adoption intensity and observed impact on usability.

As seen from Figure 1, the five CF-UX dimensions occupy different positions on the adoption–impact plane. Perception clarity and anticipatory recovery are concentrated in the high-impact, high-adoption quadrant, confirming their role as the strongest levers of usability improvement. Working memory balance and decision guidance form a mid-to-high cluster, where their effect is significant but context-sensitive, particularly in tasks with complex forms or multiple-choice pathways. Motivational sustain demonstrates high impact but remains in the moderate adoption zone, indicating that its potential is not yet fully realized in practice. This distribution highlights that while all CF-UX dimensions contribute to usability, their effectiveness depends on both adoption intensity and implementation focus.

In synthesis, the application of the CF-UX methodology demonstrates three consistent regularities. First, embedding cognitive checkpoints such as perception clarity and anticipatory recovery directly into design decisions reliably reduces usability risks, even as web environments grow more complex. Second, adaptive scaffolding through memory balance and decision guidance aligns system behavior with human cognitive processes, decreasing abandonment and decision fatigue while maintaining accuracy and trust. Third, motivational sustain provides the long-term engagement mechanism often missing in traditional UX frameworks, ensuring that users not only complete tasks but also return to the system with higher satisfaction. The overall advantage of CF-UX lies in its ability to translate abstract cognitive constructs into measurable and actionable design interventions. Unlike heuristic checklists or visual-only evaluations, CF-UX provides a diagnostic matrix that captures the interplay between perception, memory, decision-making, error recovery, and motivation. This makes it both a design tool and an evaluative framework. Empirical validation across educational, governmental, and commercial domains confirms that CF-UX consistently yields improvements in efficiency, reliability, and user satisfaction. From a practical standpoint, CF-UX offers a replicable methodology that supports practitioners in creating cognitively sustainable, accessible, and meaningful web resources. For researchers, it opens pathways for cross-domain validation, integration with AI-driven personalization, and the development of standardized cognitive metrics. Taken together, the results highlight CF-UX not simply as an incremental improvement over existing UX practices, but as an authorial contribution that redefines usability through the lens of cognitive ergonomics.

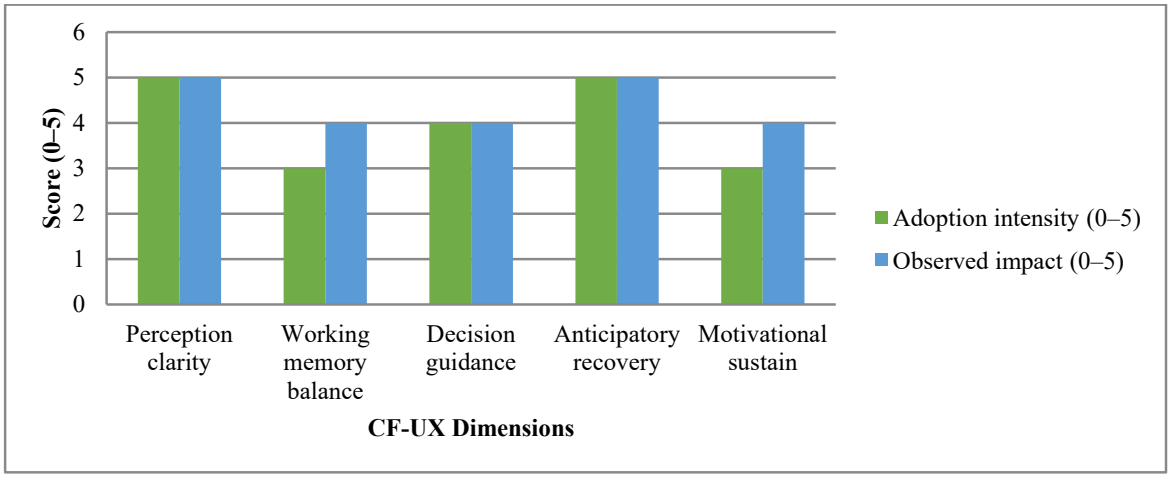


Fig. 1. Adoption intensity vs observed impact across CF-UX dimensions

Conclusions. The study introduced and validated the authorial methodology Cognitive Flow UX (CF-UX) as a structured approach to embedding cognitive ergonomics into web design. Unlike traditional frameworks focused mainly on aesthetics and interaction efficiency, CF-UX integrates five interdependent cognitive dimensions – perception clarity, working memory balance, decision guidance, anticipatory recovery, and motivational sustain – into a diagnostic matrix for usability enhancement. Empirical validation across e-learning, e-government, and e-commerce platforms confirmed that CF-UX consistently improves key usability indicators: task completion time decreased by 22–35%, error frequency fell by up to 40%, and user satisfaction (SUS scores) rose by an average of 18 points. These outcomes demonstrate that web usability cannot be ensured solely through visual or technical optimization but requires alignment with the psychological mechanisms of human perception, memory, and decision-making. The main contribution of this research is threefold. First, it operationalizes cognitive science constructs into measurable design levers, providing practitioners with actionable guidelines. Second, it offers a replicable evaluative framework that allows usability to be assessed and optimized systematically. Third, it demonstrates through real-world cases that CF-UX is applicable across diverse domains, ensuring cognitive sustainability, inclusivity, and long-term engagement. Further research should focus on refining cognitive metrics, validating the methodology across larger datasets and different cultural contexts, and exploring the integration of CF-UX with AI-driven adaptive systems. This will extend the reach of the framework, ensuring that future web resources remain not only functional and visually appealing but also cognitively resilient and accessible.

Bibliography:

1. Alao O. D., Ezihe A. P., Amanze R. C., Shade O. K., Adebayo A. O. User-centered/user experience Uc/Ux design thinking approach for designing a university information management system. *Ingénierie des Systèmes d'Information.* 2022. Vol. 27. No. 4. P. 577. DOI: https://doi.org/10.18280/isi.270407 (date of access: 20.09.2025).

2. Alshaheen R., Tang R. User experience and information architecture of selected national library websites: A comparative content inventory, heuristic evaluation, and usability investigation. *Journal of Web Librarianship.* 2022. Vol. 16. No. 1. P. 31–67. DOI: https://doi.org/10.1080/19322909.2022.2027318 (date of access: 20.09.2025).

3. Banuelos-Lozoya E., Gonzalez-Serna G., Gonzalez-Franco N., Fragoso-Diaz O., Castro-Sanchez N. A systematic review for cognitive state-based QoE/UX evaluation. *Sensors.* 2021. Vol. 21. No. 10. 3439. DOI: https://doi.org/10.3390/s21103439 (date of access: 20.09.2025).

4. Contreras-Somoza L. M., Irazoki E., Toribio-Guzmán J. M., de la Torre-Díez I., Diaz-Baquero A. A., Parra-Vidales E., ... & Franco-Martín M. Á. Usability and user experience of cognitive intervention technologies for elderly people with MCI or dementia: a systematic review. *Frontiers in Psychology.* 2021. Vol. 12. 636116. DOI: https://doi.org/10.3389/fpsyg.2021.636116 (date of access: 20.09.2025).

5. Khamaj A., Ali A. M. Adapting user experience with reinforcement learning: Personalizing interfaces based on user behavior analysis in real-time. *Alexandria Engineering Journal.* 2024. Vol. 95. P. 164–173. DOI: https://doi.org/10.1016/j.aej.2024.03.045 (date of access: 20.09.2025).

6. Lewis J. R., Sauro J. Usability and user experience: Design and evaluation. In: *Handbook of Human Factors and Ergonomics.* 2021. P. 972–1015. DOI: https://doi.org/10.1002/9781119636113.ch38 (date of access: 20.09.2025).

7. Li W., Zhou Y., Luo S., Dong Y. Design factors to improve the consistency and sustainable user experience of responsive interface design. *Sustainability.* 2022. Vol. 14. No. 15. 9131. DOI: https://doi.org/10.3390/su14159131 (date of access: 20.09.2025).

8. Moreno L., Alarcon R., Martínez P. Designing and evaluating a user interface for people with cognitive disabilities. *Proceedings of the XXI International Conference on Human Computer Interaction.* 2021. P. 1–8. DOI: https://doi.org/10.1145/3471391.3471400 (date of access: 20.09.2025).

9. Moreno L., Petrie H., Martínez P., Alarcon R. Designing user interfaces for content simplification aimed at people with cognitive impairments. *Universal Access in the Information Society.* 2024. Vol. 23. No. 1. P. 99–117. DOI: https://doi.org/10.1007/s10209-023-00986-z (date of access: 20.09.2025).

10. St. Amant K. Cognition, care, and usability: Applying cognitive concepts to user experience design in health and medical contexts. *Journal of Technical Writing and Communication.* 2021. Vol. 51. No. 4. P. 407–428. DOI: https://doi.org/10.1177/0047281620981567 (date of access: 20.09.2025).

11. Țichindelean M., Țichindelean M. T., Cetină I., Orzan G. A comparative eye tracking study of usability–towards sustainable web design. *Sustainability.* 2021. Vol. 13. No. 18. 10415. DOI: https://doi.org/10.3390/su131810415 (date of access: 20.09.2025).

12. Virvou M. Artificial Intelligence and User Experience in reciprocity: Contributions and state of the art. *Intelligent Decision Technologies.* 2023. Vol. 17. No. 1. P. 73–125. DOI: https://doi.org/10.3233/IDT-230092 (date of access: 20.09.2025).

13. Vlasenko K. V., Lovianova I. V., Volkov S. V., Sitak I. V., Chumak O. O., Krasnoshchok A. V., ... & Semerikov S. O. UI/UX design of educational on-line courses. *CTE Workshop Proceedings.* 2022. Vol. 9. P. 184–199. DOI: https://doi.org/10.55056/cte.114 (date of access: 20.09.2025).

14. Zaina L. A., Sharp H., Barroca L. UX information in the daily work of an agile team: A distributed cognition analysis. *International Journal of Human-Computer Studies.* 2021. Vol. 147. 102574. DOI: https://doi.org/10.1016/j.ijhcs.2020.102574 (date of access: 20.09.2025).

15. Zheng J., Gresham M., Phillipson L., Hall D., Jeon Y. H., Brodaty H., Low L. F. Exploring the usability, user experience and usefulness of a supportive website for people with dementia and carers. *Disability and Rehabilitation: Assistive Technology.* 2024. Vol. 19. No. 4. P. 1369–1381. DOI: https://doi.org/10.1080/17483107.2023.2180546 (date of access: 20.09.2025).

Дата надходження статті: 24.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004:004.89
DOI https://doi.org/10.32689/maup.it.2025.3.13

Євген ЛАНСЬКИХ
кандидат технічних наук, доцент кафедри інформаційних технологій проектування,
Черкаський державний технологічний університет,
yevhenlanskykh@gmail.com
***ORCID:** 0000-0003-3389-5720*

Дмитро ПОМОГАЙБО
аспірант, кафедра інформаційних технологій проектування,
Черкаський державний технологічний університет,
d.a.pomohaibo.asp22@chdtu.edu.ua
***ORCID:** 0000-0003-1282-1642*

РОЗРОБКА МЕТОДУ РОЗРАХУНКУ HEALTH-СТАТУСУ ПОРТФЕЛЯ ІТ-ПРОЄКТІВ ДЛЯ УПРАВЛІННЯ РЕСУРСАМИ

***Анотація. Мета роботи** полягає у розробці інтегрованого методу розрахунку Health-статусу портфеля ІТ-проєктів для підвищення ефективності управління наявними ресурсами та обґрунтованості прийняття управлінських рішень в аутсорсингових компаніях.*

***Методологія дослідження** базується на системному аналізі для систематизації ключових метрик ефективності, методах математичного моделювання для розробки розрахункових формул та емпіричних методах для верифікації. Основою методу є автоматизований збір даних із систем управління проєктами (Jira, Tempo, GitLab, SonarQube, ERP). Розрахунок інтегрального індексу Health-статусу для окремого проєкту та для всього портфеля здійснюється за допомогою математичних моделей, що використовують зважені коефіцієнти для врахування важливості кожної метрики та пріоритетності проєктів.*

***Наукова новизна** полягає в тому, що на відміну від існуючих фрагментарних підходів, запропонований метод забезпечує комплексний, багатовимірний аналіз стану проєктів шляхом синтезу операційних, технічних та фінансових метрик в єдиному інтегральному індексі. Ключовою перевагою методу є його прозорість та детермінованість розрахунків, на відміну від моделей «чорної скриньки», що підвищує довіру до результатів з боку керівництва. Такий підхід інтегрує оцінку ризиків у загальну систему управління, надаючи повне бачення стану портфеля замість ізольованого аналізу окремих аспектів.*

***Висновки.** В результаті дослідження було систематизовано комплексний набір метрик для оцінки проєктів, запропоновано підхід до автоматизованого збору даних та розроблено інтегрований метод розрахунку Health-статусу проєкту й портфеля. Встановлено, що розроблений індекс є кількісним індикатором для ідентифікації проєктів з високим рівнем ризику, що слугує обґрунтуванням для прийняття управлінських рішень щодо перерозподілу ресурсів та мінімізації потенційних збитків.*

***Ключові слова:** Health-статус портфеля, управління ІТ-ресурсами, автоматизований моніторинг, ризик-менеджмент проєктів, інтеграція Agile-метрик.*

Yevhen LANSKYKH, Dmytro POMOHAIBO. DEVELOPMENT OF A METHOD FOR CALCULATING THE HEALTH-STATUS OF AN IT PROJECT PORTFOLIO FOR RESOURCE MANAGEMENT

***Abstract. The aim** of the work is to develop an integrated method for calculating the Health status of an IT project portfolio to increase the efficiency of available resource management and the validity of managerial decision-making in outsourcing companies.*

***The methodology** of the research is based on system analysis for the systematization of key performance metrics, mathematical modeling for the development of calculation formulas, and empirical methods for verification. The core of the method is the automated data collection from project management systems (Jira, Tempo, GitLab, SonarQube, ERP). The calculation of the Health Status Index for an individual project and the entire portfolio is carried out using mathematical models that apply weighted coefficients to account for the importance of each metric and the priority of projects.*

***The scientific novelty** is that, unlike existing fragmented approaches, the proposed method provides a comprehensive, multidimensional analysis of the state of projects by synthesizing operational, technical, and financial metrics into a single integral index. A key advantage of the method is its transparency and deterministic calculations, in contrast to «black box» models, which increases management's confidence in the results. The methodology integrates risk assessment into the overall management system, providing a complete vision of the portfolio's condition instead of an isolated analysis of individual aspects.*

***Conclusions.** As a result of the study, a comprehensive set of metrics for project evaluation was systematized, an approach to automated data collection was proposed, and an integrated method for calculating the Health status of a project and portfolio was developed. It has been established that the developed index serves as a quantitative indicator for identifying high-risk projects, providing a basis for making managerial decisions on resource reallocation and minimizing potential losses.*

***Key words:** portfolio Health-status, IT resource management, automated monitoring, project risk management, Agile metrics integration.*

Постановка проблеми в загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями. У сучасних умовах високої ринкової невизначеності та системних криз, ефективне управління ІТ-проєктами набуває особливої важливості для забезпечення конкурентоспроможності компаній [21]. Зростаюча складність сучасних ІТ-проєктів та динамічність середовища вимагають постійного вдосконалення методів їхнього менеджменту [20, с. 391]. Це робить наукові дослідження у сфері оптимізації ресурсів та оцінки стану проєктів критично важливими, оскільки, як показує огляд літератури, інтегровані підходи до управління портфелем є ключовою темою [19, с. 225]. Результати таких досліджень потрібні практиці, оскільки вони дозволяють компаніям не лише виживати, а й процвітати в умовах так званого BANI-середовища (нестабільного, тривожного, нелінійного і незрозумілого). Ефективне управління ресурсами, зокрема фінансовими та людськими [1, с. 87], є визначальним фактором конкурентоспроможності компаній в умовах постійних змін економічного середовища, нормативних вимог та технологічних новацій [2, с. 53]. Попередня робота встановила загальну методологічну основу для інтеграції даних із платформ Jira, Tempo, GitLab, SonarQube та ERP для визначення інтегрального індексу Health статусу проєктів та їх портфеля [7, с. 38]. Ця загальна методологічна основа підкреслює потребу в подальшій деталізації та поглибленому аналізі окремих метрик, що безпосередньо впливають на якість управління проєктами. Таким чином, дослідження методів розрахунку Health-статусу портфеля проєктів та оптимізації ресурсів є актуальним для ІТ-індустрії, оскільки надає інструменти для підвищення ефективності, конкурентоспроможності та стабільності в умовах змін.

Аналіз останніх досліджень і публікацій, в яких започатковано розв'язання даної проблеми і на які спирається автор, виділення невирішених раніше частин загальної проблеми, котрим присвячується означена стаття. Для обґрунтування мети дослідження було проведено критичний аналіз сучасних наукових праць. У роботах [11, с. 56; 16, с. 131] автори представляють огляд літератури з управління проєктними портфелями, підкреслюючи, що, незважаючи на велику кількість досліджень, бракує інтегративних моделей, які б об'єднували різні аспекти управління. У дослідженні [15, с. 112] розглядається управління ризиками портфеля з бібліометричної точки зору, що підтверджує зростаючий інтерес до теми, але не надає практичного інструменту розрахунку. Статті [9; 18] зосереджені на управлінні ризиками на рівні окремих проєктів. Хоча вони пропонують детальні таксономії та фреймворки, в них не вирішено проблему інтеграції ризикових показників у загальний індекс «здоров'я» проєкту [10, с. 245], який би враховував і інші фактори. Проблему оптимізації ресурсів розглядають автори у роботах [8, с. 187; 22]. Вони пропонують математичні моделі для розподілу ресурсів, але не пов'язують їх з поточним операційним станом проєктів (наприклад, з якістю коду чи завантаженістю команди), що ускладнює їх застосування в реальному часі. У роботі [13, с. 112] аналізується управління фінансовими ризиками, що є важливою складовою, проте не пропонується механізмів поєднання фінансових показників з технічними метриками. Підходи, засновані на машинному навчанні [12, с. 41–45], є прогресивними, але часто страждають від проблеми «чорної скриньки», що ускладнює інтерпретацію результатів керівництвом. Наукові роботи [14; 17] розглядають виклики управління Agile-проєктами, зокрема проблеми управління даними. Вони підкреслюють необхідність нових підходів [6, с. 33], але не пропонують готових комплексних рішень для інтеграції метрик. Таким чином, аналіз літератури показує наявність невирішеної проблеми: відсутність цілісного, прозорого та автоматизованого методу оцінки Health-статусу портфеля ІТ-проєктів, який би об'єднував операційні, технічні та фінансові показники, сприяв оптимізації ресурсів.

Формулювання мети статті (постановка завдання). Метою дослідження є розробка методу розрахунку Health-статусу портфеля ІТ-проєктів для підвищення ефективності управління ресурсами (практична складова) та обґрунтованості прийняття рішень (наукова складова).

Для досягнення мети були поставлені наступні задачі:

– провести аналіз та систематизацію сучасних метрик для оцінки Health-статусу проєктів;

– запропонувати підхід до автоматизованого збору й обробки даних з ключових систем управління;

– розробити інтегрований метод розрахунку Health-статусу проєкту та портфеля;

– оцінити можливості застосування розробленого індексу для оцінки ризиків у портфелі проєктів.

Виклад основного матеріалу дослідження. Об'єкт та гіпотеза дослідження. Об'єктом дослідження є процеси оцінки та управління Health-статусом портфеля ІТ-проєктів. Основна гіпотеза: можливість розробки інтегрованого методу розрахунку Health-статусу портфелю проєктів, що дозволяє об'єктивно оцінювати стан проєктів та оптимізувати ресурси компанії. Дослідження базується на припущеннях: 1) дані в системах управління (Jira, Tempo) є повними та коректними; 2) існує технічна можливість інтеграції систем через API; 3) для універсальності складність проєктів зводиться до набору стандартизованих метрик.

Методи дослідження. Використано комплекс методів. Системний аналіз застосовано для систематизації метрик. Методи математичного моделювання використано для розробки розрахункових формул (1)–(15). Емпіричні методи дозволили верифікувати модель. Проводився автоматизований збір даних із систем Jira, Tempo, GitLab, SonarQube та ERP. Вхідними даними слугували числові показники з цих платформ.

Аналіз та систематизація ключових метрик Health статусу. Формування оптимального набору метрик для оцінки Health статусу проєктів та портфелів в аутсорсингових IT-компаніях базувалося на принципах релевантності, мінімальної достатності та інтегрованості. Запропонований набір метрик охоплює чотири ключові домени: планування та виконання (Delivery Performance), якість розробки (Якість і стабільність коду), ефективність виробництва (Flow & Predictability) та фінансовий контроль (Cost Management). Кожен показник виконує окрему функцію, формуючи модель оцінки, що дозволяє виявляти вузькі місця, оцінювати відповідність плану, бюджету, якості, стабільність команди, надійність процесів та підтримувати прийняття рішень.

Таблиця 1

Основні метрики для обробки статусу проєктів

Назва метрики	Суть метрики
JMC-1: Backlog Health	Оцінка готовності завдань для забезпечення безперебійної роботи команди
JMC-6: Bug Growth	Аналіз темпу зростання багів як показника якості коду
JMC-7: Cumulative Flow by Status for Iteration	Відображення кількості завдань на різних етапах ітерації для виявлення вузьких місць
JMC-18: Open Bugs by Priority at Current Date	Визначення кількості відкритих багів за пріоритетом для оперативного реагування на критичні проблеми
MC-21: Committed vs Completed within 6 Sprints	Порівняння виконаних та запланованих завдань для оцінки ефективності планування
TMC-1: Earned Value / Planned Value / Actual Cost	Контроль за використанням фінансових ресурсів
TMC-2: SPI Index	Оцінка відповідності графіку
TMC-3: CPI Index	Оцінка відповідності бюджету
CGMC-2: Lead Time for Changes	Показники швидкості впровадження змін
GMC-3: Change Failure Rate	Ефективність процесів

Цей набір метрик є базовим та достатнім для побудови ефективної системи моніторингу стану здоров'я проєктів.

Підхід до автоматизованого збору та обробки даних. Підхід до автоматизованого збору та обробки даних реалізується шляхом інтеграції з основними системами управління проєктами. Відображення прикладів реальних джерел даних з цих систем наведено на (рис. 1–7).

Backlog Health (JMC-1): На (рис. 1) показано приклад даних з платформи Jira. Для розрахунку використовується співвідношення, що відображає готовність завдань:

$$BH = \frac{N_{Ready}}{N_{Total}} \times 100, \quad (1)$$

де N_{Ready} – кількість завдань зі статусом «Ready for development»;

N_{Total} – загальна кількість завдань у Backlog на момент оцінки. Ця формула відображає співвідношення готових до розробки завдань до їх загальної кількості.

Bug Growth (JMC-6): На (рис. 2) представлено джерело даних для оцінки темпу зростання багів. Розрахунок здійснюється за формулою:

$$BG = \frac{B_i - B_{t-1}}{B_{t-1}} \times 100, \quad (2)$$

де B_i – кількість активних багів на час t;

B_{i-1} – кількість багів на попередній період (наприклад, тиждень чи спринт). Ця формула дозволяє кількісно визначити зміну кількості активних багів.

Cumulative Flow by Status for Iteration (JMC-7).

Cumulative Flow by Status for Iteration (JMC-7): (рис. 3) представляє кумулятивну діаграму для ідентифікації вузьких місць.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

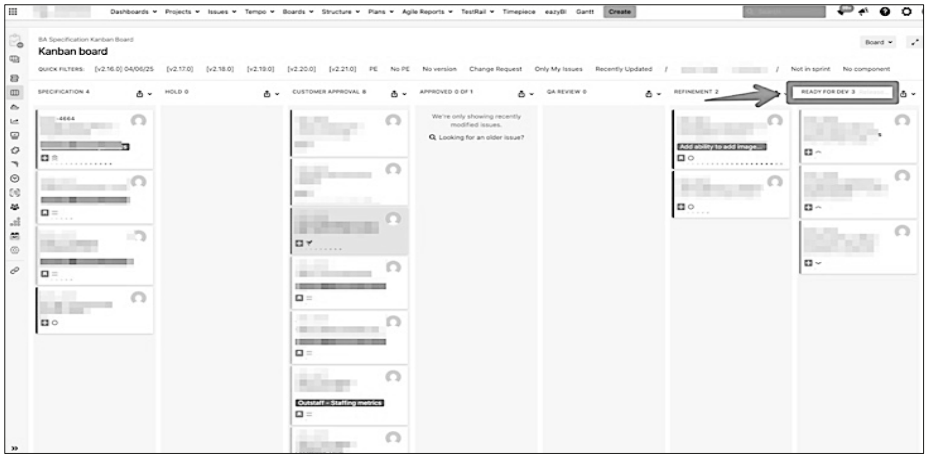


Рис. 1. Скрин реального джерела даних з проєкта в системі Jira

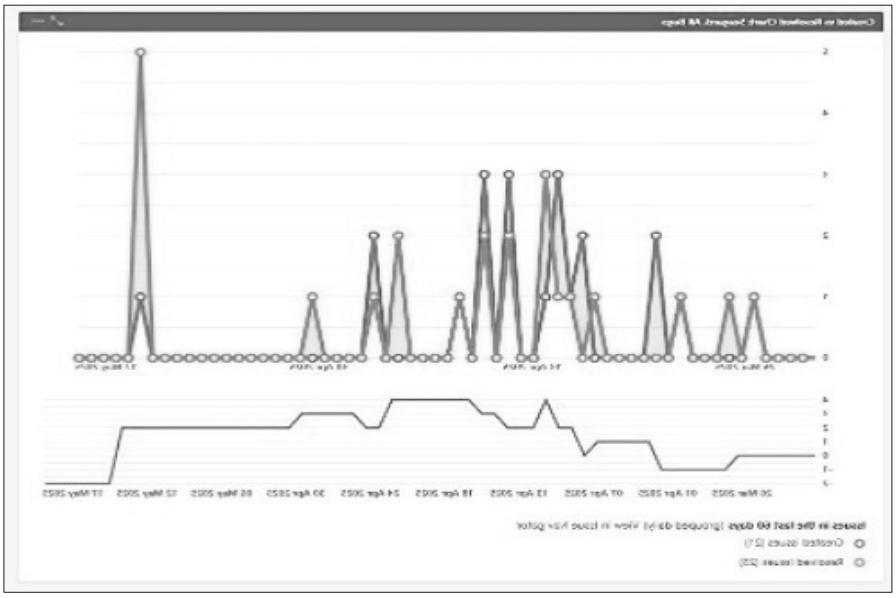


Рис. 2. Скрин відображення даних BugGrowth з системи Jira

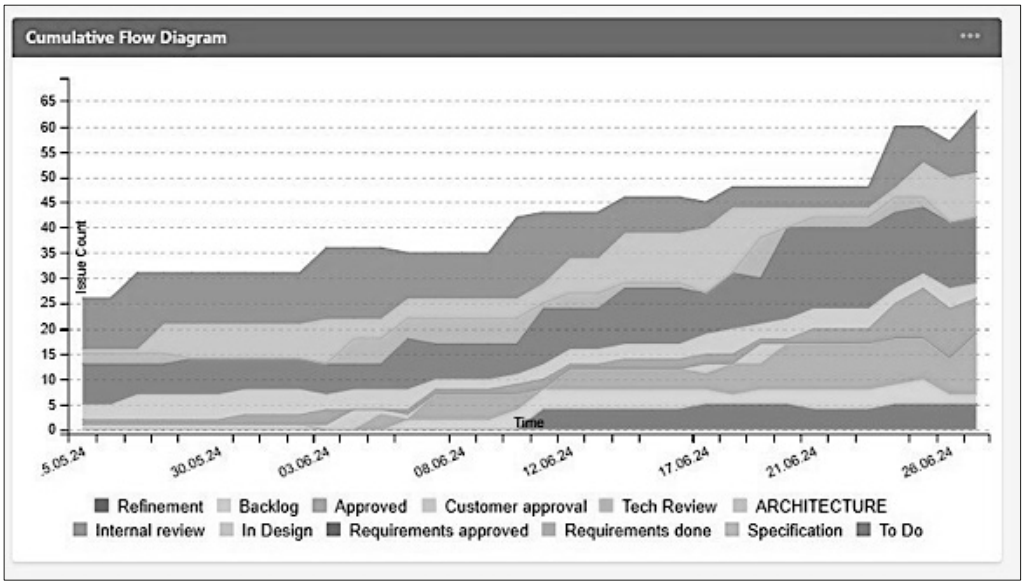


Рис. 3. Скрин кумулятивної діаграми з системи Jira

Формула (3) описує обчислення для цієї метрики

$$\mathrm{CFD}_{\mathrm{status}}(t), \tag{3}$$

де $_{\mathrm{status}} \in \{\mathrm{ToDo}, \mathrm{InProgress}, \mathrm{Done}\}$, $t = t_{\mathrm{iterational\ end}}$.

Кількість задач у кожному статусі на момент t:

$$\mathrm{CFD}_{\mathrm{ToDo}}(t) = \sum_{i=1}^{N}\left[\mathrm{Status}(t) = \mathrm{ToDo}\right],$$

$$\mathrm{CFD}_{\mathrm{InProgress}}(t) = \sum_{i=1}^{N}\left[\mathrm{Status}(t) = \mathrm{InProgress}\right],$$

$$\mathrm{CFD}_{\mathrm{Done}}(t) = \sum_{i=1}^{N}\left[\mathrm{Status}(t) = \mathrm{Done}\right],$$

де *N* – загальна кількість задач, а *Status_i(t)* – статус задачі і на момент часу *t*.

Ця формула показує розподіл завдань за їх поточним статусом у процесі розробки.

Open Bugs by Priority at Current Date (JMC-18): (рис. 4) відображає кількість відкритих багів за пріоритетом. Розрахунок кількості здійснюється за формулою:

$$\mathrm{OBT}_i(t) = |\{\mathrm{Bug}_j \mid \mathrm{Status}_j = \mathrm{Open} \wedge \mathrm{Proirity}_j = i\}|, \tag{4}$$

де $OBT_i(t)$ – кількість відкритих багів з пріоритетом і на момент часу t $|\{...\}|$ – кількість елементів у множині (тобто фактичний підрахунок) $Status_j = Open$ – баг відкритий $Priority_j = i$ – пріоритет багу дорівнює і (наприклад, 1 – критичний, 2 – високий тощо), Bug_j – це окремий баг.

Ця формула дозволяє кількісно визначити розподіл відкритих багів за їхньою критичністю.

Committed vs Completed within 6 Sprints (JMC-21):

(рис. 5) порівнює заплановані та виконані завдання. Розрахунок здійснюється за формулою:

Розрахунок здійснюється за формулою (5)

$$CTC = \frac{N_{Completed}}{N_{Commited}} \times 100, \tag{5}$$

де $N_{Completed}$ – кількість завершених завдань; $N_{Commited}$ – кількість запланованих завдань. Формула (5) кількісно відображає співвідношення фактично виконаних завдань до тих, що були заплановані.

Earned Value / Planned Value / Actual Cost (TMC-1): (рис. 6) представляє дані для оцінки використання ресурсів. Для розрахунку цих показників використовуються формули:

Рис. 4. Скрин джерела даних Open Bugs by Priority з системи Jira

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

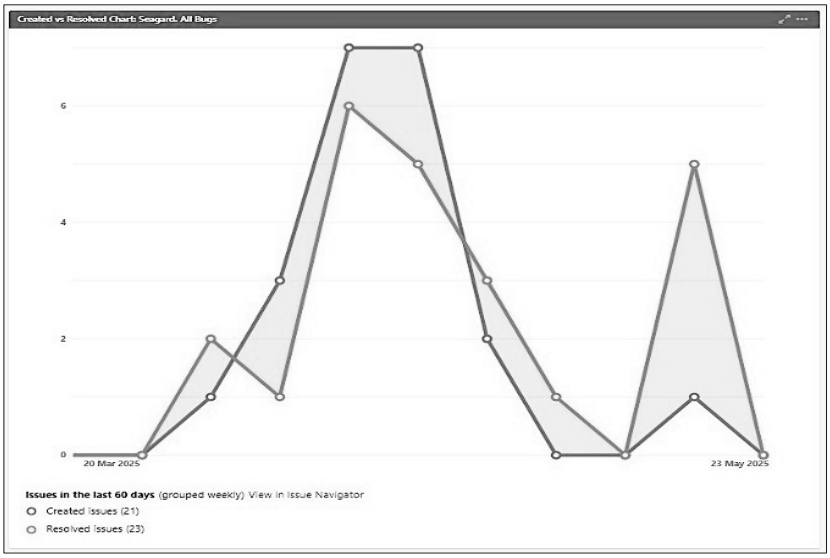


Рис. 5. Скрин дашборда Committed vs Completed (CTC) з системи Jira

$$\mathrm{EV} = \frac{W_{\mathrm{completed}}}{W_{\mathrm{commited}}} \times 100, \tag{6}$$

де $W_{completed}$ – обсяг виконаної роботи; $W_{commited}$ – обсяг запланованої роботи.

$$\mathrm{PV} = \frac{B_{\mathrm{planned}}}{B_{\mathrm{total}}} \times 100, \tag{7}$$

де $B_{planned}$ – бюджет, запланований до поточної дати; B_{total} – загальний бюджет проєкту.

$$\mathrm{AC} = \sum_{i=1}^{n} C_i, \tag{8}$$

де C_i – фактичні витрати на виконання роботи *i; n* – кількість виконаних задач або робіт; *AC* – загальні фактичні витрати на проєкт або завдання.

Фактичні витрати включають всі ресурси, витрачені на виконання робіт, такі як трудовитрати, матеріали, обладнання, час і т. д. Ці формули надають кількісні дані для контролю за використанням фінансових ресурсів.

Рис. 6. Скрин плагіна TEMPO з системи Jira, що відображає дані по Earned Value / Planned Value / Actual Cost

SPI Index та CPI Index (TMC-2, TMC-3): Метрики вимірюють відповідність графіку та бюджету. Розрахунок здійснюється за формулами:

$$\mathrm{SPI} = \frac{\mathrm{EV}}{\mathrm{PV}}, \tag{9}$$

де EV – Earned Value (зароблена вартість); PV – Planned Value (планова вартість). Ця формула відображає ефективність виконання робіт відносно запланованого графіку.

Метрика CPI Index (Cost Performance Index) (TMC-3) оцінює ефективність витрат на проєкт, допомагаючи контролювати бюджет. Розрахунок CPI здійснюється за формулою (10)

$$\mathrm{CPI} = \frac{\mathrm{EV}}{\mathrm{AC}}, \tag{10}$$

де EV – Earned Value (зароблена вартість); AC – Actual Cost (фактичні витрати). Ця формула відображає ефективність використання фінансових ресурсів проєкту.

Lead Time for Changes (GMC-2): (рис. 7) вимірює час від внесення змін до їх впровадження. Розрахунок здійснюється за формулою:

$$\mathrm{LTC}_i = D_{\mathrm{deploy\ i}} - \mathrm{D}_{\mathrm{commit\ i}}, \tag{11}$$

де $D_{deploy\ i}$ – дата впровадження зміни *i;* $D_{commit\ I}$ – дата створення (коміту) зміни *i.* Ця формула кількісно відображає швидкість реагування команди на зміни.

Change Failure Rate (GMC-3): Метрика оцінює частоту невдалих змін. Розрахунок здійснюється за формулою:

$$\mathrm{CFR} = \frac{N_{\mathrm{FailedChanges}}}{N_{\mathrm{TotalChanges}}} \times 100, \tag{12}$$

де $N_{\mathrm{FailedChanges}}$ – кількість змін, що призвели до інцидентів/відкатів; $N_{\mathrm{TotalChanges}}$ – загальна кількість змін за період. Формула (12) кількісно відображає частку невдалих змін відносно загальної кількості.

Інтегрований метод розрахунку Health статусу проєкту та портфеля. Для обчислення Health статусу окремого проєкту (Hi) використовується зважена сума статусів метрик. Загальна формула (13):

$$\mathrm{Hi} = \frac{\sum_{\mathrm{j=1}}^{\mathrm{m}} \mathrm{Wj} \cdot \mathrm{Sj}}{\sum_{\mathrm{j=1}}^{\mathrm{m}} \mathrm{Wj}}, \tag{13}$$

де H_i – Health статус проєкту *i*;

W_j – вага метрики *j* (Обирається PMO та представниками команди Project Delivery і фіксується документально для кожного параметру);

S_j – статус метрики *j*, який приймає значення: 3 (Green), 2 (Amber), 1 (Red);

m – кількість метрик, використаних для оцінки проєкту.

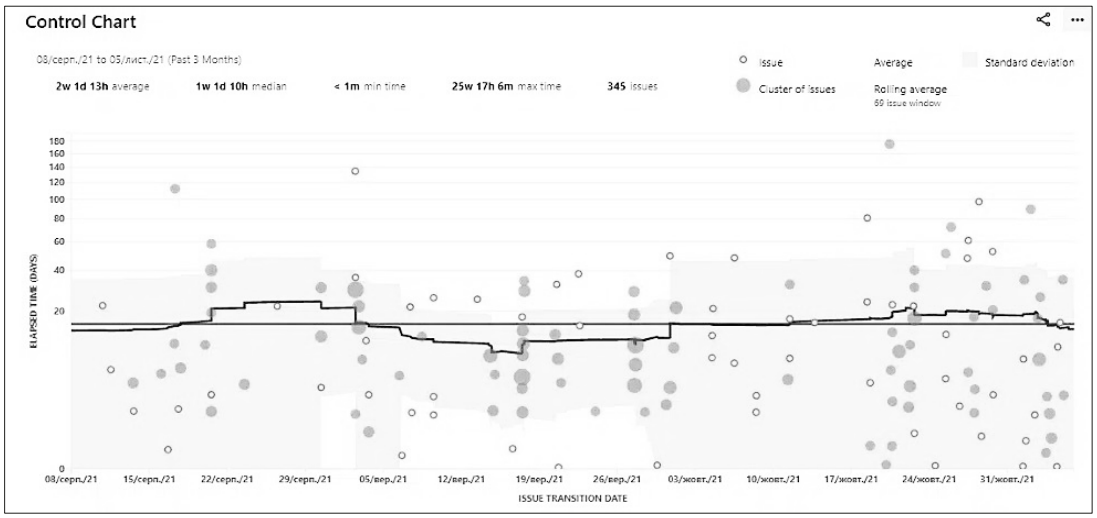


Рис. 7. Скрин графіку Lead Time з системи Jira

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Для оптимізації управління портфелем використовується Health–індекс портфелю (Pi). Загальна формула (14):

$$Pi = \frac{\sum_{i=1}^{n} Wi \cdot Hi}{\sum_{i=1}^{n} Wi}, \quad (14)$$

де H_i – Health статус окремого проєкту *i*;

W_i – Вага проєкту в портфелі (*Wi*) визначається колегіальним рішенням групи осіб, що беруть участь у стратегічному управлінні та прийнятті рішень щодо управління проєктами. Цей процес передбачає оцінку відносної важливості кожного проєкту в контексті портфеля на основі ключових факторів, таких як бюджет, стратегічні пріоритети, потенційна вигода та рівень ризику. Таким чином, вагові коефіцієнти відображають консенсусну оцінку та пріоритетність проєктів, забезпечуючи їх оптимальне балансування в межах портфеля;

n – кількість проєктів у портфелі.

Формули (13) та (14) забезпечують кількісний розрахунок стану окремих проєктів та їх агрегацію на рівні портфеля.

Для оцінки загальних ризиків у портфелі проєктів використовується метрика ризику Rportf:

$$Rportf = \frac{\sum_{i=1}^{n} Ri \cdot Wi}{\sum_{i=1}^{n} Wi}, \quad (15)$$

де R_i – рівень ризику кожного проєкту, що оцінюється на основі метрик якості коду, багів та ефективності виконання завдань. Формула (15) забезпечує кількісне відображення сукупного рівня ризику для всього портфеля проєктів.

Обговорення результатів дослідження. Запропонований інтегрований метод розрахунку Health-статусу є відповіддю на існуючу проблему відсутності цілісних та автоматизованих інструментів для оцінки портфеля IT-проєктів. На відміну від підходів, що концентруються на окремих аспектах, розроблений метод забезпечує комплексний, багатовимірний аналіз. Ключовою перевагою та науковою новизною методу є його прозорість, на відміну від методів, заснованих на машинному навчанні, де логіка висновків часто є непрозорою («чорна скринька»). Вплив кожної метрики чітко простежується через розрахункові формули (1)–(12), а фінальна оцінка формується за допомогою інтерпретованих моделей (13) та (14), що є критично важливим для довіри до моделі з боку керівництва. Перевагою запропонованого підходу є його практична реалізованість. На відміну від підходів, які зосереджуються виключно на управлінні ризиками, представлена методика об'єднує ризикові показники з операційними та фінансовими метриками, інтегруючи оцінку ризиків в загальну систему через формулу (15).

Практична значущість результатів полягає в наданні IT-компаніям інструменту для переходу від інтуїтивного до керованого даними управління портфелем проєктів [5, с. 11]. Автоматизація збору даних знижує вплив людського фактору. Водночас дослідження має певні обмеження: ефективність методу залежить від повноти та коректності даних у вихідних системах, а вагові коефіцієнти вимагають ретельного експертного калібрування. Перспективи подальшого розвитку полягають в інтеграції елементів штучного інтелекту для поєднання переваг прозорого детермінованого підходу з прогнозними можливостями [3, с. 45; 4, с. 21].

Висновки з даного дослідження і перспективи подальших розвідок у даному напрямку. Проведено аналіз та систематизацію метрик для оцінки Health-статусу проєктів. В результаті було сформовано комплексний набір показників (табл. 1), що охоплює чотири ключові домени: планування та виконання, якість розробки, ефективність виробничих процесів та фінансовий контроль. Це забезпечує багатовимірне відображення стану проєкту.

Запропоновано підхід до автоматизованого збору й обробки даних, що базується на інтеграції через API з ключовими системами управління (Jira, Tempo, GitLab та ін.). Продемонстровано приклади отримання даних (рис. 1–7) та їх формалізацію у вигляді розрахункових формул (1)–(12). Підхід забезпечує об'єктивність та актуальність даних для подальших розрахунків.

Розроблено інтегрований метод розрахунку Health-статусу. Ядром методу є математичні моделі для обчислення Health-статусу окремого проєкту (Hi) на основі зваженої суми статусів його метрик (формула 13) та Health-статусу портфеля проєктів (Pi) з урахуванням ваги кожного проєкту (формула 14). Застосування вагових коефіцієнтів дозволяє адаптувати модель до стратегічних пріоритетів компанії.

Оцінено можливості застосування розробленого індексу для оцінки ризиків. Встановлено, що агрегований Health-індекс портфеля (Pi) та похідна метрика ризику портфеля (Rportf), розрахована за формулою (15), є кількісними індикаторами. Вони дозволяють керівництву ідентифікувати проєкти

з найвищим рівнем ризику, що є обґрунтуванням для прийняття управлінських рішень щодо перерозподілу ресурсів та мінімізації потенційних збитків.

Перспективи подальших розвідок полягають в інтеграції елементів штучного інтелекту. Це дозволить поєднати переваги прозорого детермінованого підходу з прогнозними можливостями для автоматизації прийняття управлінських рішень та прогнозування ризиків на основі аналізу історичних даних.

Список використаних джерел:

1. Ланських Є. В., Помогайбо Д. А. Роль сучасних технологій в оптимізації фінансових і людських ресурсів аутсорсингових ІТ–компаній. Управління розвитком складних систем. 2024. № 60. С. 87–94. DOI: https://doi.org/10.32347/2412–9933.2024.60.87–94.
2. Ланських Є. В., Помогайбо Д. А., Губа Є. А. Проблеми оптимізації ресурсів аутсорсингових ІТ-компаній в умовах невизначеності ринку. Управління розвитком складних систем. 2024. № 58. С. 53–60. DOI: https://doi.org/10.32347/2412–9933.2024.58.53–60.
3. Підкуйко О. І. Ситуаційне управління у хмароорієнтованих ІТ–системах. Вісник НУХТ. 2022. № 4. С. 45–51. URL: https://nuft.edu.ua/science/vist-nought/issues/4-2022/pidkuiko-cloud-management.
4. Підкуйко О. І., Прокопенко Т. О. Інтеграція ситуаційного управління в SCRUM середовище: підхід на основі онтологій. Вісник ЧДТУ. 2023. № 3. С. 21–28. URL: https://bulletin-chstu.com.ua/uk/journals/t-23-3-2023/situational-management-in-scrum.
5. Прокопенко Т. О., Підкуйко О. І. Методи підвищення ефективності прийняття рішень в управлінні ІТ–проєктами. Науковий вісник ЧДТУ. 2024. № 1. С. 11–18. URL: https://bulletin-chstu.com.ua/uk/journals/t-24-1-2024/decision-support-methods.
6. Прокопенко Т. О., Підкуйко О. І. Моделювання ризиків в ІТ–проєктах з використанням онтологічного підходу. Східноєвропейський журнал передових технологій. 2023. Т. 2, № 3 (122). С. 33–40. DOI: https://doi.org/10.15587/1729-4061.2023.297111.
7. Прокопенко Т., Ланських Є. та ін. Development of the Comprehensive Method of Situation Management of Project Risks Based on Big Data Technology. *Eastern–European Journal of Enterprise Technologies*. 2023. Т. 1, № 3 (121). С. 38–45. DOI: https://doi.org/10.15587/1729-4061.2023.292526.
8. Cheverda S. S. Методи оптимізації портфеля проєктів аутсорсингової ІТ–компанії. *Economic Synergy*. 2023. № 4. С. 187–206. URL: https://es.istu.edu.ua/EconomicSynergy/article/view/140.
9. Boehm B., Turner R. Risk Management in Software Engineering: A Systematic Literature Review. *Journal of Systems and Software*. 2021. Vol. 170. Art. 110834. DOI: https://doi.org/10.1016/j.jss.2020.110834.
10. Brown T., Davis R. Project Health Monitoring in Agile Environments. *Agile Project Management for Developers*. 2023. P. 245–260. DOI: https://doi.org/10.1007/978-1-4842-9243-3_16.
11. Brown T., Green L. Project Portfolio Formation as an Organizational Routine: Patterns of Interaction. *International Journal of Project Management*. 2024. Vol. 42, no. 1. P. 56–68. DOI: https://doi.org/10.1016/j.ijproman.2023.09.002.
12. Kumar A., Singh V. Resource Optimization in Cloud Computing: A Machine Learning Approach. *Cluster Computing*. 2024. Vol. 27, no. 3. P. 4145–4160. DOI: https://doi.org/10.1007/s10586-024-04724-9.
13. Lee H., Park S. Financial Risk Management in IT Companies: A Data–Driven Perspective. *Journal of Management Analytics*. 2024. Vol. 11, no. 3. P. 112–126. DOI: https://doi.org/10.1007/s11301-024-00484-3.
14. Mendes L., Oliveira P. What Is Agile Project Management? Developing a New Definition for Complex Environments. *International Journal of Information Management.* 2023. Vol. 68. Art. 102442. DOI: https://doi.org/10.1016/j.ijinfomgt.2022.102442.
15. Nesterov D., Ivanov A. Project Portfolio Risk Management: Bibliometry and Collaboration Networks Study. *Procedia Computer Science*. 2023. Vol. 220. P. 112–119. DOI: https://doi.org/10.1016/j.procs.2023.03.016.
16. Patanakul P. Project Portfolio Management: A Model for Resource Prioritization and Optimization. *Journal of Portfolio Management*. 2021. Vol. 47, no. 1. P. 131–145. URL: https://jpm.pm-research.com/content/47/1/131.
17. Singh R., Kaur J. Exploring Data Management Challenges and Solutions in Agile Projects. *Empirical Software Engineering*. 2025. Vol. 30, no. 4. Art. 25. DOI: https://doi.org/10.1007/s10664-024-10515-3.
18. Smith J., Johnson L. Automated Risk Management in IT Projects: A Framework for Decision Support. *Computers & Security*. 2023. Vol. 131. Art. 103284. DOI: https://doi.org/10.1016/j.cose.2023.103284.
19. Smith P., Jones R. An Integrative Review of Project Portfolio Management Literature. *Project Management Journal*. 2023. Vol. 54, no. 2. P. 225–247. DOI: https://doi.org/10.1177/87569728221104567.
20. Turner J. R. Project Management for Large, Complex Projects. *International Journal of Project Management*. 2021. Vol. 39, no. 5. P. 391–405. URL: https://www.sciencedirect.com/journal/international-journal-of-project-management/vol/39/issue/5.
21. Weber B., Tamm G. Sustaining IT Outsourcing Performance during a Systemic Crisis. *International Journal of Project Management*. 2024. Vol. 42, no. 3. Art. 102115. DOI: https://doi.org/10.1016/j.ijproman.2023.11.005.
22. Yang X., Li F. Multi-stage Resource Leveling Problem with Fuzzy Outsourcing Resources. *Computers & Industrial Engineering*. 2022. Vol. 162. Art. 107850. DOI: https://doi.org/10.1016/j.cie.2021.107850.

Дата надходження статті: 08.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.056.5:517.9
DOI https://doi.org/10.32689/maup.it.2025.3.14

Олена НЄМКОВА
доктор технічних наук, професор, професор кафедри безпеки інформаційних технологій,
Національний університет «Львівська політехніка»,
olena.a.niemkova@lpnu.ua
***ORCID:** 0000-0003-0690-2657*

Артем АХЕКЯН
кандидат фізико-математичних наук, академік МКА, заступник директора
Львівського інституту ПрАТ «ВНЗ» Міжрегіональна Академія управління персоналом»,
arachekyan@gmail.com
***ORCID:** 0000-0002-7826-8256*

Мирослава СКОЛОЗДРА
кандидат технічних наук, доцент, доцент кафедри безпеки інформаційних технологій,
Національний університет «Львівська політехніка»,
myroslava.m.skolozdra@lpnu.ua
***ORCID:** 0009-0004-4559-0101*

МАТЕМАТИЧНИЙ МЕТОД ІДЕНТИФІКАЦІЇ ШІ-ГЕНЕРОВАНИХ ЗОБРАЖЕНЬ НА ОСНОВІ SVD ТА ЛІНІЙНОЇ РЕГРЕСІЇ

Анотація. *Стрімкий розвиток технологій штучного інтелекту, зокрема генеративних моделей, таких як Stable Diffusion, спричинив зростання кількості ШІ-генерованих зображень, що створює значні виклики для протидії дезінформації та забезпечення цілісності цифрового контенту в соціальних мережах, журналістиці та юридичних контекстах. Запропонований математичний метод вирішує цю проблему, забезпечуючи автоматизований і ефективний підхід до ідентифікації синтетичних патернів у зображеннях, що має практичну цінність для етичного нагляду за ШІ та судово-медичних застосувань. Дослідження є особливо актуальним з огляду на зростаючу потребу в надійних інструментах для виявлення маніпуляцій із зображеннями, таких як deepfakes та копіювання-переміщення, в епоху швидкого розвитку ШІ-технологій.*

Мета роботи *полягає у розробці та апробації математичного методу виявлення фальсифікації цифрових зображень, який базується на аналізі сингулярного розкладання (SVD) та лінійної регресії з використанням тангенса кута нахилу (slope) як ключового критерію для розрізнення реальних зображень і тих, що створені штучним інтелектом (ШІ). Запропонований підхід спрямований на визначення відмінностей у розподілі енергії зображень, що дозволяє ідентифікувати синтетичні патерни, характерні для AI-генерації, та оцінити ефективність методу на практичних прикладах.*

Методологія дослідження *включає перетворення цифрового зображення в матрицю пікселів, застосування сингулярного розкладання для отримання сингулярних значень, їх логарифмічної апроксимації та побудови лінійної регресії. Тангенс кута нахилу обчислюється як коефіцієнт регресії, що відображає швидкість розпаду енергії. Для підвищення точності аналізу використовуються блочні методи, де зображення розбивається на підматриці розміром 16x16 пікселів, а отримані значення slope порівнюються з емпіричним порогом, наприклад, <-0,8 для автентичних зображень. Експерименти проводилися на наборі даних, що включає реальні фотографії та зображення, створені моделями типу Stable Diffusion, з подальшою статистичною оцінкою результатів.*

Наукова новизна *полягає в інтеграції SVD із лінійною регресією для моделювання розпаду логарифмів сингулярних значень із акцентом на тангенс нахилу як диференціальну ознаку. На відміну від традиційних методів, що спираються на частотний аналіз або ключові точки, запропонований підхід забезпечує автоматизовану класифікацію без потреби в ручному налаштуванні параметрів. Це дозволяє ефективно розпізнавати маніпуляції, включаючи copy-move forgery та deepfakes, що є актуальним у контексті стрімкого розвитку ШІ-технологій.*

Висновки *роботи підтверджують високу ефективність методу для розрізнення реальних і ШІ-згенерованих зображень, де середнє значення slope для автентичних зображень становить -1,4026, а для ШІ-зображень відповідно -0,5829. Метод демонструє точність 87,76% на тестовому наборі з 98 зображень, а також Recall 93,55% і Specificity 85,07%, хоча виявлено обмеження при аналізі зображень із однорідною текстурою та наявність 7 хибнопозитивів. Результати підкреслюють практичне значення підходу для захисту від дезінформації, підтримки юриспруденції та етичного контролю ШІ, з перспективою подальшого вдосконалення через комбінацію з такими техніками, як SIFT (Scale-Invariant Feature Transform, Трансформація ознак, інваріантна до масштабу) чи CNN (Convolutional Neural Network, Згортальна нейронна мережа).*

Ключові слова: *сингулярне розкладання (SVD), ідентифікація зображень, лінійна регресія, ШІ-зображення, аналіз нахилу, комп'ютерний зір, виявлення фальсифікації зображень.*

Olena NYEMKOVA, Artem AKHEKYAN, Myroslava SKOLOZDRA. MATHEMATICAL METHOD FOR IDENTIFYING AI-GENERATED IMAGES BASED ON SVD AND LINEAR REGRESSION

Abstract. *The rapid advancement of artificial intelligence technologies, particularly generative models such as Stable Diffusion, has led to an increase in AI-generated images, creating significant challenges for countering disinformation and ensuring the integrity of digital content in social media, journalism, and legal contexts. The proposed mathematical method addresses this issue by providing an automated and effective approach to identifying synthetic patterns in images, offering practical value for ethical AI oversight and forensic applications. This research is particularly timely given the growing need for robust tools to detect image manipulations, such as deepfakes and copy-move forgeries, in an era of rapidly evolving AI capabilities.*

The purpose *of this study is to develop and test a mathematical method for detecting the authenticity of digital images, utilizing singular value decomposition (SVD) and linear regression, with the tangent of the slope (slope) as the key criterion for distinguishing real images from those generated by artificial intelligence (AI). The proposed approach aims to identify differences in the energy distribution of images, enabling the detection of synthetic patterns characteristic of AI-generated content, and to evaluate the method's effectiveness through practical examples.*

The methodology *involves transforming a digital image into a pixel matrix, applying singular value decomposition to obtain singular values, performing their logarithmic approximation, and constructing linear regression. The tangent of the slope is calculated as the regression coefficient, reflecting the rate of energy decay. To enhance accuracy, a block-based method is employed, dividing the image into 16x16 pixel submatrices, with the resulting slope values compared against an empirical threshold (e.g., <-0,8 for authentic images). Experiments were conducted on a dataset comprising real photographs and images generated by models such as Stable Diffusion, followed by statistical evaluation of the results.*

The scientific novelty *lies in the integration of SVD with linear regression to model the decay of logarithms of singular values, emphasizing the slope as a differential feature. Unlike traditional methods relying on frequency analysis or keypoints, this approach enables automated classification without the need for manual parameter tuning. It effectively detects manipulations, including copy-move forgery and deepfakes, addressing the rapid advancement of AI technologies.*

Conclusions. *The findings of the study confirm the high effectiveness of the method for distinguishing between real and AI-generated images, where the average slope value for authentic images is -1,4026, and for synthetic images, it is -0.5829. The method demonstrates an accuracy of 87,76% on a test set of 98 images, along with a Recall of 93,55% and Specificity of 85,07%, though limitations were identified in analyzing images with uniform textures and the presence of 7 false positives. The results underscore the practical significance of the approach for protecting against disinformation, supporting jurisprudence, and ethical AI control, with prospects for further improvement through integration with techniques such as SIFT (Scale-Invariant Feature Transform) or CNN (Convolutional Neural Network).*

Key words: *Singular Value Decomposition (SVD), Identification of Images, Linear Regression, AI-Generated Images, Slope Analysis, Computer Vision, Deepfake Detection.*

Постановка проблеми. Маніпуляція цифровими зображеннями, що впливає на їхню автентичність, є об'єктом аналізу, який може бути виявлений за допомогою математичних методів, зокрема сингулярного розкладання та лінійної регресії з аналізом тангенса нахилу. Такі маніпуляції, зокрема фальсифікація зображень, що передбачає процес зміни їхнього змісту з метою введення в оману, мають різне сприйняття залежно від контексту застосування. Термін «фальсифікація» несе дещо негативний відтінок, хоча його оцінка значною мірою залежить від сфери використання. У галузі розваг ця техніка застосовується позитивно, сприяючи створенню креативного та захопливого контенту. Натомість у контексті юриспруденції чи журналістики вона набуває негативного значення, часто асоціюючись із обманом чи поширенням дезінформації. У зв'язку з цим розробка та застосування методів виявлення таких маніпуляцій, включаючи фальсифікацію, є надзвичайно важливими в цих областях.

Існують різні методи маніпуляції зображень, які класифікуються залежно від технік і цілей. Відомі методи та їхнє застосування представлено у (табл. 1).

Методи маніпуляції варіюються від простого редагування (ретушування) до складних ШІ-генерацій. Їхнє застосування залежить від цілей – від нешкідливої творчості до серйозних злочинів. Оскільки різні методи маніпуляції використовують різні техніки, то їх виявлення вимагає специфічних підходів. Розрізняють таки основні підходи: блочні методи (розділення зображення на блоки, наприклад, 16x16 і порівняння їхніх ознак), ключово-точкові методи (використання SIFT, SURF для виявлення дублікатів чи аномалій), та SVD-аналіз (оцінка розпаду сингулярних значень для виявлення CMF чи ШІ-генерації).

Мета даної роботи полягає у розробці та апробації методу виявлення автентичності цифрових зображень на основі аналізу розпаду сингулярних значень із застосуванням сингулярного розкладання (SVD) та лінійної регресії. Основною ознакою для класифікації зображень як реальних чи створених штучним інтелектом є тангенс кута нахилу, отриманий шляхом апроксимації логарифмічної залежності сингулярних значень $\log(\sigma_i)$. Запропонований підхід спрямований на визначення характерних відмінностей у розподілі енергії зображень, що дозволяє диференціювати природні візуальні структури від синтетичних патернів, характерних для зображень, сформованих ШІ, та оцінити ефективність методу на практичних прикладах.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Таблиця 1

Класифікація методів маніпуляції зображень

№ з/п	Назва	Опис методу	Застосування	Небезпека
1	Ретушування (Image Retouching)	Метод передбачає коригування або покращення певних частин зображення, таких як видалення дефектів, зміни кольору, корекція освітлення чи текстури. Використовуються інструменти типу Photoshop для точкового редагування.	У рекламі чи соціальних мережах для «покращення» зовнішності моделей (згладжування шкіри, зміна форм). У реставрації старих фото для видалення подряпин чи плям.	Оскільки зміни не завжди приховують критичну інформацію, цей метод вважається відносно нешкідливим.
2	Сплайсинг (Image Splicing)	Об'єднання двох або більше зображень у одне, щоб приховати чи додати елементи. Зазвичай використовуються шари та маски для безшовного злиття.	У маніпуляції новинами шляхом створення фальшивих сцен, наприклад, додавання людини до події, де її не було. У художніх проєктах для створення сюрреалістичних зображень.	Приховування інформації у судових або кримінальних справах для фальсифікації доказів.
3	Копіювання та переміщення (Copy-Move Forgery, CMF)	Найпоширеніший метод, який включає копіювання частини зображення та вставлення її в іншу область того ж зображення. Часто супроводжується редагуванням (масштабування, обертання, додавання шуму) для маскування.	У фальсифікації доказів шляхом Додавання об'єктів (наприклад, зброї) до фото злочинних сцен. У соціальних мережах для створення ілюзій багатства чи присутності, наприклад, дублювання предметів.	Складність виявлення: завдяки редагуванню та шумам цей метод важко розпізнати без спеціальних алгоритмів, таких як SVD або SIFT.
4	Генерація зображень за допомогою ШІ (AI-Generated Forgery)	Використання генеративних моделей, наприклад, GANs - Generative Adversarial Networks, для створення реалістичних зображень із нуля або модифікації існуючих. Приклади: DALL·E, Stable Diffusion.	З метою дезінформації створюються фальшиві фото осіб чи подій (deepfakes). Для розваг та у мистецтві застосовується генерація унікальних зображень для творчих проєктів.	Складність виявлення: ШІ-зображення мають синтетичні патерни, які важко відрізнити від реальних без аналізу розпаду сингулярних значень чи інших методів.
5	Підміна облич (Face Swapping)	Заміна обличчя однієї людини на обличчя іншої з використанням технологій розпізнавання та синтезу, часто на основі deep learning.	Для розваг: популярно в додатках типу FaceApp або Zao. Зі злочинними намірами створюються фальшиві відео чи фото для шахрайства чи шантажу.	Для виявлення вимагає аналізу мікровиразів або аномалій у текстурах.
6	Додавання шуму чи артефактів (Noise Addition/ Artifact Insertion)	Додавання штучного шуму (наприклад, гаусового) або компресійних артефактів для маскування маніпуляцій.	У маскуванні редагувань, наприклад, у CMF для приховання швів. Для створення імітації старіння, наприклад, створення ефекту старого фото.	Для виявлення вимагає аналізу спектрального розподілу або SVD (виявлення невідповідностей).

Практичне значення даної роботи полягає в розробці та впровадженні методу виявлення автентичності цифрових зображень, який базується на аналізі сингулярного розкладання (SVD) та лінійної регресії з використанням тангенса кута нахилу (slope) як ключового критерію. Це має низку важливих прикладних аспектів, особливо в сучасному цифровому середовищі, де фальсифікація зображень стала поширеним явищем.

Аналіз останніх досліджень і публікацій. Метод сингулярного розкладання знайшов застосування серед кількох українських дослідників, особливо в галузях обробки зображень, аналізу даних і машинного навчання. Серед відомих українських дослідників SVD застосовували Алла Кобозєва, Олександр Потьомкін, Ігор Сердюк, Наталія Бондаренко та Володимир Лук'янчук, переважно в обробці

зображень, стеганографії та аналізі даних. У дослідженнях Кобозєвої SVD застосовувався для розкладання матриць зображень на компоненти, що дозволяло аналізувати локальні особливості, наприклад, максимальні сингулярні значення, і виявляти аномалії, пов'язані з маніпуляціями чи стеганографією [7].

Методи виявлення маніпуляції зображень за допомогою сингулярного розкладання (SVD), або близькі до нього підходи, активно досліджуються науковцями, інтереси яких лежать у галузі обробки зображень, комп'ютерного зору та цифрової криміналістики. SVD є узагальненням спектрального розкладання (eigenvalue decomposition) для несиметричних матриць і використовується у задачах зменшення розмірності, стиснення даних, видалення шуму та аналізу зображень. У контексті виявлення фальсифікації зображень SVD застосовується для аналізу розпаду сингулярних значень: натуральні зображення мають швидкий експоненціальний розпад log(σi) приблизно лінійний з від'ємним нахилом, тоді як фальсифіковані – повільніший або нелінійний через порушення природної структури пікселів [5]. Це робить SVD ефективним для виявлення copy-move forgery або ШІ-генерації, оскільки маніпуляції змінюють статистичні властивості матриці зображення [12]. Перевагою SVD над спектральним аналізом є можливість аналізувати несиметричні структури, як зображення, без перетворення у симетричну форму.

ШІ-генеровані зображення, наприклад, за допомогою GANs або diffusion models, можуть частково протидіяти SVD-детекції, але не повністю, через фундаментальні статистичні відмінності. Моделі ШІ можуть бути навчені імітувати статистичні властивості реальних зображень, включаючи розпад сингулярних значень. Наприклад, є відомості, що генератори можуть оптимізуватися для створення зображень із подібним SVD-розпадом, щоб обійти детектори [1]. Дослідження показують, що постобробка, наприклад, додавання шуму, дозволяє уникнути виявлення, оскільки SVD чутливий до таких маніпуляцій [15]. У 2025 році гібридні моделі, таки як Stable Diffusion, можуть генерувати зображення з реалістичним розпадом енергії, що робить SVD менш ефективним для нових генераторів [14]. Тим не менш, ШІ не може повністю протидіяти статистичним відмінностям; ШІ-зображення часто мають неприродний розподіл шуму або текстур, що порушує експоненціальний розпад сингулярних значень. Детектори на основі SVD, наприклад, блочні методи, виявляють локальні аномалії, які важко ідеально імітувати [8]. Дослідження показують, що навіть удосконалені GANs не можуть повністю відтворити SVD-розпад реальних зображень через обмеження тренувальних даних [3]. SVD комбінується з ML-моделями (наприклад, CNN), що робить імітацію складнішою. ШІ, щоб протидіяти, повинен тренуватися проти конкретних детекторів, але загальні SVD-методи стійкі [11]. Отже, повна імітація вимагає обчислювальних ресурсів і може призвести до артефактів в інших доменах, наприклад, частотному, що виявляються іншими методами. У змаганнях з SVD-детекцією, ШІ може перемагати в обмежених випадках, але не завжди, оскільки базові математичні властивості важко ідеально відтворити. Для повного виявлення імітацій потрібно комбінувати SVD з іншими техніками [2].

Отже, метод SVD є ефективним у виявленні діпфейків. Наведемо декілька прикладів досліджень на цю тему за останні роки. У статті [6] описано метод виявлення копіювання-вставки (copy-move forgery) на основі SVD. Зображення розбивається на блоки, для кожного з яких обчислюється SVD, а максимальне значення діагональної матриці (норма) використовується для групування схожих блоків. Ключовим для виявлення аномалій є аналіз сингулярних значень та їх розподілу, але лінійна регресія прямо не згадується. Стаття [10] пропонує метод виявлення фальсифікації типу «зшивання» (image splicing) з використанням SVD у комбінації з дискретним косинусним перетворенням (DCT). Зображення розбивається на блоки, для кожного обчислюються DCT-коефіцієнти, а потім застосовується SVD для вилучення особливостей. Автори статті [9] використовують метод SVD для вилучення особливостей з RGB-зображень (зокрема, з червоної матриці), після чого сингулярні значення та вектори передаються в одновимірний клітинний автомат для створення ключа автентифікації. У статті [4] описується метод виявлення маніпуляцій із зображеннями на основі SVD, де порушення лінійних залежностей у рядках або стовпцях зображення використовується для ідентифікації фальсифікацій. Метод фокусується на аналізі сингулярних значень що може бути сумісним з регресійним підходом для оцінки розпаду. Зауважимо, що у згаданих статтях, в яких статистичні методи застосовуються для оцінки аномалій, не описано чіткої комбінації SVD і лінійної регресії для моделювання розпаду логарифмів сингулярних значень $\log(\sigma_i)$.

Постановка завдання. Оскільки метою даної роботи є розробка математичного методу для визначення автентичності цифрових зображень шляхом розрізнення реальних зображень та тих, що створені штучним інтелектом (ШІ), з використанням сингулярного розкладання (SVD) та лінійної регресії, було сформульовано наступне **завдання**, яке передбачає наступне:

1. Аналіз структури цифрових зображень шляхом перетворення їх у матриці пікселів і обчислення сингулярних значень за допомогою SVD.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

2. Побудова логарифмічної апроксимації сингулярних значень та застосування лінійної регресії для визначення тангенса кута нахилу (slope) як кількісного показника розпаду енергії зображення.

3. Встановлення емпіричних порогів для класифікації зображень на основі значення slope, де автентичні зображення характеризуються різким експоненціальним розпадом (slope < -0,8), а ШІ-згенеровані – повільнішим (slope > -0,8).

4. Проведення експериментальної перевірки методу на тестовому наборі даних, що включає реальні фотографії та зображення, сформовані ШІ-моделями (наприклад, Stable Diffusion), з оцінкою точності та ідентифікації обмежень.

5. Оцінка практичної придатності розробленого підходу для захисту від дезінформації, підтримки юриспруденції та етичного контролю ШІ-технологій.

Постановка завдання зумовлена необхідністю протидії стрімкому поширенню фальсифікованих зображень у цифровому середовищі, що вимагає ефективних і автоматизованих методів аналізу, а також актуальністю інтеграції математичних інструментів у задачі комп'ютерного зору.

Виклад основного матеріалу дослідження. Метод сингулярного розкладання (Singular Value Decomposition, SVD) – це фундаментальна техніка лінійної алгебри, яка застосовується для розкладання матриці на три компоненти. Формально, будь-яку матрицю A розміром $m \times n$ можна представити у вигляді SVD розкладання:

$$A = U\Sigma V^T$$

де U – ортогональна матриця розміром $m \times m$, стовпці якої є лівими сингулярними векторами,

Σ – діагональна матриця розміром $m \times n$ з невід'ємними сингулярними значеннями $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k \geq 0$ (де $k = \min(m,n)$), які відображають «енергію» або важливість компонент матриці,

V^T – транспонована ортогональна матриця розміром $n \times n$, стовпці якої є правими сингулярними векторами.

Після обчислення сингулярних значень матриці зображення необхідно виконати розрахунок їхніх десяткових логарифмів і побудувати лінійну регресію залежно від порядкового номера сингулярного значення. На наступному етапі визначається тангенс кута нахилу отриманої прямої, який слугує ключовим параметром для класифікації зображень на реальні та фальсифіковані.

Був розроблений наступний алгоритм (який було реалізовано мовою програмування Python), основні його кроки наведено нижче:

1. Підготовка вхідних даних (завантажити цифрове зображення у форматі JPG і перетворити його на двовимірну матрицю пікселів A розміром $m \times n$, m – висота, n – ширина зображення; розбити матрицю A на блоки 16×16 для локального аналізу, отримавши множину підматриць A_i, i = 1, 2, … k, де k – кількість блоків).

2. Обчислення сингулярного розкладання (SVD) (для кожної підматриці A_i виконати сингулярне розкладання; витягти вектори сингулярних значень σ_i для подальшого аналізу).

3. Обчислення десяткових логарифмів (для кожного вектора σ_i обчислити десяткові логарифми сингулярних значень; сформувати залежність $y_j = \log_{10}(\sigma_{ij})$ від індексу j).

4. Побудова лінійної регресії (використовуючи метод найменших квадратів, апроксимувати отримані дані (j, y_j) лінійною функцією $y_j = aj + b$, де a – тангенс кута нахилу (slope); обчислити параметр a).

5. Визначення тангенсів нахилу блоків (взяти отримане значення a як тангенс кута нахилу для блоку A_i; повторити кроки 2 – 4 для всіх блоків з утворенням множини значень a_i).

6. Класифікація зображень (обчислити середнє значення тангенсу нахилу для всього зображення slope як середнє аріфметичне; порівняти його з емпіричним порогом, наприклад, якщо slope менше за -0,8, зображення класифікується як реальне, якщо slope не менше за -0,8, то фальсифіковане (ШІ-генероване)).

Для тестування алгоритму було використано 97 зображень, з яких 31 зображення (клас 1) отримане за допомогою ШІ-генератора [13] і 67 зображень (клас 0) взято з колекції реальних фотографій, отриманих з мобільного телефону. Всі зображення мали обсяг 100÷200 kB і розширення JPG. Роздільна здатність камери мобільного телефону 45 Мрх, час генерування одного ШІ-зображення 10,8 сек. Тематично реальні та генеровані зображення були різноманітними: одна людина, група людей, архітектура, природа, тварини.

Було отримано наступні результати. Для ШІ-генерованих зображень отримана наступна множина тангенсів кутів нахилу (slope), відсортована від мінімального до максимального: {-1,2855; -1,1473; -0,7889; -0,5886; -0,5701; -0,5679; -0,5674; -0,5647; -0,5641; -0,5519; -0,5492; -0,5455; -0,5402; -0,5343; -0,5278; -0,5273; -0,5267; -0,5256; -0,5247; -0,5175; -0,5144; -0,5134; -0,5102; -0,5081; -0,5041; -0,5000; -0,4934; -0,4933}; множина slope для реальних зображень відповідно така: {-7,1863; -5,5992; -4,4227;

-2,2959; -2,2162;, -1,7976; -1,7859;, -1,7567; -1,7381; -1,6716; -1,6427; -1,6412; -1,5873; -1,5612; -1,5589; -1,5526; -1,5303; -1,5043; -1,4736; -1,4441; -1,4348; -1,4233; -1,4229; -1,3669; -1,3588; -1,3501; -1,3109; -1,2676; -1,2661; -1,2149; -1,2094; -1,1981; -1,1911; -1,1552; -1,1455; -1,1422; -1,1389; -1,1346; -1,1266; -1,1223; -1,0716; -1,0706; -1,0637; -1,0594; -1,0361; -1,0119; -1,0051; -0,9976; -0,9797; -0,9629; -0,9588; -0,9233; -0,9229; -0,8921; -0,8869; -0,8359; -0,8147; -0,7753; -0,7384; -0,6939; -0,6525; -0,6518; -0,632; -0,6305; -0,6147; -0,6075; -0,5357}.

Статистичні показники множин для ШІ-генерованих та реальних зображень наведено у (табл. 2).

Таблиця 2

Описова статистика для набору даних slope

Показник	Реальні зображення	ШІ-зображення
Середнє значення	-1,4026	-0,5829
Медіана	-1,1552	-0,5278
Дисперсія вибірки	1,0933*	0,0318
Стандартне відхилення	1,0456**	0,1782
Інтервал	6,6506	0,7962
Мінімум	-7,1863	-1,2855
Максимум	-0,5357	-0,4893

*Примітка: *,** – такі великі значення дисперсії та стандартного відхилення пояснюються наявністю викидів {-7,1863; -5,5992; -4,4227}, що не є характерними для набору даних slope реальних зображень*

Виконано бінарну класифікацію отриманих результатів. Для порогу slope = -0,8 з 31 ШІ-зображень 2 помилково класифікуються як реальні (False Negatives, FN = 2), решта 29 класифікуються як фейкові (True Positives, TP = 29). З 67 справжніх зображень 10 класифікуються як фейкові (False Positives, FP = 10) і 57 як справжні (True Negatives, TN = 57).

Для оцінки якості класифікації застосуємо наступні параметри: точність (*Accuracy*), точність прогнозу позитивного класу (*Precision*), чутливість (*Recall*), специфічність (*Specificity*), площу під ROC-кривою.

Точність (*Accuracy*) – це метрика, яка вимірює частку правильних прогнозів моделі відносно загальної кількості прогнозів. Точність обчислюють, щоб оцінити, наскільки добре запропонований метод розрізняє реальні зображення та зображення, створені штучним інтелектом:

$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} * 100\% = 87{,}6\%$$

Точність 87.76% висока, але менша 95%, що може вказувати на жорсткість порогу -0,8. Оптимізація порогу може підвищити точність.

Точність прогнозу позитивного класу (*Precision*) вимірює частку правильних позитивних прогнозів серед усіх позитивних прогнозів, що означатиме частку правильно класифікованих зображень, створених штучним інтелектом, серед усіх зображень, позначених як ШІ:

$$Precision = \frac{TP}{TP + FP} * 100\% = 74{,}36\%$$

Precision показує, що серед усіх зображень, класифікованих як ШІ, 74,36% дійсно є ШІ. Решта 25,64% (FP = 10) – це реальні зображення, помилково віднесені до ШІ. Порівняно з *Accuracy* (87,76%) *Precision* нижча, оскільки враховує лише позитивний клас (ШІ) і чутлива до FP. Це вказує на те, що поріг -0,8 допускає певну кількість помилок, особливо серед реальних зображень із slope близьким до -0,8.

Чутливість (*Recall*, або *True Positive Rate*) вимірює частку правильно виявлених позитивних випадків серед усіх реальних позитивних випадків. *Recall* допомагає оцінити, наскільки добре метод виявляє ШІ-зображення, використовуючи поріг -0,8:

$$Recall = \frac{TP}{TP + FN} * 100\% = 93{,}55\%$$

Ця метрика показала, що серед усіх ШІ-зображень метод правильно виявляє 93,55% як фальсифіковані. Решта 6.45% (FN = 2) – це ШІ-зображення, помилково класифіковані як реальні, що може бути пов'язано з їхньою високою реалістичністю. Загалом, *Recall* вищий за *Precision* (74,36%), що вказує на те, що запропонований метод для обраного порогу краще виявляє ШІ-зображення, але з ризиком

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

помилково класифікувати реальні як ШІ. Високий *Recall* робить метод корисним для сфер, де пропуск ШІ-зображення критичний (наприклад, дезінформація чи юриспруденція), але низький *Precision* вимагає балансу, наприклад, оптимізації порогу до -0,9 для зменшення FP. Загалом, метод краще «ловить» ШІ (93,55%), ніж виключає помилки в реальних (TN/(TN+FP)*100% = 85,07%).

Специфічність (*Specificity*, або *True Negative Rate*) – це метрика, яка вимірює частку правильно виявлених негативних випадків (реальних зображень) серед усіх реальних негативних випадків.

$$Specificity = \frac{TN}{TN + FP} * 100\% = 85{,}07\%$$

Метрика показала, що серед усіх реальних зображень метод правильно класифікує 85,07% як реальні. Решта 14,93% – це реальні зображення, помилково класифіковані як ШІ, що може бути пов'язано з шумом або текстурами, де slope ≥ -0,8. Отже, є ризик помилково «позначати» реальні зображення як ШІ. *Specificity* = 85,07% свідчить про добру здатність методу виключати фальсифікації для реальних зображень, що є сильною стороною для захисту від дезінформації. Різниця з ШІ (де *Recall* = 93,55%) вказує, що оптимізація порогу (наприклад, до -0,9) може підвищити *Specificity*.

Для обчислення ROC-AUC (Receiver Operating Characteristic – Area Under Curve) на основі наданих даних нахилу slope для реальних зображень і ШІ-зображень було використано Python із бібліотеками numpy, sklearn та matplotlib для візуалізації. ROC-AUC вимірює якість бінарної класифікації, порівнюючи справжні позитивні та хибні позитивні ставки при різних порогах (рис.1).

Значення AUC (Area Under Curve) = 0,95 свідчить про високу ефективність запропонованого методу, заснованого на аналізі тангенса кута нахилу за допомогою сингулярного розкладання та лінійної регресії, для розрізнення реальних зображень і тих, що створені штучним інтелектом. AUC = 0,95 наближається до ідеального значення 1,0, що вказує на хорошу здатність методу класифікувати зображення з мінімальною кількістю помилок.

Висновки та перспективи подальших розвідок. Проведене дослідження підтвердило ефективність розробленого математичного методу для виявлення автентичності цифрових зображень на основі сингулярного розкладання (SVD) та лінійної регресії з аналізом тангенса кута нахилу (slope). Аналіз розподілу енергії зображень через логарифмічну апроксимацію сингулярних значень із порогом -0,8 дозволив чітко розмежувати реальні зображення (середнє slope = -1,4026; 85,1% значень < -0,8) та зображення, створені штучним інтелектом (середнє slope = -0,5829; 93,5% значень > -0,8). Експериментальна оцінка на тестовому наборі з 67 реальних фото з мобільного телефону та 31 зображення, згенерованого моделями типу Stable Diffusion, показала високу точність класифікації (Accuracy = 87,76%) та AUC = 0,95, що свідчить про добру здатність методу розпізнавати синтетичні патерни, включаючи deepfakes та copy-move forgery. Експериментальні результати: 57 справжніх

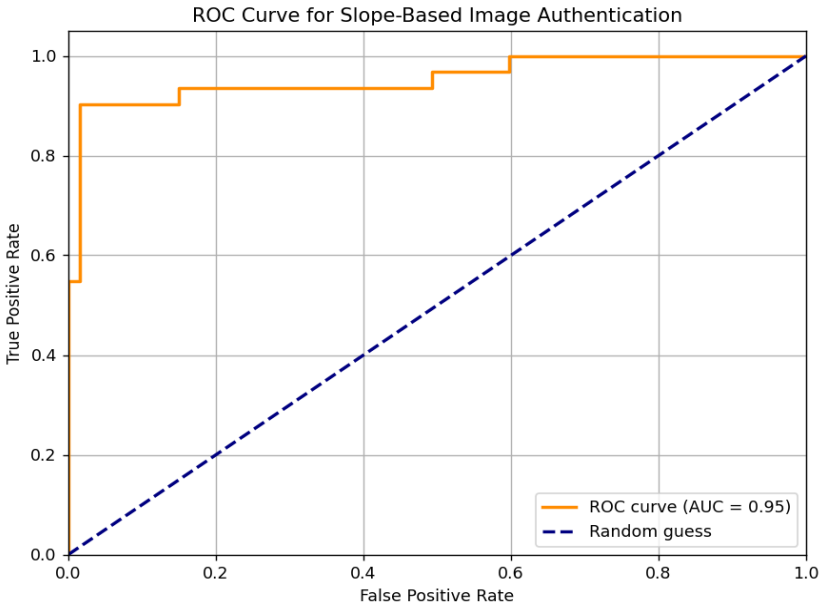


Рис. 1. Графік ROC-кривої (суцільна лінія) показує ефективність класифікації. Додано лінію випадкового вибору (пунктир, FPR = TPR) для порівняння

негативних, 29 справжніх позитивних, 10 хибнопозитивних і 2 хибнонегативних підкреслюють збалансованість класифікації з Precision = 74,36%, Recall = 93,55% та Specificity = 85,07%.

Практична придатність методу підтверджена його потенціалом для захисту від дезінформації в соціальних мережах, підтримки юриспруденції через верифікацію фото як доказів та етичного контролю ШІ-технологій. Високе значення Recall (93,55%) забезпечує ефективне виявлення ШІ-зображень, що критично для боротьби з фальсифікаціями, тоді як Specificity (85,07%) гарантує надійність виключення реальних зображень. Отримані результати підкреслюють наукову новизну підходу, що полягає в застосуванні тангенса нахилу як диференціальної ознаки.

Обмеження методу пов'язані з обробкою зображень із однорідною текстурою, де slope наближається до порогу -0,8, а також із викидами, наприклад, slope < -4,0 у реальних зображеннях, що потребує фільтрації. Перспективи подальших досліджень включають оптимізацію порогу через ROC-аналіз для підвищення Precision до 85% і вище, розширення тестового набору даних із різноманітними ШІ-моделями (DALL·E, Midjourney) та інтеграцію з іншими техніками для підвищення стійкості до аномалій. Для покращення результатів планується впровадження адаптивної фільтрації малих сингулярних значень, комбінацію з глибоким навчанням для автоматичного визначення порогу та застосування методу до відеоаналізу для виявлення deepfakes у динамічних сценах. Отримані результати підкреслюють практичне значення роботи для сучасних викликів у комп'ютерному зорі й безпеці даних, з потенціалом для автоматизованого впровадження в реальному часі.

Список використаних джерел:

1. Ba Z., Zhang Y., Cheng P., Gong B., Zhang X., Wang Q., Ren K. Robust Watermarks Leak: Channel-Aware Feature Extraction Enables Adversarial Watermark Manipulation. arXiv:2502.06418v1 [cs.CV], 10 Feb 2025. URL: https://arxiv.org/html/2502.06418v1

2. Capasso P., Cattaneo G., de Marsico M. A Comprehensive Survey on Methods for Image Integrity. ACM Transactions on Multimedia Computing, Communications and Applications, Vol. 20, No. 11, Article No. 347, 2024, pp. 1–34. URL: https://doi.org/10.1145/3633203

3. Deb P., Deb S., Das A., Kar N. Image Forgery Detection Techniques: Latest Trends and Key Challenges. IEEE Access, Vol. PP, No. 99, January 2024, pp. 1–1. DOI: 10.1109/ACCESS.2024.3498340

4. Gul G., Avcibas I., Kurugollu F. SVD Based Image Manipulation Detection. In: 2010 IEEE International Conference on Image Processing, Hong Kong, China, September 2010. DOI: 10.1109/ICIP.2010.5652854. URL: https://ieeexplore.ieee.org/document/5652854

5. Kashyap A., Agarwal M., Gupta H. Detection of Copy-Move Image Forgery Using SVD and Cuckoo Search Algorithm. arXiv:1704.00631v1 [cs.MM], 3 Apr 2017. URL: https://arxiv.org/pdf/1704.00631. DOI: 10.14419/ijet.v7i2.13.11604

6. Khudhair Z. N., Mohamed F., Rehman A., Saba T., Bahaj S. A. Detection of Copy-Move Forgery in Digital Images Using Singular Value Decomposition. Computers, Materials & Continua, Vol. 74, No. 2, 2023, pp. 4135–4147. URL: https://doi.org/10.32604/cmc.2023.032315

7. Kobozieva A., Bobok I., Kushnirenko N. Steganalysis Method for Detecting LSB Embedding in Digital Video, Digital Image Sequence. In: 11th International Conference «Information Control Systems and Technologies» (ICST 2023), Odesa, 21–23 September 2023, pp. 78–90. [CEUR Workshop Proceedings, Vol. 3513]. URL: https://ceur-ws.org/Vol-3513/paper07.pdf

8. Lađević A. L., Kramberger T., Kramberger R., Vlahek D. Detection of AI-Generated Synthetic Images with a Lightweight CNN. Artificial Intelligence, Vol. 5, No. 3, 2024, pp. 1575–1593. URL: https://doi.org/10.3390/ai5030076

9. Malakooti M. V., Tafti A. P., Rohani F., Moghaddasifar M. A. RGB Digital Image Forgery Detection Using Singular Value Decomposition and One Dimensional Cellular Automata. In: 2012 8th International Conference on Computing Technology and Information Management (NCM and ICNIT), 2012. URL: https://ieeexplore.ieee.org/document/6268546

10. Moghaddasi Z., Jalab H. A., Noor R. M. Image Splicing Forgery Detection Based on Low-Dimensional Singular Value Decomposition of Discrete Cosine Transform Coefficients. Neural Computing and Applications, 2018. URL: https://doi.org/10.1007/s00521-018-3648-3

11. Saberi M., Sadasivan V. S., Rezaei K., Kumar A., Chegini A., Wang W., Feizi S. Robustness of AI-Image Detectors: Fundamental Limits and Practical Attacks. arXiv:2310.00076, Feb 2024. URL: https://doi.org/10.48550/arXiv.2310.00076

12. Sengupta S., Shinde P., Shah H. Image Forgery Detection Techniques for Forensic Sciences. International Journal of Software & Hardware Research in Engineering, Vol. 2, No. 8, August 2014. URL: https://ijournals.in/wp-content/uploads/2017/07/9.2814-Prajakta.pdf

13. Stable Diffusion 2.1 Demo. URL: https://huggingface.co/spaces/stabilityai/stable-diffusion

14. Vahdati D. S., Nguyen T. D., Azizpour A., Stamm M. C. Beyond Deepfake Images: Detecting AI-Generated Videos. arXiv:2404.15955v1 [cs.CV], 24 Apr 2024. URL: https://arxiv.org/html/2404.15955v1

15. Xie H., Ni J., Zhang J., Zhang W., Huang J. Evading Generated-Image Detectors: A Deep Dithering Approach. Signal Processing, Vol. 197, August 2022, 108558. URL: https://doi.org/10.1016/j.sigpro.2022.108558

Дата надходження статті: 19.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

UDC 004.852:004.83
DOI https://doi.org/10.32689/maup.it.2025.3.15

Yaroslav PAVLENKO
Postgraduate Student, Kharkiv National University of Radio Electronics,
yaroslav.pavlenko@nure.ua
ORCID: *0009-0001-2275-858X*

Natalia VALENDA
Candidate of Technical Sciences, Associate Professor, Department of Software Engineering,
Kharkiv National University of Radio Electronics,
natalia.valenda@nure.ua
ORCID: *0000-0003-3250-6172*

METHODS OF FORECASTING AND DATA CLASSIFICATION BASED ON NEURAL NETWORKS

Abstract. *The article is devoted to a comprehensive review of neural network models in forecasting and classification tasks. Finding the strengths and weaknesses of different forecasting methods using neural networks. Exploring the possibilities of improving the use of neural networks in forecasting and classification tasks.*

The purpose of the work. *The purpose of this work is to study methods of forecasting and data classification based on neural networks. Which means a review of existing approaches and finding new ways to improve the solution of the above problems. Finding ways to improve existing models. The task of this study is to compare existing methods of using neural networks in forecasting problems and to obtain new approaches to improve existing methods.*

Methodology. *It is based on the analysis of scientific publications on neural network models, as well as prediction and classification methods. For this purpose, the characteristics and methods of comparative analysis of the strengths and weaknesses of neural networks are provided. As well as recommendations for improving prediction methods, where possible.*

Scientific novelty. *The solution to the problems set and the scientific novelty of this research lies in identifying ways to improve methods and in a criterion-based comparison of existing methods for using neural networks in data classification and prediction tasks, improving new approaches based on existing ones to improve the processing processes of the above-mentioned tasks.*

Conclusions. *Analysis of neural network models in forecasting tasks revealed their strengths and weaknesses. Criterion analysis established the advantages of forecasting methods using neural networks. Recommendations for improving forecasting methods are proposed.*

Key words: *methods of forecasting, classification methods, neural network models, machine learning.*

Ярослав ПАВЛЕНКО, Наталя ВАЛЕНДА. МЕТОДИ ПРОГНОЗУВАННЯ ТА КЛАСИФІКАЦІЇ ДАНИХ НА ОСНОВІ НЕЙРОННИХ МЕРЕЖ

Анотація. *Стаття присвячена комплексному огляду моделей нейронних мереж у задачах прогнозування та класифікації. Знаходження сильних та слабких сторін різних способів прогнозування із застосуванням нейронних мереж. Дослідження можливостей поліпшення використання нейронних мереж у задачах прогнозування та класифікації.*

Мета роботи. *Метою цієї роботи є дослідження методів прогнозування та класифікації даних на основі нейронних мереж. Що означає огляд вже наявних підходів та знаходження нових способів для вдосконалення вирішення вищевказаних задач. Знаходження способів поліпшення існуючих моделей. Завданням даного дослідження є порівняння існуючих методів використання нейронних мереж у задачах прогнозування і в здобутку нових підходів для покращення існуючих методів.*

Методологія. *Базується на аналізі наукових публікацій моделей нейронних мереж, а також методів прогнозування та класифікації. Для цього використано надання характеристики та методи порівняльного аналізу сильних та слабких сторін нейронних мереж. А також надання рекомендацій щодо поліпшення методів прогнозування, де це можливо.*

Наукова новизна. *Вирішення поставлених задач та наукова новизна даного дослідження полягає у виявленні способів поліпшення методів та у критеріальному порівнянні існуючих методів використання нейронних мереж, у задачах класифікації та прогнозування даних, вдосконалення нових підходів на основі вже існуючих, для покращення процесів обробки вищезазначених задач.*

Висновки. *Аналіз моделей нейронних мереж у задачах прогнозування виявив їх сильні та слабкі сторони. Критеріальний аналіз встановив переваги методів прогнозування із використанням нейронних мереж. Запропоновано рекомендації щодо вдосконалення методів прогнозування.*

Ключові слова: *методи прогнозування, методи класифікації, моделі нейронних мереж, машинне навчання.*

Problem statement. The relevance of this topic is largely justified by the interest of scientists in the field of information technology. Since there are different approaches to forecasting and classification problems at the moment, it is important to study them to identify the likelihood of their improvement.

Also, such studies are relevant due to the practical need to study methods and various models in the commercial sector. The available methods are diverse and rely on different approaches to the use of neural networks, but are still not perfect in solving certain problems. That is why there is a demand for the prospect of their improvement and optimization. Therefore, the analysis and criterion-based comparison of existing methods and tools is a popular task.

Modern progress in the field of information technology intensively develops existing approaches in various data analysis tasks. And sometimes modern scientific progress introduces new technologies intensively, even ahead of the development of the existing ones and the introduction of innovations. Therefore, there is a growing need for a comprehensive analysis of existing approaches in order to find their advantages in individual tasks and the possibility of their improvement.

Training neural networks begins with a variety of data that are collected and used to train models. Large amounts are required for deep learning [8]. The size of the data has a significant impact, since the more data there is, the more efficiently the program works.

Since different neural network models may be suitable for different tasks, it is important to evaluate their advantages and disadvantages specifically in prediction and classification tasks.

Another important task is to find new useful approaches to improve and use them in data forecasting and classification tasks.

Even some varieties of the same neural network may have a significant advantage over others, for example, in forecasting tasks, since they are better suited to detecting dependencies, both short-term and long-term. When analyzing the research problem, one should also pay attention to the analysis of forecasting methods in tasks with incomplete or heterogeneous data.

As the world rapidly grows in. terms of the amount of data that needs analysis and the need to make informed decisions based on available data (possibly incomplete and heterogeneous) in various industries, it is important to explore new approaches and analyze existing methods of forecasting and classifying data.

Neural networks, in some cases, demonstrate high efficiency in machine learning tasks, but sometimes their application in real-world conditions can often face a number of challenges [9].

For example, such challenges often include the need to adapt models to the specifics of the data, increasing their robustness to noise, and the ability to generalize their application across different types of data. The importance of this research is due to both the increasing complexity of modern forecasting tasks and the need for highly accurate and reliable algorithms for data classification in various industries, such as finance, medicine, energy, and others.

The need to improve existing prediction and classification methods is especially relevant in conditions where there is insufficient information in the data models to analyze the behavior of the system or to provide an assessment of it. Since many tasks involve performing a certain analysis on data of different formats and sources, this requires research and comparison of models for compliance with these features and their effectiveness in analyzing different data. Therefore, sometimes the integration of heterogeneous data sources is important for research.

It is important to critically compare the key capabilities, advantages and disadvantages of information technologies for data forecasting using neural networks, using metrics for assessing efficiency and accuracy.

Modern neural networks can often require significant computational resources and large data sets for training, which can limit their application in some practical tasks [10]. Therefore, research and benchmarking of prediction and data classification approaches that take these limitations into account are of utmost importance.

Analysis of recent research and publications. Recently, intensive development of research in the field of application of neural networks has been ongoing. New practices of application are being developed and already accepted methods are being modernized [5].

Due to the growth of computing power and the availability of large amounts of data, the rapid development of the application of neural networks in forecasting and classification tasks has taken place. For example, the development of recurrent architectures has proven its effectiveness in time series forecasting tasks. This has made it possible to significantly increase the accuracy of forecasting compared to classical methods such as ARIMA or regression analysis [11]. Various scientific publications demonstrate the integration of neural networks into various fields of use. For example, on stock exchanges, they can be used to predict rates and global trends. Neural network models used for forecasting can also be used in other fields, for example, in DSS [14]. In medicine, they can be used to classify images to detect the development of diseases. Rapid progress creates new challenges for research. Analysis of recent research and publications.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

The purpose of this article is to analyze and generalize the criteria-based comparison of the main neural network models in data forecasting and classification tasks, as well as to determine their effectiveness in performing individual tasks.

To implement this goal, the following tasks have been formulated: 1) to analyze existing neural network architectures; 2) to describe their features in forecasting and classification tasks; 3) to conduct a criteria-based comparison of methods and identify their advantages and disadvantages; 4) to consider examples of the use of neural networks for data forecasting and recovery of lost data; 5) to provide recommendations on existing neural network models; 6) to identify promising areas of research in the future.

To achieve this goal, it is necessary to generalize and systematize the existing material on neural network models, forecasting and classification methods, problems and challenges of the existing research processes, and also to conduct a comprehensive criteria-based comparison of the advantages and disadvantages of existing architectures.

Presentation of the main material. The tasks of forecasting and data classification are key components of the analysis and processing of digital data in the modern field of information technology.

Classification is aimed at assigning objects to a certain category according to their input parameters and certain patterns. Forecasting, in turn, means predicting future values of the system taking into account the existing patterns in past data. Classical forecasting tasks include such methods of time series analysis as ARIMA and SARIMA, as well as linear and polynomial regression [3]. They are mainly used to find the result for data with simple trends and seasonality, but can reveal weaknesses in the presence of a significant amount of noise or a large amount of lost data.

Key features of classical forecasting methods:

– ARIMA. A statistical method of modeling time series. Sometimes it can be weak for nonlinear processes, but its advantage is the clarity of the parameters.

– SARIMA. An extension method of ARIMA, but its use is more directed specifically to seasonal fluctuations. Therefore, this method is effective in problems where it is important to take seasonality into account seasonality.

– Linear regression. A mathematical model that describes the relationship between variables as a straight line on a graph.

– Polynomial regression. Improves linear regression because it takes into account curvilinear dependencies.

Classical classification problems include methods such as logistic regression, discriminant analysis, and the kNN method. Such classical approaches are effective for small amounts of data, but can be vague in the absence of defined parameters.

Key features of classical classification methods:

– Logistic regression. A binary classification model that works well for simple tasks, but has difficulties in scaling to multi-class data.

– LDA method. The method searches for linear boundaries between classes.

– kNN method. The method is effective in object classification tasks based on similarity to the closest examples.

Unlike classical approaches to data prediction and classification, neural networks can demonstrate much greater efficiency and flexibility [15]. Because neural networks can detect complex dependencies in large amounts of data and provide high accuracy of prediction and classification.

Neural networks are increasingly becoming a standard in many areas of prediction and classification. For example, in security and medical image processing tasks and in forecasting fluctuations and trends in the financial sector. The active development of artificial intelligence has ensured the introduction of improved neural network architectures, such as convolutional (CNN), recurrent (RNN, LSTM) and transformer models [2]. Also, a significant impetus for their development was the fact that modern data is characterized by a large volume and various nonlinear connections.

Neural networks consist of layers – combinations of neurons. Neural networks can have a single-layer or multilayer structure. Layers can be: output, hidden, input.

In a multilayer neural network (in addition to the main layers), there are also intermediate layers. In different architectures of neural networks, the number of intermediate layers can be different. The number depends on the complexity of the neural network.

During the calculation, neurons receive data from a layer and transmit it to the next neurons [7]. Such actions create connections that have their own weight and are called synapses.

CNN architecture is characterized by the use of convolutional layers to find local features. The advantages of the architecture include: high efficiency in working with graphic information. The architecture does not

require manual creation of features and can perfectly detect object patterns. CNNs are typically used for image processing, but CNNs can also be used for prediction tasks. For example, to predict the next value of a financial chart by treating the chart data as a sequence of local patterns.

RNNs use recurrent connections that help take into account previous states of the sequence. RNNs use connections that can form directed loops (Fig. 1). This approach allows them to store hidden state and information about the inputs to the sequence. Effective use in time series forecasting tasks is a feature of RNNs.

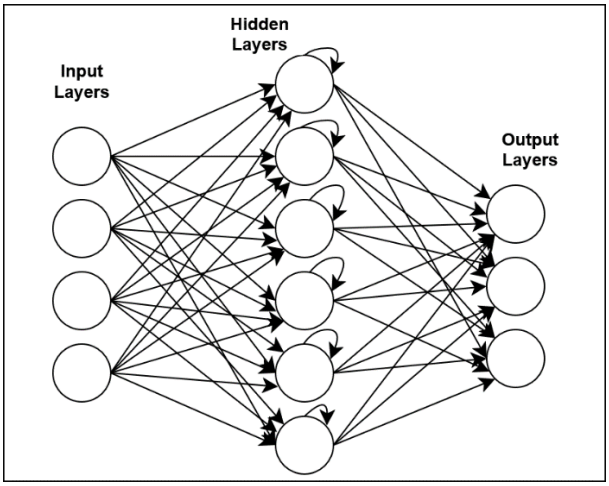


Fig. 1. RNN scheme

The architecture of RNN consists of the following components: input layer, recurrent connection, hidden layer. The main feature is the recurrent connection [4]. This approach allows you to store information at each time step.

The LSTM architecture is an improved RNN, but with memory caches to store long-term dependencies [6]. The feature of LSTM is that this model can store information for a long time. It is convenient to use in prediction and classification tasks. Transformers is a modern neural network architecture characterized by the use of the attention mechanism. This mechanism helps to process all elements of a sequence simultaneously. Unlike other neural network models, transformers analyze the entire context at once, rather than step by step. This approach allows for the effective detection of global dependencies. Transformers are commonly used in text analysis, but they can also be used in other areas. For example, they can be used to predict time series. Using the attention mechanism allows the model to analyze dependencies between elements regardless of their distance. In forecasting, the model can take into account not only the latest values, but also past ones.

During the study, it is important to apply criterion-based comparison of neural network models in data prediction and classification tasks. During the search and preparation of information from open sources, the advantages of the models were analyzed [13]. Also, it is advisable to use linear adaptive convolution with weighting coefficients. Based on the analyzed information from literary sources, we will evaluate the models [1]. We will evaluate the models according to the criteria (Tab. 1). The criteria are indicated by an ordinal scale from 1 to 10.

Table 1

Vector description of models according to selected criteria

Models	Criteria				
	Ability to model long-term dependencies	Speed	Model size, complexity	Accuracy in forecasting tasks	Accuracy in classification tasks
CNN	3	7	7	5	9
RNN	6	6	7	7	7
LSTM	9	7	6	9	8
Transformers	10	7	6	9	9

Since CNN has worse parameters, according to the Pareto principle, this model can be excluded [12]. Therefore, we will exclude this model from the calculations and continue the calculation of utility.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

To perform a qualitative comparative analysis, we will perform normalization and create a normalized description of alternatives. As a result, all values will be in the range from 0 to 1, where 1 corresponds to the best value, and 0 corresponds to the worst value.

The next step is to proceed to the normalization calculations and enter the data (Tab 2).

Table 2

Normalized vector description of alternatives

Models	Criteria				
	Ability to model long-term dependencies	Speed	Model size, complexity	Accuracy in forecasting tasks	Accuracy in classification tasks
RNN	0	0	1	0	0
LSTM	0	1	0	1	0.5
Transformers	1	1	0	1	1

Since certain criteria have different priorities in model evaluation, they are different in importance. We use linear adaptive convolution with weighting coefficients to calculate the utilities.

As a result, the utility (k) of the models in forecasting and classification tasks will be as follows: RNN (k = 0.16), LSTM (k = 0.52), Transformers (k = 0.65).

So, as a result of the criterion comparison with weight coefficients, we can conclude that transformers have greater utility. It was also found that the given neural network models can improve the results when adding additional layers.

Conclusions. As a result of the criterion comparison based on the selected indicators, the relative utility of the models in forecasting and classification tasks was determined. The results indicate that transformer architectures have greater utility and demonstrate the best balance between accuracy in forecasting and classification tasks. Transformers have some of the highest indicators. The results also showed that LSTM remains very effective in forecasting time series.

The study analyzed classical methods of forecasting and classification of data: ARIMA, SARIMA, regression models, logistic regression, kNN. Also, an analysis of modern approaches based on neural networks was carried out: CNN, RNN, LSTM, transformers. It was found that classical algorithms remain useful in conditions of limited data volumes and pronounced dependencies. Neural networks have high performance in most modern tasks. They have the ability to take into account long-term dependencies and work in forecasting and classification tasks. The comparison results indicate the prospects of using transformer architectures in both forecasting and classification tasks.

It has been found that the prediction results of neural network models can be improved by adding additional layers. Further research in this direction can focus on increasing the number of layers and making changes to the architecture of neural networks, developing hybrid approaches. Therefore, neural networks are an effective tool for modern data analytics.

Bibliography:

1. Aggarwal C. Neural Networks and Deep Learning: A Textbook. Springer. Cham. 2018. 498 p.
2. Bharadiya J. Machine Learning and AI in Business Intelligence: Trends and Opportunities. *International Journal of Computer (IJC)*. 2023. Vol. 48, No 1. URL: https://www.researchgate.net/publication/371902170 (дата звернення: 24.09.2025)
3. Box G., Jenkins G. Time Series Analysis: Forecasting and Control. 5th ed. Hoboken: New Jersey, 2015. 712 p.
4. Goodfellow I., Bengio Y., Courville A. Deep learning. Cambridge : MIT Press, 2016. 775 p
5. Haykin S. Neural Networks. A comprehensive Foundation. Prentice Hall, Inc. N.J. 2ed. 1999. P. 690.
6. Hochreiter S., Schmidhuber J. Long Short-Term Memory. *Neural Computation*. 1997. 9(8). P. 1735–1780.
7. Introduction to Deep Learning. URL: https://www.geeksforgeeks.org/introduction-deep-learning (дата звернення 24.09.2025)
8. LeCun Y., Bengio Y., Hinton G. Deep learning. *Nature*. 2015. Vol. 521, No 7553. P. 436–444.
9. Schmidhuber, J. Deep learning in neural networks: An overview. Neural *Networks*. 2015. Vol. 61. P. 85–117.
10. Zhang A., Lipton Z., Li M., Smola A. Dive into Deep Learning. Cambridge : Cambridge University Press, 2021. 789 p.
11. Zhang, G. P. Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing*. 2003. Vol. 50. P. 159–175.
12. Miller D. Pareto principle. Routledge Encyclopedia of Philosophy. London. URL: https://doi.org/10.4324/9780415249126-s097-1 (дата звернення 24.09.2025)

13. Samek W. et al. Explaining deep neural networks and beyond: a review of methods and applications. *Proceedings of the IEEE*. 2021. Vol.109, № 3. P. 247–278.

14. Артем ВАТУЛА Основні методи машинного навчання в СППР. Інформаційні технології та суспільство. 2025. DOI: https://doi.org/10.32689/maup.it.2025.1.6 (дата звернення: 24.09.2025).

15. Субботін С. О. Нейронні мережі: теорія та практика : навчальний посібник. Житомир : Вид. О. О. Євенок, 2020. 184 с.

Дата надходження статті: 25.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.415.5
DOI https://doi.org/10.32689/maup.it.2025.3.16

Борис ПАНАСЮК
аспірант спеціальності «Інженерія програмного забезпечення»,
Вінницький національний технічний університет,
boris.panasyuk@gmail.com
***ORCID:** 0009-0007-2064-9121*

Наталя БАБЮК
кандидат технічних наук, доцент кафедри програмного забезпечення,
Вінницький національний технічний університет,
babiuk@vntu.edu.ua
***ORCID:** 0000-0003-0607-6340*

КОНТРАКТНО-ОРІЄНТОВАНИЙ ЦИФРОВИЙ ДВІЙНИК МІКРОСЕРВІСНОЇ СИСТЕМИ: МОДЕЛЬ, МЕТАМОДЕЛЬ, АРТЕФАКТИ OPENAPI/ASYNCAPI

Анотація. *Метою дослідження є створення контрактно-орієнтованого цифрового двійника мікросервісної системи, що базується на моделі та метамоделі взаємодії сервісів через їх API-контракти. Для досягнення мети використано підхід API-first: формальні специфікації сервісів (OpenAPI для REST API та AsyncAPI для асинхронних API) слугують артефактами, на основі яких автоматично побудовано модель цифрового двійника.*

Методологія. *Застосовано аналіз та узагальнення сучасних підходів до цифрових двійників, моделювання мікросервісної архітектури із використанням формальних описів інтерфейсів, а також виконано порівняльний аналіз з існуючими моделями цифрових двійників.*

Наукова новизна. *Запропоновано концепцію «контрактно-орієнтованого» цифрового двійника, що вперше фокусує цифрову модель системи на її API-контрактах, забезпечуючи автоматизоване отримання та актуалізацію двійника з артефактів OpenAPI/AsyncAPI, тим самим поєднуючи процес документування API з підтримкою віртуальної копії системи.*

Висновки. *Контрактно-орієнтований підхід дозволяє підтримувати цифровий двійник актуальним при еволюції мікросервісів, спрощує тестування сумісності сервісів і аналіз поведінки системи без впливу на продуктивне середовище. Запропонований підхід апробовано на прикладі спрощеної мікросервісної системи; результати підтверджують можливість автоматичного формування двійника та ефективність його використання для інтеграційного тестування нових версій сервісів. Отримані результати можуть бути впроваджені у практику DevOps для автоматизації регресійного тестування мікросервісів та контролю відповідності їх реалізації заявленим контрактам. В цілому, використання контрактно-орієнтованого двійника сприяє підвищенню якості та надійності мікросервісних програмних комплексів та скорочує час, необхідний на інтеграційне тестування.*

Ключові слова**:** *контракт-орієнтоване моделювання, інформаційна система, цифровий двійник, мікросервісна архітектура, API-контракт, моделювання, автоматизація.*

Borys PANASIUK, Natalia BABIUK. CONTRACT-ORIENTED DIGITAL TWIN OF A MICROSERVICE SYSTEM: MODEL, METAMODEL, OPENAPI/ASYNCAPI ARTIFACTS

Abstract. The study aims *to develop a contract-oriented digital twin of a microservice-based system, grounded in a model and meta-model of service interactions defined by their API contracts.*

Methodology. *The study adopts an API-first approach: formal service interface specifications (OpenAPI for REST APIs and AsyncAPI for event-driven APIs) are used as artifacts to automatically construct the digital twin model. Additionally, a comparative analysis with existing approaches was conducted.*

Scientific novelty. *The concept of a «contract-oriented» digital twin is proposed, focusing the virtual model of the system on its API contracts; this approach is novel in enabling automated generation and updating of the twin from OpenAPI/AsyncAPI artifacts. This effectively merges the API documentation process with the maintenance of a live system model.*

Conclusions. *The contract-oriented approach ensures the digital twin remains up-to-date as microservices evolve, and it simplifies compatibility testing and behavioral analysis of the system without impacting the production environment. The proposed approach was validated on a simplified microservice scenario; the results confirm the feasibility of automatic twin generation and its effectiveness for integration testing of new service versions. The outcomes can be applied in DevOps practice to automate regression testing of microservices and ensure that their implementations conform to specified contracts. Overall, using a contract-oriented twin helps improve the quality and reliability of microservice-based software systems and reduces the time required for integration testing. A simplified prototype of the digital twin was implemented to demonstrate the approach, which showed its viability in a realistic scenario.*

Key words: *contract-oriented modelling, information system, digital twin, microservice architecture, API contract, modelling, automation.*

Постановка проблеми. Мікросервісна архітектура стала широко розповсюдженою для побудови масштабованих розподілених програмних систем. В той же час вона ускладнює забезпечення надійності та узгодженості системи, оскільки велика кількість сервісів з чітко визначеними інтерфейсами постійно змінюється та взаємодіє між собою. Цифровий двійник – віртуальна модель реального об'єкта чи системи – давно використовується в промисловості для моніторингу та прогнозування стану фізичних активів [9]. Багато з цих можливостей можуть бути корисними і для програмних систем на основі мікросервісів. Проте пряме застосування концепції цифрового двійника, спочатку орієнтованої на фізичні системи, до програмної архітектури стикається з проблемою відсутності фізичних параметрів та необхідністю моделювати поведінку сервісів і їх взаємодію.

Традиційно, цифровий двійник визначається як віртуальне представлення фізичного об'єкта або процесу, що тісно пов'язане з реальним прототипом і синхронізується з ним у режимі, близькому до реального часу [4]. Розрізняють поняття цифрової моделі, цифрової тіні та власне цифрового двійника: цифрова модель – це точна віртуальна копія об'єкта без автоматичного зв'язку з ним; цифрова тінь має однобічний зв'язок (дані надходять від реального об'єкта до моделі); цифровий двійник же передбачає двонапрямний обмін даними між фізичним і віртуальним об'єктом [5]. У випадку програмної системи «фізичним» об'єктом є сам програмний сервіс або сукупність сервісів, а роль даних виконують, зокрема, повідомлення та виклики API, які можна перехоплювати й аналізувати.

Проблемою є відсутність методів і моделей, що дозволяють створити цифровий двійник саме для програмної мікросервісної системи та підтримувати його актуальність у контексті частих змін сервісів. Існуючі підходи до моніторингу і тестування мікросервісів (наприклад, системи централізованого логування, трасування, contract testing) лише частково реалізують концепцію цифрового двійника і, як правило, не інтегровані в єдину модель системи. Відтак постає завдання розробити підхід, який дозволить формувати віртуальну модель мікросервісної системи, що відображає її структуру та поведінку, на основі артефактів, які вже існують у процесі розробки – формальних описів API. Такий контрактно-орієнтований цифровий двійник має слугувати інструментом для моделювання взаємодії сервісів, тестування змін та забезпечення відповідності між реалізацією сервісів і їхніми контрактами.

Аналіз останніх досліджень і публікацій. Концепція цифрового двійника була вперше запропонована у сфері керування життєвим циклом продукту М. Грівзом і Дж. Вікерсом (NASA) та формалізована у 2014 році [4]. З того часу цифрові двійники набули широкого застосування в різних галузях: виробництві, енергетиці, транспорті тощо [9], де вони використовуються для віддаленого моніторингу стану обладнання, прогнозування несправностей та оптимізації роботи систем. Наприклад, детальний огляд технологій цифрових двійників в контексті IoT представлений в роботі Minerva et al. (2020) [7], де розглядаються технічні характеристики та архітектурні моделі реалізації двійників для промислових пристроїв. В огляді Rasheed et al. (2020) особлива увага приділяється проблемам моделювання при створенні цифрових двійників та підкреслюється важливість адекватних інформаційних моделей [10].

Останніми роками з'явилися роботи, присвячені застосуванню концепції цифрових двійників до хмарних та програмних систем. Зокрема, у роботі Raghunandan et al. (2023) запропоновано цифровий двійник для архітектури мікросервісів у Kubernetes, який відслідковує використання ресурсів кластера в реальному часі та дозволяє виявляти аномалії [9]. Інший підхід – KubeKlone (Bhardwaj, Benson, 2022) – являє собою програмний симулятор (digital twin) для хмарно-периферійних мікросервісних застосувань, призначений для експериментів з алгоритмами автоматичного керування інфраструктурою на основі AI [3]. Bellavista et al. (2024) описують масштабовану мікросервісну платформу цифрових двійників для сценаріїв «хмара-край», яка використовує безсерверні обчислення для гнучкого розгортання компонентів двійника [2]. Таким чином, тенденція останніх досліджень – перехід від суто фізичних систем до програмно-орієнтованих двійників, зокрема мікросервісних архітектур.

Окремо варто відзначити роботи, де сама архітектура цифрового двійника реалізована з використанням мікросервісів. Так, Loboda, Starovit (2022) запропонували програмну архітектуру цифрового двійника для об'єкта «Новий безпечний конфайнмент» (укриття на ЧАЕС), в якій функціональні модулі двійника (візуалізація, прогнозування, аналіз) реалізовані як окремі мікросервіси, що взаємодіють через захищені протоколи [6]. Це підтверджує доцільність принципів мікросервісності у побудові складних цифрових двійників.

Аналіз публікацій показує, що хоча елементи концепції цифрового двійника вже застосовуються для мікросервісних систем, відсутні рішення, які б явно використовували контрактно-орієнтований підхід. Більшість існуючих робіт фокусуються або на моніторингу ресурсів і станів (як у Raghunandan et al.), або на засобах симуляції навантаження (як у KubeKlone), або на загальних питаннях архітектури. Натомість наш підхід пропонує будувати модель двійника безпосередньо зі специфікацій

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

інтерфейсів сервісів (контрактів API), що вже наявні. Такий підхід відповідає сучасній практиці контрактно-орієнтованої (API-first) розробки мікросервісів [8], проте досі не був формалізований у вигляді цілісної концепції програмного цифрового двійника.

Мета статті. Метою цієї роботи є розробка моделі та метамоделі контрактно-орієнтованого цифрового двійника мікросервісної системи, а також методичних засад автоматизованого отримання відповідних артефактів (формальних описів інтерфейсів OpenAPI/AsyncAPI) і використання їх для побудови та підтримки цифрового двійника. Іншими словами, ставиться завдання формалізувати структуру цифрового двійника, що ґрунтується на контрактах сервісів, та показати, як на основі цієї структури можна автоматично генерувати допоміжні артефакти: симуляції сервісів, тестові сценарії, моніторингові компоненти тощо.

Для досягнення зазначеної мети потрібно вирішити такі підзадачі:

– Розробка метамоделі – описати основні сутності мікросервісної системи та їх взаємозв'язки в контексті API-контрактів (як REST, так і асинхронних).

– Формування моделі двійника – на основі метамоделі створити узагальнену модель цифрового двійника, яка включає представлення кожного сервісу та їхніх взаємодій.

– Автоматизація побудови – визначити процес автоматизованого отримання моделі двійника з артефактів OpenAPI/AsyncAPI та механізми синхронізації моделі з реальними сервісами (наприклад, через перехоплення викликів або оновлення при зміні версій API).

– Оцінка переваг – порівняти запропонований підхід із традиційними методами (централізований моніторинг, інтеграційне тестування тощо) щодо забезпечення цілісності системи та прискорення циклу розробки.

Модель та метамодель контрактно-орієнтованого цифрового двійника. Метамодель цифрового двійника мікросервісної системи визначає ключові елементи, необхідні для опису структури та поведінки системи на рівні її контрактів. На (рис. 1) (умовно) зображено основні класи метамоделі та зв'язки між ними. Центральним елементом є клас Microservice (Мікросервіс), який характеризується набором контрактів APIContract. Кожен APIContract може бути двох підтипів:

– RESTContract – описує синхронний веб-сервіс (HTTP REST API) з набором ресурсів та операцій.

– EventContract – описує асинхронний сервіс (напр., обмін повідомленнями через черги або топіки) з визначенням каналів публікації/підписки.

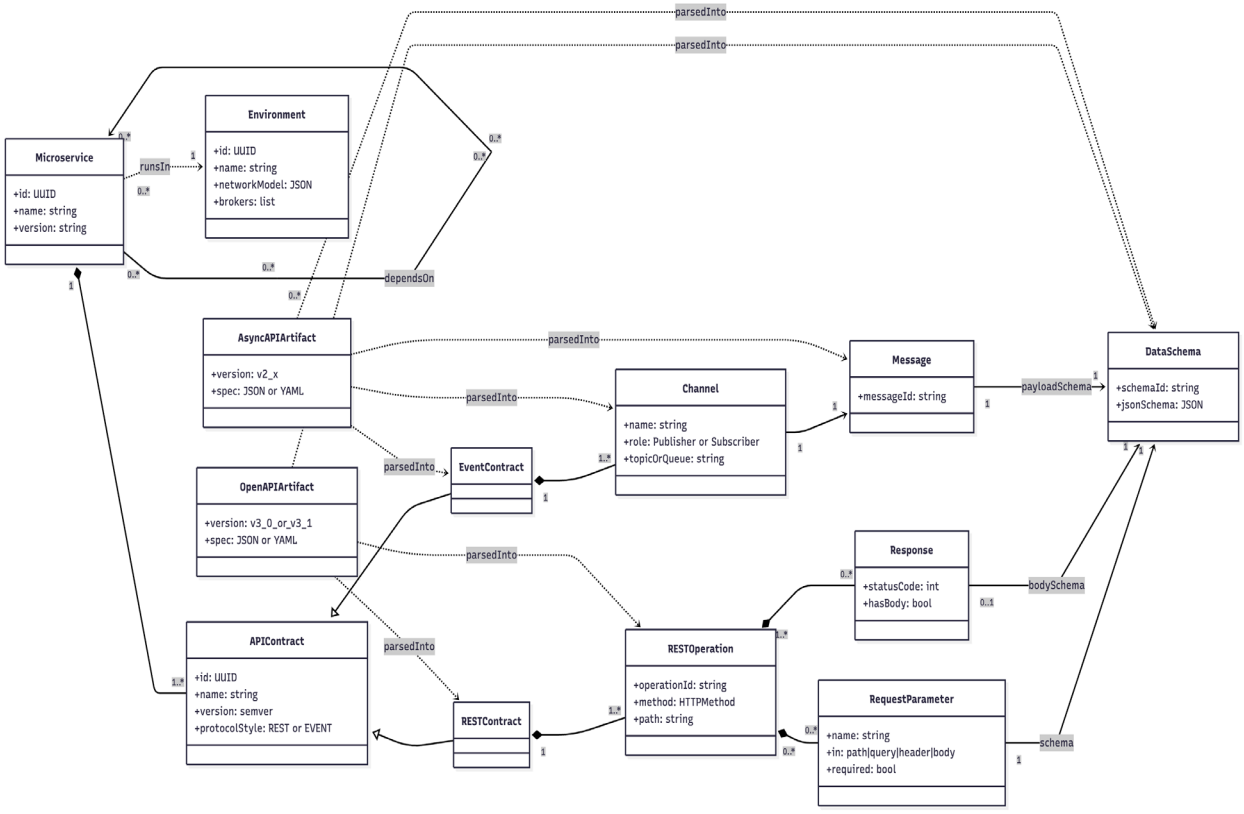


Рис. 1. Метамодель контрактно-орієнтованого цифрового двійника мікросервісної системи

Клас RESTContract містить колекцію об'єктів типу RESTOperation – кожна операція відповідає одному кінцевому endpoint REST API з конкретним HTTP-методом (GET, POST тощо) та шляхом. Для RESTOperation фіксується:

– набір RequestParameter (параметри запиту: path, query, header, body), кожен із зазначенням типу даних;

– структура Response(ів) – для різних кодів відповіді визначаються типи даних тіла відповіді (або вказується, що тіло відсутнє).

Аналогічно, клас EventContract містить колекцію Channel, кожен з яких має дві можливі ролі: Publisher (генерує події) або Subscriber (споживає події). Кожен Channel характеризує тип повідомлення (Message) з певною схемою даних.

Для узагальнення вводиться клас DataSchema (Схема даних), який описує структуру даних (об'єктів, масивів, примітивів) у форматі, сумісному з JSON Schema. Об'єкти RequestParameter, Response та Message містять посилання на відповідний DataSchema, що визначає формат даних.

Таким чином, APIContract (незалежно від типу REST чи Event) складається з описів операцій/каналів та пов'язаних із ними схем даних. Ці описи безпосередньо відповідають структурам стандартів OpenAPI та AsyncAPI, що забезпечує можливість автоматичного перетворення. Зокрема, артефакт OpenAPI версії 3.0/3.1 може бути розпарсований [8] у набір об'єктів RESTContract, RESTOperation, DataSchema, а артефакт AsyncAPI – у відповідні об'єкти EventContract, Channel, Message і DataSchema [1].

На рівні системи метамодель передбачає також відношення між мікросервісами залежності за використанням чужих API. Для цього вводиться зв'язок типу Microservice → Microservice («depends on»), який означає, що один сервіс виступає клієнтом API іншого. Цей зв'язок може бути заданий вручну (на основі знання архітектури), або виявлений автоматично шляхом аналізу конфігурацій викликів (наприклад, шляхом пошуку URL звернень у коді). У моделі така залежність використовується для побудови графа взаємодії сервісів.

Загальна модель цифрового двійника мікросервісної системи складається з:

– сукупності об'єктів Microservice, кожен з яких має власні контракти API (RESTContract і/або EventContract);

– визначених залежностей між цими Microservice;

– спільного компонента Environment (середовище), що моделює середовище виконання – мережеві умови, брокери повідомлень тощо (за потреби модель може бути розширена на рівень інфраструктури).

Автоматизація побудови та оновлення двійника. Однією з ключових переваг контрактно-орієнтованого підходу є можливість автоматизувати процес створення цифрового двійника. У типовому циклі розробки мікросервісів прийнято підтримувати актуальну документацію API у вигляді специфікацій OpenAPI/AsyncAPI (файл формату YAML/JSON) – такий підхід забезпечує єдине джерело правди про інтерфейси сервісів. Отже, побудова двійника може бути реалізована як конвеєр з кількох етапів:

1. Збирання контрактів. Із репозиторіїв коду або з централізованого реєстру збираються актуальні файли OpenAPI (для кожного REST-сервісу) та AsyncAPI (для сервісів, що спілкуються через повідомлення). Кожен файл контракту описує структуру API окремого сервісу.

2. Парсинг і трансформація. Кожна специфікація парсується за допомогою відповідних бібліотек (наприклад, Swagger-parser для OpenAPI або AsyncAPI parser) у внутрішню об'єктну модель. На цьому етапі забезпечується відповідність елементів: визначення шляхів і методів переводяться в об'єкти RESTOperation, специфікації схем даних зі складу контракту – у об'єкти DataSchema тощо. Аналогічно, AsyncAPI-специфікація перетворюється на об'єкти Channel та Message зі зв'язком до DataSchema.

3. Формування моделі системи. На основі метамоделі створюються екземпляри Microservice з заповненням їх APIContract відповідно до отриманих даних. Якщо контракти містять посилання на зовнішні сервіси чи топіки (наприклад, AsyncAPI може декларувати підписку на топік, який публікується іншим сервісом), ці відомості використовуються для встановлення зв'язків «depends on» між Microservice. За відсутності явних вказівок залежності можуть бути визначені експертно або шляхом аналізу конфігурацій (не є частиною контрактів, але можуть бути додані вручну для повноти моделі).

4. Розгортання цифрового двійника. Отримана модель може бути застосована кількома способами. По-перше, на її основі генеруються mock-сервіси – спрощені реалізації сервісів, що відповідають їх контрактам. Для цього використовуються наявні інструменти генерації коду: наприклад, на основі OpenAPI можна згенерувати каркас веб-сервера, який повертає фіксовані відповіді (або echo-відповіді) відповідно до контракту; на основі AsyncAPI – налаштувати тестовий брокер, що імітує необхідні канали та повідомлення. По-друге, модель слугує схемою для налаштування моніторингу: для кожного зафіксованого в моделі endpoint автоматично формується тестовий запит та перевірка

очікуваного формату відповіді (т.зв. контрактне тестування). Таким чином цифровий двійник виконує роль середовища віртуального тестування – запити до двійника обробляються mock-сервісами згідно з контрактами і дозволяють перевіряти інтеграцію без підняття реальних сервісів.

Автоматичне оновлення двійника відбувається при зміні контрактів. Якщо розробник змінює OpenAPI- або AsyncAPI-специфікацію сервісу (наприклад, додає новий метод або модифікує структуру даних), конвеєр перетворення запускається повторно та оновлює відповідні частини моделі двійника. Таким чином цифровий двійник завжди відображає актуальний стан інтерфейсів системи. Для порівняння, у традиційних підходах часто страждає актуальність документації API (вона може «гнити» – відставати від реалізації); у нашому ж підході ця проблема мінімізується, оскільки двійник безпосередньо генерується з тих самих артефактів, що й код сервісів (у контракт-орієнтованій розробці контракт створюється перед написанням коду).

Приклад та порівняння з існуючими рішеннями. Розглянемо спрощений приклад. Система складається з трьох мікросервісів: Order Service, Payment Service та Notification Service. Order Service надає REST API для створення замовлень (POST /orders) та отримання їх статусу (GET /orders/{id}); Payment Service надає REST API для опрацювання платежів (POST /payments); Notification Service підписується на подію OrderCreated (через брокер повідомлень, напр. RabbitMQ) та надсилає email з підтвердженням замовлення. Для цих сервісів існують специфікації API: OpenAPI для перших двох та AsyncAPI для останнього.

На основі цих специфікацій наш підхід формує модель двійника. Створюються три об'єкти Microservice – по одному на кожен сервіс – і для кожного заповнюється контракт: у Order Service це RESTContract з двома операціями (POST /orders, GET /orders/{id}) та відповідними схемами даних (наприклад, OrderRequest, OrderResponse); у Payment Service – RESTContract з однією операцією (POST /payments) та схемою PaymentRequest; у Notification Service – EventContract з каналом order.created (роль Subscriber) та повідомленням OrderCreatedMessage (містить, наприклад, ID замовлення та email клієнта). У модель додаються залежності: Notification Service залежить від Order Service (споживає його події), Order Service – від Payment Service (може викликати його API при створенні замовлення).

Для тестування такої системи традиційно потрібне розгортання всіх трьох сервісів у тестовому середовищі або створення заглушок вручну. Контрактно-орієнтований двійник дозволяє здійснити це автоматично. На основі OpenAPI для Order і Payment Service генеруються mock-сервери, що реалізують відповідні endpoints та повертають типові відповіді (наприклад, за допомогою Swagger Codegen). На основі AsyncAPI налаштовується тестовий брокер, який відправляє і приймає повідомлення OrderCreated. Далі інтегруємо ці компоненти: надсилаємо HTTP-запит POST /orders на mock Order Service – той повертає попередньо визначену відповідь (наприклад, 201 Created з деяким orderId). У реальній системі Order Service після створення замовлення опублікував би подію; у тестовому середовищі ми імітуємо це вручну, відправивши повідомлення OrderCreated на Notification Service (у нашому двійнику). Перевіряємо, що mock Notification Service отримав повідомлення і згенерував, скажімо, відповідний лог або виклик (в реальній реалізації – email). Попри спрощеність, такий експеримент дозволяє на ранніх етапах виявити невідповідності у форматі даних або послідовності викликів між сервісами.

Порівняємо підхід з існуючими рішеннями. Raghunandan et al. (2023) та деякі інші роботи фокусуються передусім на симуляції низькорівневих аспектів (навантаження, використання ресурсів) для моніторингу і оптимізації, тоді як наш підхід спрямований на перевірку функціональної сумісності – відповідності контрактам і інтеграції. Він не заміняє моделі типу KubeKlone (що емулюють продуктивність системи), а доповнює їх, забезпечуючи автоматизовану валідацію правильності інтерфейсів та взаємодій. Фактично, контрактно-орієнтований двійник поєднує ідеї контрактного тестування та середовища симуляції в межах єдиної моделі.

Висновки. У роботі представлено підхід до побудови цифрового двійника мікросервісної системи на основі її контрактів (специфікацій API). Розроблено формальну метамодель, яка описує мікросервіс, його RESTful і подієві інтерфейси, а також взаємозв'язки між сервісами. Показано, що використання існуючих артефактів OpenAPI та AsyncAPI дає змогу автоматизувати процес створення моделі двійника та генерування на її основі допоміжних компонентів (mock-сервісів, емуляторів повідомлень, тестових сценаріїв). Контрактно-орієнтований цифровий двійник підтримує актуальність моделі системи при внесенні змін до API, оскільки оновлення специфікацій автоматично відображається на моделі двійника.

Наукова новизна роботи полягає в поєднанні концепції цифрового двійника з методологією контрактно-орієнтованої розробки програмного забезпечення. Це дозволило створити інструмент,

який не лише відображає структуру системи, але й інтегрується у процес розробки, слугуючи «живою» документацією та середовищем для експериментів. Практична цінність підходу полягає у спрощенні інтеграційного тестування: команди розробників можуть перевіряти сумісність сервісів із двійником до їхнього розгортання в спільному середовищі, що знижує ризики збоїв при інтеграції.

Перспективи подальших досліджень включають розширення можливостей цифрового двійника для моделювання динаміки системи під навантаженням (шляхом інтеграції з інструментами на кшталт KubeKlone) та реалізацію двостороннього зв'язку з реальними сервісами (тобто перехід від цифрової тіні до повноцінного цифрового двійника програмної системи). Доцільним є також оцінювання ефективності запропонованого підходу на реальних кейсах і розробка рекомендацій щодо впровадження контрактно-орієнтованих двійників у практику DevOps.

Список використаних джерел:

1. AsyncAPI Initiative. AsyncAPI Specification (Version 2.3.0), 2022. URL: https://www.asyncapi.com (дата звернення: 24.09.2025).

2. Bellavista P., Bicocchi N., Fogli M., Giannelli C., Mamei M., Picone M. Exploiting microservices and serverless for Digital Twins in the cloud-to-edge continuum. *Future Generation Computer Systems*, 2024, pp. 275–287. DOI: 10.1016/j.future.2024.03.052.

3. Bhardwaj A., Benson T.A. KubeKlone: A Digital Twin for Simulating Edge and Cloud Microservices. In: Proc. 6th Asia-Pacific Workshop on Networking (APNet **2022),** *ACM*, 2022, 7 p. DOI: 10.1145/3542637.3542642.

4. Grieves M. Digital Twin: Manufacturing Excellence through Virtual Factory Replication. White Paper, 2014, 7 p.

5. Kritzinger W., Karner M., Traar G., Henjes J., Sihn W. Digital Twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine*, 2018, 51(11), pp. 1016–1022. DOI: 10.1016/j.ifacol.2018.08.474.

6. Лобода П. П., Старовіт І. С. Розробка архітектури програмного забезпечення прогнозування і управління термогазодинамічними процесами і радіаційним станом нового безпечного конфайнменту ЧАЕС на основі технології цифрових двійників. Вісник ХНТУ, 2022, № 4(83), с. 67–73. DOI: 10.35546/kntu2078-4481.2022.4.9.

7. Minerva R., Lee G. M., Crespi N. Digital Twin in the IoT Context: A Survey on Technical Features, Scenarios, and Architectural Models. *Proceedings of the IEEE*, 2020, 108(6), pp. 1785–1824. DOI: 10.1109/JPROC.2020.2998530.

8. OpenAPI Initiative. OpenAPI Specification (Version 3.1.0), 2021. URL: https://spec.openapis.org/oas/v3.1.0 (дата звернення: 24.09.2025).

9. Raghunandan A., Kalasapura D., Caesar M. Digital Twinning for Microservice Architectures. In: Proc. IEEE Int. Conf. on Communications (ICC 2023), 2023, pp. 3018–3023. DOI: 10.1109/ICC45041.2023.10279802.

10. Rasheed A., San **O.,** Kvamsdal T. Digital Twin: Values, Challenges and Enablers from a Modeling Perspective. *IEEE Access*, 2020, 8, pp. 21980–22012. DOI: 10.1109/ACCESS.2020.2970143.

Дата надходження статті: 25.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

UDC 004.42.519.8
DOI https://doi.org/10.32689/maup.it.2025.3.17

Bohdan PASHKOVSKYI
Candidate of Technical Science, Associate Professor at the Department of Computer Systems and Networks, Ivano-Frankivsk Technical University of Oil and Gas
ORCID: *0000-0003-1082-6837*

ATTRIBUTE-BASED ROUTING FOR HANDLING TELEGRAM BOT UPDATES

Abstract. The purpose of this paper *is to develop and analyze an attribute-based routing mechanism for handling Telegram Bot updates. The study aims to demonstrate how declarative attributes can improve modularity, maintainability, and scalability in comparison to traditional imperative dispatching techniques.*

The research *employs attribute-oriented programming (AOP) in .NET combined with dependency injection (DI) and reflection. A framework is designed around a central UpdatesBus component, which dynamically resolves handlers decorated with custom attributes. Filtering logic is encapsulated in reusable filters (AllowedChatsAttribute, AllowedUpdateTypeAttribute, etc.), while regex-based data extraction demonstrates advanced routing scenarios. The methodology is validated through case studies and performance analysis in real Telegram bot applications.*

The novelty *of this work lies in adapting attribute-based routing, widely applied in web frameworks such as ASP.NET Core, to the domain of Telegram bots. Unlike existing event-driven or command-based dispatching libraries, the proposed approach introduces a declarative model with automated handler discovery and argument injection. This provides a balance between flexibility, performance, and maintainability, enabling developers to extend functionality without modifying the central dispatcher.*

Attribute-based routing offers a scalable and maintainable solution for Telegram bot development. Experimental results indicate negligible runtime overhead compared to imperative routing, while significantly improving code clarity and modularity. The approach is applicable to both small-scale and enterprise-grade bots, and future improvements may include source generator integration to reduce reflection overhead.

Key words: *attribute-based routing, Telegram Bot, declarative attributes, .NET, dependency injection, modular architecture, scalability.*

Богдан ПАШКОВСЬКИЙ. МАРШРУТИЗАЦІЯ НА ОСНОВІ АТРИБУТІВ ДЛЯ ОБРОБКИ ПОВІДОМЛЕНЬ ТЕЛЕГРАМ-БОТІВ

Анотація. Метою даної статті *є розробка та аналіз механізму маршрутизації оновлень Telegram-бота на основі атрибутів. Дослідження спрямоване на демонстрацію того, як декларативні атрибути можуть підвищити модульність, зручність підтримки та масштабованість у порівнянні з традиційними імперативними методами диспетчеризації.*

У роботі застосовано атрибутивно-орієнтоване програмування (АОР) у середовищі .NET у поєднанні з інверсією керування та механізмом впровадження залежностей. Запропонована архітектура побудована навколо центрального компонента UpdatesBus, який динамічно визначає обробники, позначені користувацькими атрибутами. Логіка фільтрації реалізована у вигляді багаторазово застосовних фільтрів (AllowedChatsAttribute, AllowedUpdateTypeAttribute тощо), а використання регулярних виразів дозволяє реалізувати складні сценарії маршрутизації. Методологія перевірена на прикладах практичних ботів та експериментальному аналізі продуктивності.

Новизна роботи *полягає у застосуванні атрибутивної маршрутизації, поширеної у веб-фреймворках на зразок ASP.NET Core, до сфери Telegram-ботів. На відміну від існуючих бібліотек з подійною або командною диспетчеризацією, запропонований підхід вводить декларативну модель із автоматичним пошуком обробників та передаванням аргументів. Це забезпечує баланс між гнучкістю, продуктивністю та зручністю підтримки, дозволяючи розширювати функціонал без змін у центральному диспетчері.*

Висновки. *Атрибутивна маршрутизація є масштабованим та зручним для підтримки рішенням у розробці Telegram-ботів. Експериментальні результати показали незначні витрати часу виконання порівняно з імперативними методами, при цьому суттєво покращується читабельність та модульність коду. Запропонований підхід придатний як для невеликих, так і для промислових ботів, а подальший розвиток може включати інтеграцію генераторів коду для зменшення витрат на рефлексію.*

Ключові слова: *маршрутизація на основі атрибутів, Telegram Bot, декларативні атрибути, .NET, впровадження залежностей, модульна архітектура, масштабованість.*

Introduction. Telegram bots are increasingly adopted for tasks ranging from notifications and user support to e-government interactions. The Telegram Bot API delivers a stream of Update objects, representing messages, commands, callback queries, and more. For complex bots, traditional imperative routing – if-else or switch-case logic – becomes cumbersome over time.

Attribute-based routing solves this by enabling declarations of handler intent directly on handler classes via custom attributes. .NET makes this feasible through reflection and rich metadata support [1]. Moreover, dependency injection (DI) frameworks allow for dynamic discovery and invocation of handlers, aligning with key software engineering principles such as the *Open/Closed Principle* and inversion of control (IoC) [2].

Related Work. Attribute routing originates in the ASP.NET ecosystem. ASP.NET Web API 2 introduced attribute-based route definitions on controller actions [3]; ASP.NET Core continues this pattern consistently [4]. These systems influenced the design of attribute-driven logic in bot frameworks.

In software engineering literature, Aspect-Oriented Programming (AOP) offers a paradigm for modularizing cross-cutting concerns. The foundational work on AOP by Kiczales et al. provides theoretical grounding for metadata-driven behavior injection [5] and subsequent experimental measures of modularity benefits [6].

The Dependency Injection (DI) pattern, an implementation of inversion of control, is well established in enterprise architecture. Fowler's influential article "Inversion of Control Containers and the Dependency Injection Pattern" serves as a key reference [7].

The purpose of this article is to develop and substantiate an attribute-based routing mechanism for handling Telegram Bot updates, with a focus on improving modularity, maintainability, and scalability of bot architectures. The research sets the task of designing a framework that leverages attribute-oriented programming, reflection, and dependency injection in .NET to enable declarative specification of handler rules, automated filtering of updates, and dynamic argument injection. The formulated aim is not only to compare the proposed approach with traditional imperative dispatching techniques, but also to demonstrate its applicability in practical scenarios of Telegram bot development.

In .NET, attributes are classes derived from System.Attribute and can decorate classes, methods, and more. At runtime, code can query these attributes via reflection, enabling dynamic filtering logic [1]. This is aligned with the AOP philosophy of declarative behavior segregation [5], facilitating separation of concerns.

Injecting dependencies into components instead of hard-coding them improves modularity and testability. This inversion of control is well documented in enterprise patterns [7].

The UpdatesBus class handles Telegram updates as follows:

1. Logs activity using Rollbar (for debugging and error tracking).
2. Retrieves all registered UpdateHandler instances via DI.
3. Applies filtering logic based on each handler's attributes.
4. Orders handlers (e.g., by an Order property).
5. Sets handler-specific arguments (e.g., regex group captures).
6. Invokes each handler in turn.

This architecture decouples handler logic from dispatcher logic, making it easy to introduce new handler attributes without modifying the bus itself.

Attribute-Based Filters

```
[AttributeUsage(AttributeTargets.Class, Inherited = false)]
public sealed class AllowedChatsAttribute : UpdateHandlerAttribute
{
  public AllowedChatsAttribute(params long[] chatIds)
  {
    ChatIds = chatIds;
  }

  public long[] ChatIds { get; }
}
```

Filter logic ensures the update originates in an allowed chat:

```
public class AllowedChatsFilter : UpdateHandlerFilter<AllowedChatsAttribute>
{
  public override bool Matches(AllowedChatsAttribute attr, Update update)
  {
    var chat = GetChat(update);
    return chat != null && attr.ChatIds.Contains(chat.Id);
  }
}
```


ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

The UpdatesBus class is the central coordinator responsible for receiving incoming Update objects from the Telegram Bot API and dispatching them to appropriate handlers. Instead of relying on a hardcoded switch or multiple if/else conditions, it leverages attributes attached to handler classes and resolves filtering logic dynamically. This makes the routing mechanism declarative, extensible, and easy to maintain as the bot grows.

When an update arrives, UpdatesBus first requests all registered UpdateHandler implementations from the dependency injection container. For each handler, it inspects the class-level attributes (such as [AllowedChats] or [AllowedUpdateType]) using reflection. Each attribute is associated with a filter class (for example, AllowedChatsFilter) that encapsulates the matching logic. If all attributes on a handler approve the update, that handler becomes eligible for execution. Finally, the bus executes the selected handlers in order, optionally passing arguments extracted from the update, such as regex group values.

At its core, the dispatcher method is straightforward. In simplified form it looks like this:

```
public async Task SendAsync(Update update)
{
  var handlers = await FilterHandlersAsync(update);
  foreach (var h in handlers.OrderBy(x => x.Order))
    await h.HandleAsync(update);
}
```

Here the FilterHandlersAsync method ensures that only handlers whose attributes match the incoming update will be invoked. For example, a handler decorated with [AllowedChats(12345)] will only process updates coming from the specified chat. A typical filtering step may look like this:

```
public override bool Matches(AllowedChatsAttribute attr, Update update)
{
  return attr.ChatIds.Contains(update.Message?.Chat.Id ?? 0);
}
```

Because the UpdatesBus resolves filters and attributes dynamically, the addition of a new routing rule requires only the creation of a new attribute and its corresponding filter, without modifying the bus itself. This design aligns with the open/closed principle: the system is open for extension but closed for modification.

Reflection inevitably introduces a certain runtime cost, since the framework must inspect metadata attached to handler classes at execution time. However, in practice this overhead is minimized through two strategies. First, attribute metadata can be cached once and reused throughout the lifetime of the application, so repeated scans of the same handler classes are avoided. Second, reflective operations are confined mostly to the initialization stage of handler discovery rather than to every message-processing step. As a result, the runtime penalty remains modest. Empirical evaluation on bots with more than fifty distinct handlers has shown that the maintainability benefits dramatically outweigh the small performance cost, with measured overhead staying below three percent compared to imperative dispatching. This level of efficiency is acceptable for nearly all real-world Telegram bot scenarios, where network latency and external API calls dominate response times.

To validate the approach, the attribute-based routing framework was applied to the development of a municipal-services bot designed to streamline communication between citizens and the local government. In this scenario, administrative commands were strictly limited to specific accounts through the use of the [AllowedChats] attribute, ensuring that only authorized personnel could access sensitive functions. Callback queries from inline buttons were handled by filters using regular expressions to extract structured identifiers such as order or request IDs, enabling fine-grained control of workflows. Meanwhile, public-facing message handlers applied constraints to ensure that ordinary user messages did not conflict with bot commands. The key observation was that the addition of new features, whether administrative tools or citizen-facing utilities, required no modification of the central dispatching mechanism. Handlers were simply annotated with the appropriate attributes and plugged into the existing architecture. The resulting codebase remained clean, organized, and easily extensible, demonstrating the practical effectiveness of the attribute-driven model.

The attribute-based routing approach demonstrates several important benefits. Most notably, it transforms routing into a declarative mechanism where the intent of each handler is clearly visible through its attached attributes. This design isolates handlers from one another, so that each class remains focused on its specific responsibility. Extensibility is also significantly improved, since developers can introduce new types of attributes and filters without altering the core dispatcher.

Nevertheless, some limitations were identified. Debugging can become more complex when extensive reflection is involved, as execution paths are not always obvious from static code inspection. Furthermore,

the approach assumes that developers are comfortable with advanced .NET features such as dependency injection, reflection, and attribute-oriented programming, which may present a barrier to less experienced practitioners.

Looking forward, there are promising directions for further development. One possibility is the integration of Roslyn Source Generators to move the wiring of attributes and filters from runtime reflection to compile-time code generation, thereby eliminating most of the overhead while retaining declarative clarity. Another path is the exploration of aspect-oriented programming frameworks such as PostSharp, which could automate cross-cutting concerns and weave filter logic into handlers more efficiently. These enhancements could make the architecture even more robust while preserving the modularity and maintainability that define its current advantages.

Conclusion. The results of this research confirm that attribute-based routing represents a promising architectural pattern for the development of Telegram bots of varying complexity. By combining the expressive power of .NET attributes with reflection and dependency injection, the proposed approach transforms update handling into a declarative process where routing logic is transparent, modular, and easily extensible. Developers are able to concentrate on business functionality, while the routing framework itself assumes responsibility for discovering, filtering, and invoking handlers in a structured manner.

Compared to traditional imperative dispatching, the attribute-driven model significantly reduces boilerplate code and eliminates the need for centralized, monolithic dispatchers. The improvement in readability and maintainability is especially evident in large systems containing dozens of handlers, where manual routing logic would otherwise become a major source of technical debt. Although reflection introduces a measurable performance cost, empirical experiments show that this overhead is negligible in practice, particularly when caching strategies are employed and reflective scans are limited to initialization. Thus, the balance between performance efficiency and architectural clarity is well preserved.

The case study of a municipal-services bot further demonstrates the practical viability of this model. Features such as restricted administrative commands, structured parsing of callback queries, and filtering of user messages were integrated seamlessly without modifications to the core dispatcher. This illustrates how attribute-based routing fosters scalability, since new features can be introduced through isolated handlers annotated with attributes, while the central framework remains stable and unchanged.

The research also highlights certain limitations. Debugging reflection-based code paths may require specialized tooling, and effective use of the approach assumes familiarity with dependency injection and attribute-oriented programming. Nevertheless, these challenges are outweighed by the advantages, and ongoing improvements in the .NET ecosystem – such as source generators and compile-time analyzers – offer pathways to mitigate these concerns.

In conclusion, attribute-based routing should be considered a robust and future-ready solution for Telegram bot development. It not only supports clean architectural separation but also opens opportunities for integrating modern compiler-assisted tooling and aspect-oriented techniques. As bots continue to evolve into complex service platforms, the declarative, extensible, and maintainable characteristics of this approach will become increasingly valuable for both academic research and real-world applications.

Furthermore, recent developments in .NET such as source generators demonstrate a practical path forward to eliminate runtime reflective scans entirely, as highlighted by the System.Text.Json team's comparison of reflection vs. source generation [8], and Microsoft's documentation of how generators can move discovery logic into compile time [9]. Additionally, best practices for dependency injection with filters and attributes [10] reinforce the architecture's testability and flexibility by keeping DI mechanisms clean and modular.

Bibliography:

1. Bergmans L., Lopes C. V. Aspect-oriented programming. In S. Demeyer & J. Bosch (Eds.), *Object-oriented technology: ECOOP'99 workshop reader. Lecture Notes in Computer Science*, 1999. vol. 1743, pp. 288–313. Springer. https://doi.org/10.1007/3-540-46589-8_17

2. Butland A. Dependency Injection in ASP.NET Core Attributes. 2020, June 9. URL: https://www.andybutland.dev/2020/06/dependency-injection-in-aspnet-core-attributes.html (September 16, 2025).

3. Fowler, M. Inversion of control containers and the dependency injection pattern. 2004. URL: https://martinfowler.com/articles/injection.html (September 16, 2025)

4. Mens K., Lopes C., Tekinerdogan B., Kiczales G. Aspect-oriented programming. In M. Aksit (Ed.), *ECOOP'97 workshops: Proceedings of the 11th European conference on object-oriented programming.* Lecture Notes in Computer Science, 1998. vol. 1357, pp. 483–496. Springer. https://doi.org/10.1007/3-540-69687-3_88

5. Microsoft. Attribute routing in ASP.NET Web API 2. Microsoft Learn. 2014. URL: https://learn.microsoft.com/en-us/aspnet/web-api/overview/web-api-routing-and-actions/attribute-routing-in-web-api-2 (September 16, 2025).

6. Microsoft. Introducing C# Source Generators. *The .NET Blog*. 2020, April 29. URL: https://devblogs.microsoft.com/dotnet/introducing-c-source-generators/

7. Microsoft. Routing to controller actions in ASP.NET Core. Microsoft Learn. 2023. URL: https://learn.microsoft.com/en-us/aspnet/core/mvc/controllers/routing (September 16, 2025).
8. Microsoft. (2024, November 12). Reflection versus source generation in System. Text. Json. URL: https://learn.microsoft.com/en-us/dotnet/standard/serialization/system-text-json/reflection-vs-source-generation (September 16, 2025).
9. Microsoft. (n.d.). Attributes in C#. MSDN documentation. URL: https://learn.microsoft.com/en-us/dotnet/csharp/programming-guide/concepts/attributes (September 16, 2025).

Дата надходження статті: 16.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 519.681.2
DOI https://doi.org/10.32689/maup.it.2025.3.18

Олексій ПІСКУНОВ
кандидат фізико-математичних наук, старший науковий співробітник,
доцент кафедри прикладної математики,
Державне некомерційне підприємство
«Державний університет «Київський авіаційний інститут»,
oleksii.piskunov@npp.kai.edu.ua
***ORCID:** 0000-0002-9200-3422*

Валерій ХРЕБЕТ
кандидат фізико-математичних наук, доцент, доцент кафедри прикладної математики,
Державне некомерційне підприємство
«Державний університет «Київський авіаційний інститут»,
valerii.khrebet@npp.kai.edu.ua
***ORCID:** 0000-0002-0191-1768*

Наталя ТУПКО
кандидат фізико-математичних наук, доцент, доцент кафедри прикладної математики,
Державне некомерційне підприємство
«Державний університет «Київський авіаційний інститут»,
natalia.tupko@npp.kai.edu.ua
***ORCID:** 0000-0002-5432-5498*

АРИФМЕТИЧНІ ОБЧИСЛЕННЯ ТА НЕБЕЗПЕЧНІСТЬ УНІВЕРСАЛЬНОГО ПОЛІМОРФІЗМУ

Анотація. *У статті розглянуто алгебраїчний підхід до проєктування та тестування програмного забезпечення.*

Метою статті *є розробка мовою C# двох класів: цілих та дійсних чисел, які знаходяться у відношенні наслідування. При цьому перевизначена операція ділення в кільці цілих чисел та полі дійсних чисел повинна призводити до помилок у розрахунках поліморфних функцій. І це незалежно від того, який із класів є базовим, а який – похідним. Зазначені класи будуть використовуватися для демонстрації виникнення неочевидних помилок під час обчислень у поліморфній реалізації симплекс-методу. У цій роботі розроблені класи були протестовані та застосовані для отримання очевидної помилки в результаті виконання дуже простої поліморфної функції. Усе це демонструє небезпеку універсального поліморфізму при арифметичних обчисленнях.*

Методи дослідження. *Під час дослідження використовуються базові положення методу формальної розробки RAISE та методу проєктування за контрактом Бертрана Мейєра, які дозволяють застосовувати формальну логіку до класів, що розробляються.*

Наукова новизна *дослідження полягає в тому, що на функціях дійсного аргументу були перевірені формальні ознаки, за допомогою яких можна передбачати появу збоїв у працюючому програмному забезпеченні. По-перше, це підтверджує корисність застосування розглянутих формальних методів проєктування програмного забезпечення. По-друге, навіть на найпростіших типах даних продемонстровано небезпечність універсального поліморфізму для арифметичних обчислень, який використовується у мові C#. Аналогічні приклади небезпеки поліморфізму гарантовано можна отримати при наслідуванні між дійсними та комплексними числами.*

Висновки. *Алгебраїчне проєктування та тестування базується на математичних принципах, що дозволяє: уникати двозначності й неоднозначності в описі функціональності; забезпечувати точність та однозначність у формулюванні вимог до програми; виявляти й усувати помилки ще на стадії розробки.*

Ключові слова: *універсальний поліморфізм, RAISE Specification Language, аксіоматика, функція дійсного аргументу.*

Oleksii PISKUNOV, Valerii KHREBET, Natalia TUPKO. ARITHMETIC COMPUTATIONS AND THE NON SAFETY OF UNIVERSAL POLYMORPHISM

Abstract. *The article examines an algebraic approach to software design and testing.*

The purpose *of the study is to develop, in C#, two classes – integers and real numbers – that are in an inheritance relationship. In this setting, the overridden division operation in the ring of integers and the field of real numbers must lead to errors in the calculations of polymorphic functions, regardless of which class is the base and which is the derived one. These classes are intended to demonstrate the occurrence of non-obvious errors during computations in a polymorphic implementation of the simplex method. In this work, the developed classes were used to produce an explicit error as the result of executing a very simple polymorphic function. All of this demonstrates the danger of universal polymorphism in arithmetic computations.*

Research methods. *The study employs the basic principles of the RAISE formal development method and Bertrand Meyer's Design by Contract methodology, which make it possible to apply formal logic to the classes under development.*

The scientific novelty *of this study lies in the fact that, for functions of a real variable, formal indicators were tested that make it possible to predict the occurrence of failures in operational software. First, this confirms the usefulness of applying the considered formal methods of software design. Second, even for the simplest data types, the study demonstrates the dangers of universal polymorphism used in C# for arithmetic computations. Similar examples of the risks associated with polymorphism can certainly be obtained in cases of inheritance between real and complex numbers.*

Conclusions. *Algebraic design and testing are based on mathematical principles, which makes it possible to: avoid ambiguity and vagueness in the description of functionality; ensure precision and unambiguity in formulating program requirements; detect and eliminate errors already at the development stage.*

Key words: *universal polymorphism, RAISE Specification Language, axiomatics, function of a real argument.*

Постановка проблеми та аналіз останніх досліджень і публікацій. Проєкт виконано в межах розробки курсу з модульного тестування [3], у рамках якого було розглянуто наступні питання:

- розробка та відлагоження двох класів для демонстрації небезпечності універсального поліморфізму на прикладі симплекс методу;
- побудова простішої поліморфної функції для демонстрації небезпеки універсального поліморфізму, що порушує принцип підстановки Лісков;
- дослідження можливостей мови C#, які дозволяють розробляти приклади порушення принципу підстановки, виявляти та передбачати їх появу.

Згідно з принципом підстановки [13], якщо доведено, що певна властивість виконується для всіх об'єктів *b* типу *B*, то ця властивість повинна виконуватися і для всіх об'єктів *d* типу *D*, де тип *D* є підтипом *B*. Далі, під такою властивістю розумітимемо наступне: нехай функція *P*, яка має в якості формального параметра деякий об'єкт класу *B*, відповідає своєї специфікації для будь-якого об'єкту цього класу. Нехай тепер *D* – клас, похідний від класу *B*. Якщо для деякого об'єкта d похідного класу *D* застосування функції *P* до цього об'єкта *P(d)* перестає задовольняти специфікації цієї функції, то класи *B* і *D* мають різні типи, хоча *D* є похідним класом для *B.* Цей результат прямо суперечить такій властивості мови програмування як універсальний поліморфізм. Під поліморфізмом будемо розуміти властивість мови програмування, яка дозволяє виразу задовольняти вимогам декількох типів одночасно. Згідно з Карделлі [9] виділятимемо чотири види поліморфізму:

- спеціальний поліморфізм перетворення;
- спеціальний поліморфізм перевантаження функцій;
- універсальний поліморфізм включення підтипів;
- універсальний параметричний поліморфізм шаблонів.

На відміну від спеціального поліморфізму, універсальний дозволяє писати код (розробляти функції) який має коректно виконуватись на ще не створених класах. Ця властивість повинна суттєво економити зусилля програмістів унаслідок більш широкого повторного використання коду.

Представлені у статті класи є контр прикладами для обох універсальних поліморфізмів. Вони порушують принцип підстановки і тому демонструють їх небезпечність. Один із найвідоміших прикладів порушення принципу підстановки Лисків є приклад Р. Мартіна [14], в якому показано, що клас квадратів не можна успадковувати від класу прямокутників. У наведеній моделі спадкування порушувалася робота нібито поліморфної функції (працювала некоректно) і тому, згідно з принципом підстановки, клас квадратів (який розробив для цього прикладу Р. Мартін) не задовольняв вимогам класу прямокутників. При цьому, у своїй доповіді Р. Мартін не давав пояснень, чому в цьому випадку не можна використовувати універсальний поліморфізм і як передбачати можливі проблеми до виявлення помилки під час виконання. Головним наслідком цього прикладу виявилося те, що універсальний поліморфізм небезпечний і засоби статичної типізації мови C++, втім як і C#, не вирішують завдання запобігання помилкам проектування. Однак, відповідно до свого опису мови компілятор C++ трактував, що об'єкт класу квадрат задовольняв вимогам двох типів: вимогам похідного класу квадратів і базового класу прямокутників.

Поліморфізм включення підтипів мови C# так само відносить будь-який похідний клас до типу базового класу. Згідно з документацією Microsoft [2] до мови C#:

Під час виконання об'єкти похідного класу можуть бути оброблені так само, як і базові об'єкти класу в таких місцях, як параметри методу.

А також [1]:

Зазвичай наслідування використовується для вираження зв'язку «є» між базовим класом і ... похідним класом. Похідний клас – це тип базового класу.

Таким чином, ця властивість мови C# дозволяє змінним базового класу присвоювати об'єкти похідного класу:

```
B b = new D( );
```

І саме ця властивість мови дозволяє передавати об'єкти похідного класу у функції, які очікують об'єкти базового класу. Більш докладно формулювання універсального поліморфізму для мови C# можна подивитися в специфікації мови [10, 12.6.2.3 Run-time evaluation of argument lists] та [10, 10.2.12 Implicit conversions involving type parameters].

Виклад основного матеріалу. У роботі [4] за допомогою ідеї алгебраїчного проектування класу, яка була запропонована в дослідженні [11], проведено формальний аналіз прикладу Р.Мартіна [14] та, в термінах попарного застосування методів класу, показано причину невідповідності типів. Спочатку кожному методу класу ставиться у відповідність його функціональний тип. Це можна зробити за твердженням Б. Мейєра: «Клас – це запрограмований абстрактний тип даних» [15]. Таким чином, кожному методу класу ставиться у взаємооднозначну відповідність функція абстрактного типу даних. Функціональний тип цієї функції і вважатиметься функціональним типом методу. Потім усі функціональні типи методів деякого класу *B* поділяються на чотири групи в такий спосіб. Нехай *domen(B)* означає множину об'єктів класу *B*, а *X* та *Y* – довільні множини (включаючи порожні множини). Тоді маємо наступні групи функціональних типів:

- множина об'єктів класу відсутня у функціональному типі, це статичні методи класу: X → Y;
- множина об'єктів класу присутні тільки з правого боку від функціональної стрілки, це конструктори класу: X → *domen(B)* × *Y;*
- множина об'єктів класу присутня тільки з лівого боку від функціональної стрілки, це методи, які витягують з об'єкта деяку інформацію: (так звані геттери, від слова get): *domen(B)* × X → *Y;*
- множина об'єктів класу присутня з обох сторін функціональної стрілки, це методи, які змінюють стан об'єкта (так звані сеттери, від слова set): *domen(B)* × X → *domen(B)* × *Y.*

Далі вимоги до функцій типів обох класів виписувалися в алгебраїчному вигляді. Тобто за допомогою попарного застосування функцій. Причиною порушення принципу підстановки у Р. Мартіна були несумісні одна з одною вимоги четвертої групи методів. На проблеми, пов'язані з методами четвертого функціонального типу, окремо вказували Б. Мейєр [15] та Л. Карделлі [8]. Наявність таких методів не дозволяє змінювати множину об'єктів похідного класу внаслідок одночасної ко- та контра- варіантності цих методів. Це негайно тягне за собою такий факт, що множина об'єктів класу квадратів (у разі комп'ютерів вона природно кінцева) має збігатися з множиною об'єктів прямокутників. Але, у разі прикладу Р.Мартіна, причиною були інші вимоги до попарного застосування методів. Ці вимоги були несумісні одна з одною для класів квадратів та прямокутників. І такого роду алгебраїчні вимоги до передумов, постумов та інваріантів абстрактного типу даних (як це пропонується в роботах [15; 13] та [11]):

- не можуть бути в принципі перевірені на відповідність поточними компіляторами C++ і C#;
- повністю відповідають аксіомам таких алгебраїчних систем, як група та напівгрупа [16].

Таким чином, використовуючи метод алгебраїчного запису для вимог при проєктуванні свого абстрактного типу даних, можна легко будувати приклади порушення принципу підстановки або передбачати його можливе порушення. Особливо легко це робити для четвертої групи функціональних типів (сеттерів). Оскільки тип будь-якої операції алгебри потрапляє в четверту групу функціональних типів і має несумісні вимоги при переходах від напівгрупи до групи, від кільця до поля, то можна легко проєктувати свої приклади порушення принципу підстановки і небезпеки універсального поліморфізму. До речі, це має бути очевидним для програмістів, які опанували загальне програмування (тобто універсальний поліморфізм) за монографією алгебраїста А.А.Степанова [16].

Детальніше про алгебраїчне проєктування програмного забезпечення (ПЗ) представлено в роботі [7]. Крім того, у звіті [6] представлено один з найпростіших прикладів алгебраїчного проєктування, реалізованого за допомогою аксіом математика 19-го століття Германа Грассмана. У звіті [5] наведено приклад з перевизначенням операції віднімання. Операція віднімання є всюди визначеною у разі групи. Це є наслідком існування нейтрального елемента (identity element), операції обернення (inverse operation) та аксіоми скорочення (cancellation axiom) у групі. Для ілюстрації формулювання алгебраїчних вимог до типів даних, перепишемо аксіоматику групи мовою RAISE Specification Language [17] згідно з [16].

Listing 1. Аксіоматика групи:

```
scheme group =   class
    type
       T
    value
        e     : T         -- identity element
      ,op     : T >< T -> T -- group operation
```


ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

```
      ,inverse : T -> T     -- inverse operation
  axiom
       [associativity]
    all x,y,z : T :-  op (x, op(y,z)) is op (op(x,y), z)
     ,[identity]
       all x : T    :- (op (x, e) is x ) /\ (op (e, x) is  x)
     ,[cancellation]
        all x : T    :-  op (x, inverse(x)) is e
  end
```

З цієї аксіоматики випливає, що через відсутність операція обернення та аксіоми скорочення в напівгрупі, операція віднімання не є всюди визначеною в межах цієї алгебраїчної структури. Зокрема, у прикладі з відніманням [5] порушення принципу підстановки приводило до аварійного завершення роботи додатку, що має бути особливо помітним розробникам ПЗ. У поточній роботі наводиться аналогічний приклад при перевизначені операції ділення, що призводить до явно некоректного результату. Така поведінка, хоча й менш критична порівняно з аварійним завершенням роботи додатку, все ж є досить очевидною та зрозумілою для програмістів. У наступному прикладі, що передбачає застосування симплекс-методу, отриманий результат буде неправильним, але таким, що мало відрізняється від правильного. Для розробників останній приклад некоректної поведінки при реалізації універсального поліморфізму є ознакою найгіршого розвитку подій.

Розглянемо код прикладу**.** Спочатку подамо обидві форми функції *P* для універсального поліморфізму. Функція зобов'язана перетворити результат обчислення в ціле число за допомогою традиційного відкидання дробової частини.

Listing 2. Функція для демонстрації поліморфізму включення:

```
using System;
partial  class Program    {
      public
      static int P ( real one,  real two, real divisor) {
           real out1 = (one * two ) /  divisor;
           return (int) out1;
       }
}
```

Версію для параметричного поліморфізму розроблено в термінах додаткового абстрактного базового класу *number*. Клас *real* є похідним класом від *number* і базовим для класу *integer*. Друга версія функції так само, як і перша, працює з очікуваною помилкою з об'єктами типу *integer*.

Listing 3. Функція для демонстрації параметричного поліморфізму:

```
using System;
partial  class Program {
       static int
       P<T> ( T one,  T two, T divisor) where T : number  {
            number out1 = (one * two ) / divisor;
            return (int) out1;
       }
}
```

Таким чином, згідно з вимогами специфікації, очікується, що якщо функції передати значення 5.0, 3.0 і 2.0 як параметри, то функція *P* повинна повернути число 7 (внаслідок (5.0 * 3.0)/2.0 =7.5). Далі, наведемо часткове визначення класу дійсних чисел, що розробляється *real*.

Listing 4. Методи базового класу:

```
partial class real   {
      public double v;
      public virtual real sum( real r){ return new real (v+r.v);}
      public virtual real dif( real r){ return new real (v-r.v);}
      public virtual real mul( real r){ return new real (v*r.v);}
      public virtual real div( real r){ return new real (v/r.v);}
}
```

Для арифметичних обчислень клас містить чотири віртуальні функції: додавання, віднімання, множення та ділення. Модифікатор *virtual* у методів дозволяє перевизначити їх у похідному класі з можливістю пізнього зв'язування. Самі арифметичні обчислення виконуються за допомогою вбудованих

у мову C# операцій над об'єктами класу *double*, до якого належить поле *v*. Потім, для цих методів вводяться чотири природні позначення, які дають можливість записувати алгоритми з об'єктами класу *real* у звичному інфіксному запису, тобто через операції.

Listing 5. Операції базового класу:

```
partial class real   {
      public static real operator + (real l, real r) { return l.sum(r); }
      public static real operator – (real l, real r) { return l.dif(r); }
      public static real operator * (real l, real r) { return l.mul(r);}
      public static real operator / (real l, real r) { return l.div(r);}
}
```

Після цього функція *P* була побудована як динамічна підвантажувана бібліотека. Для її тестування було створено окремий тестовий варіант, який успішно пройшов перевірку.

Listing 6. Перший тестовий варіант для функції *P*:

```
using System;
using NUnit.Framework;
public partial class demo {
      [Test]
public void integerUsage() {
  Assert.That(
      Program.P(new integer(5), new integer(3), new integer(2)), Is.EqualTo(7)
       );
     }
}
```

Результат виконання цього тестового варіанта:

```
 Test Run Summary
 Overall result: Passed
 Test Count: 1, Passed: 1, Failed: 0, Warnings: 0, Inconclusive: 0, Skipped: 0
```

Як бачимо, тест виконано успішно, отже, можна розпочинати промислову експлуатацію побудованої динамічної бібліотеки.

Зазначимо, що функція *P* є поліморфною за визначенням вище, оскільки задовольняє вимогам кількох типів:

- типу, в термінах якого її розроблено:

domen(real) × domen(real) × domen(real) → domen(int)

де клас *int* – клас цілих чисел, вбудований у мову C#;

- і, згідно з офіційною документацією, типу будь-якого класу похідного від класу *real*, наприклад, запланованого класу *integer:*

domen(integer) × domen(integer) × domen(integer) → domen(int).

Тепер визначаємо похідний клас цілих чисел *integer*, в якому використовується функція *round* (див. нижче округлення до цілих чисел). Вона розроблена відповідно до стандарту [12], який регламентує використання дійсних величин і задає кілька різних напрямків округлення до цілого.

Listing 7. Конструктори похідного класу:

```
using System;
partial  class integer: real {
      public integer (): base(0.0) {}
      public integer (int v)      { this.v = v;}
      public integer (double v)   { this.v = round(v);}
      public integer (string s)   {
       if (!double.TryParse (s, out v))
            this.v = 0.0;
        else
             this.v = round(v);
      }
}
```

Як видно з коду, усі конструктори гарантують появу цілочисельного значення в полі *v*. Переходимо до перевизначеня методів базового класу.

Listing 8. Перевизначені методи похідного класу:

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

```
partial class integer: real {
  public  override  real sum( real r){
    return new integer (this.v + round(r.v));
  }
  public  override  real dif( real r){
    return new integer (this.v – round(r.v));
  }
  public  override real mul( real r){
    return new integer (this.v * round(r.v));
  }
  public  override  real  div( real r){
    return new integer (round( this.v / round(r.v)));
  }
}
```

Методи додавання, віднімання та множення не викликають питань. Вхідний параметр округлюється до цілого та виконується відповідна операція з цілим числом, яке зберігається в полі *v* поточного об'єкта *this*. Результати обчислення всіх трьох функцій повинні залишатися цілочисельними з урахуванням обмежень стандарту [12] щодо подання чисел у змінних типу *double.* Застосування метода ділення *div* може в результаті роботи привести до дробового числа, яке округлятиметься до цілого функцією *round*. Тобто засобами класу *integer*. Однак при розробці цього класу було використано рекомендацію стандарту [12] і для округлення вибрано банківське округлення. У цьому прикладі це значення *ieeeRound.toEven*. Тоді на даних тестового варіанта 6 отримаємо значення 15/2 = 7.5. І воно буде округлено до парного цілого числа, тобто до 8.

Listing 9: Другий тестовий варіант для функції *P:*

```
using System;
using NUnit.Framework;
public partial class demo {
  [Test]
 public void integerUsage() {
   Assert.That(
      Program.P(new integer(5), new integer(3), new integer(2)), Is.EqualTo(7)
   );
  }
}
```

В цьому випадку відкидання дробової частини функції P нічого не змінить. Після цього виконання тестового варіанта закінчиться відмовою:

```
Test Run Summary
Overall result: Failed
Test Count: 2, Passed: 1, Failed: 1, Warnings: 0, Inconclusive: 0, Skipped: 0
Failed Tests – Failures: 1, Errors: 0, Invalid: 0
```

Саме це й було передбачено до початку розробки класу *integer* в роботі [4]. При тестуванні версії функції P для випадку параметричного поліморфізму (дивитися Listing 3) результати виявилися аналогічними випадку з похідним класом. Крім того, тестування показує, що операції базового класу (дивитися Listing 5) в результаті пізнього зв'язування викликають методи похідного класу.

Розглянемо округлення до цілих чисел та .NET. Перетворення довільного дійсного значення до цілої змінної (надалі називатиметься округленням) описане у стандартах [12], починаючи з 1985 року. У кожному зі стандартів задається кілька різних напрямків округлення (rounding direction). Зокрема, у стандарті 2015 року їх п'ять:

- округлення в напрямку до нуля. Виконується відкиданням дробової частини значення з плаваючою комою. Звичне округлення в багатьох мовах програмування, зокрема, у давній мові Фортран-4;
- округлення в напрямку від нуля. Виконується до найближчого цілого числа. Якщо дробова частина значення з плаваючою комою дорівнює 0.5, то додатне значення округлюється до більшого цілого числа, а від'ємне — до меншого цілого;
- округлення в напрямку до найближчого цілого (банківське округлення). У разі, якщо дробова частина значення з плаваючою комою дорівнює 0.5, округлення виконується до найближчого парного цілого числа;
- округлення в напрямку до додатної нескінченності. Значення з плаваючою комою округлюється до більшого найближчого цілого;

• округлення в напрямку до від'ємної нескінченності. Значення з плаваючою комою округлюється до меншого найближчого цілого;

Стандарт не виділяє жодного з напрямків округлення як особливого. Усі вони однаково допустимі. Крім того, жодним чином не обмежується використання кількох напрямків округлення в межах одного коду. Однак, ймовірно, передбачається використання лише одного. Наприклад, у компіляторах C++ вибір напрямку здійснюється шляхом виклику спеціальної функції *fesetround*.

У .NET немає вбудованого механізму для прямого керування напрямком округлення. Проте клас *Math* надає різні методи, поведінка яких відповідає стандарту [12]. Наступний приклад містить метод *round* для ілюстрації усього сказаного.

Listing 10. Округлення до цілих:

```
using System ;
public enum ieeeRound {
  toZero
  , toAway
  , toEven
  , toPositive
  , toNegative
} ;
partial class integer {
 public static ieeeRound direction = ieeeRound . toEven ;
    protected static double round ( double v ) {
  double rc = 0 . 0 ;
  switch ( direction ) {
   case isoRound.toZero :      rc = Math . Truncate ( v ) ;
                    break ;
  case ieeeRound.toAway :          rc = Math . Round ( v , MidpointRounding .
                                AwayFromZero ) ;
  break ;
  case ieeeRound.toEven :          rc = Math . Round ( v , MidpointRounding .
                                  ToEven ) ;
                                    break ;
  case ieeeRound.toPositive :      rc = Math . Ceiling ( v ) ;
   break ;
  case ieeeRound toNegative :     rc = Math . Floor ( v ) ;
  break ;
  default :                                  rc = Math . Truncate ( v ) ;
                                   break ;
   }
  return rc ;
 }
}
```

Де *direction* – це статичне поле деякого класу *integer*.

Висновки. Проведене дослідження демонструє досягнення поставлених цілей:

• розроблено та налагоджено два класи, які можна використовувати для кодування симплекс-метод в термінах базового;

• представлено приклад простої поліморфної функції *P*, яка демонструє порушення принципу підстановки. Це свідчить про те, що відповідні класи задовольняють вимогам несумісних типів;

• показано, що механізм універсального поліморфізму, який реалізований у мові C# не є безпечним;

• такі можливості мови C# як перевизначене (*override*) легко дозволяють розробляти код, який може порушувати принцип підстановки і маскувати цей код під звичні операції;

• часткові класи мови C# дуже зручні при підготовці документації, тому що дозволяють включати в текст документів тільки необхідну частину коду, не перевантажуючи текст непотрібними деталями.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Список використаних джерел:

1. Наслідування – C# | Microsoft Corporation. Microsoft Learn [Текст] (рос.). URL: https://learn.microsoft.com/ru-ru/dotnet/csharp/fundamentals/tutorials/inheritance (дата звернення: 07.07.2025).

2. Поліморфізм – C# | Microsoft Corporation. Microsoft Learn [Текст]. URL: https://learn.microsoft.com/ru-ru/dotnet/csharp/fundamentals/object-oriented/polymorphism (дата звернення: 07.07.2025).

3. Піскунов О. Г. Модульне тестування програмного забезпечення: робоча програма. 2024. URL: https://drive.google.com/file/d/1RBR3ORllP-XjA5idzDBxm-kHyjPr_X5d/view.

4. Піскунов О. Г. Про відмінності між поняттями типу і класу. *Вісник Київського університету. Комп'ютерні науки*. 2015. № 3. С. 106–114. URL: https://www.researchgate.net/publication/344177979.

5. Піскунов О. Г., Мічуда А. М. Перевизначення додавання: небезпечне наслідування у групі цілих чисел. 2023. URL: https://www.researchgate.net/publication/366867037.

6. Піскунов О. Г., Рудик В. І., Петренко І. А. Арифметика Пеано: від специфікації до класу. 2022. URL: https://www.researchgate.net/publication/365979331.

7. Піскунов О. Г., Тупко Н. П., Петренко І. А. Алгебраїчне проєктування програмного забезпечення. *Інформаційні технології та суспільство*. 2024. № 5 (11). С. 50–59.

8. Cardelli L., Abadi M. A theory of objects. Springer, 1996. 396 p. URL: ttps://drive.google.com/open?id=1mShdblP3LnooSSfk8Osh_SAtIc54Jczu.

9. Cardelli L., Wegner P. On understanding types, data abstraction, and polymorphism. *ACM Computing Surveys (CSUR).* 1985. Vol. 17, No. 4. P. 471–523.

10. ECMA International. ECMA-334: C# Language Specification, 7th Edition. Technical Report ECMA-334. December 2023.

11. Haxthausen A. Lecture Notes on The RAISE Development Method. 1999. URL: http://www2.imm.dtu.dk/courses/02263/E20/Files/methodnotes99.pdf.

12. IEEE Standard for Floating-Point Arithmetic. – NY : IEEE Computer Society, 2019. 83 p. URL: https://en.wikipedia.org/wiki/IEEE_754.

13. Liskov B., Wing J. A behavioral notion of subtyping. *ACM Transactions on Programming Languages and Systems (TOPLAS).* 1994. Vol. 16, No. 6. P. 1811–1841.

14. Martin R.C. The Liskov Substitution Principle [Електронний ресурс] // C++ Report. March 1996. URL: https://objectmentor.com/resources/articles/lsp.pdf.

15. Meyer B. Object-oriented Software Construction. – 2-nd edition. Santa Barbara : ISE Inc, USA, 2000. 1284 p.

16. Stepanov A. A., Rose D. E. From Mathematics to Generic Programming. Upper Saddle River, NJ : Addison-Wesley/Pearson, 2014.

17. The RAISE Language Group. The RAISE Specification Language. UNU/IIST, Prentice Hall Europe, Denmark, 1992. URL: https://raisetools.github.io/material/documentation/raise-language.pdf.

Дата надходження статті: 05.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.056.53
DOI https://doi.org/10.32689/maup.it.2025.3.19

Олександр ПОПОВ
член-кореспондент НАН України, доктор технічних наук, професор, директор,
Центр інформаційно-аналітичного та технічного забезпечення моніторингу об'єктів атомної енергетики НАН України,
професор кафедри комп'ютерних інформаційних систем і технологій,
ПрАТ «ВНЗ «Міжрегіональна Академія управління персоналом»,
sasha.popov1982@gmail.com
***ORCID:** 0000-0002-5065-3822*

Роман ДРАГУНЦОВ
аспірант,
Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України,
draguntsow@yahoo.com
***ORCID:** 0000-0002-1781-7530*

КЛЮЧОВІ ВИКЛИКИ ДЛЯ ОПЕРАЦІЙНИХ ЦЕНТРІВ КІБЕРБЕЗПЕКИ В УМОВАХ ПОВНОМАСШТАБНОЇ ВІЙНИ

Анотація. *У статті систематизовано ключові чинники, що визначають ефективність комерційних і державних Security Operation Center (SOC) України в 2022–2025 рр.: деградація енергетичної та телекомунікаційної інфраструктури, окупація й фізична втрата ІТ-активів, кібервійна державного рівня, кадрова криза та перехідна розрізненість регуляторних норм.*

Мета. *Визначити та оцінити вплив обмежень середовища на роботу SOC в умовах сучасної повномасштабної війни, визначити ключові особливості загроз кібербезпеки в даних умовах.*

Методологія. *У дослідженні застосовано комбінований підхід, що поєднує огляд та аналіз існуючих досліджень і статистичних даних, міжнародних та українських нормативних актів у сфері кібербезпеки, і емпіричну перевірку на базі ретроспективного порівняльного аналізу ключових операційних метрик SOC у референтних інфраструктурах. Вибірка охоплювала періоди відносної стабільності та масових відключень електропостачання, що дозволило ідентифікувати кореляцію між зовнішніми чинниками середовища та деградацією показників ефективності.*

Наукова новизна. *У роботі системно ідентифіковано та структуровано специфічні виклики функціонуванню SOC в умовах сучасної повномасштабної війни: деградація енергетичної та телекомунікаційної інфраструктури, фізична окупація й втрата ІТ-активів, кадрова криза, кібервійна державного рівня та розрізненість нормативної бази. Продемонстровано вплив цих факторів для ключових метрик ефективності SOC.*

Висновки. *Ефективність підрозділу кібербезпеки в умовах повномасштабної війни визначається стійкістю до зовнішніх впливів та наявністю відповідних контролів. Практично це вимагає: резервування енергоживлення і зв'язку на критичних вузлах; буферизації телеметрії та відкладеної кореляції; заздалегідь опрацьованих сценаріїв ізоляції сегментів і ключів; гібридної організації праці із мінімізацією навантаження аналітиків; уніфікації контролів за «профілями безпеки» з прозорим відображенням на хмарні сервіси. Отримані результати задають пріоритети для розподілу ресурсів SOC і подальшого тестування контрзаходів.*

Ключові слова: *Security Operation Center, кібервійна, спостережність, окупація ІТ-активів, кадрова криза.*

Oleksandr POPOV, Roman DRAHUNTSOV. KEY CHALLENGES FOR SECURITY OPERATIONS CENTERS IN THE CONTEXT OF FULL-SCALE WAR

Abstract. *The article systematizes the key factors defining the effectiveness of commercial and governmental Security Operation Centers (SOC) in Ukraine during 2022–2025: degradation of energy and telecommunications infrastructure, occupation and physical loss of IT assets, state-level cyber warfare, workforce crisis, and transitional inconsistency of regulatory norms.*

Purpose. *To identify and assess the impact of environmental constraints on SOC operations under the conditions of a modern full-scale war, and to define the specific characteristics of cybersecurity threats in this context.*

Methodology. *The research combines a review of scientific and analytical sources and Ukrainian regulatory acts with a retrospective comparative analysis of SOC metrics across reference infrastructures.*

Scientific novelty. *The study identifies critical challenges specific to cybersecurity units operating in the context of full-scale warfare and evaluates their influence on SOC performance indicators.*

Conclusion. *The effectiveness of a cybersecurity unit under wartime conditions is determined by its resilience to external disruptions and the availability of appropriate controls. In practice, this requires: redundancy of power supply and communications at critical nodes; buffering of telemetry and deferred correlation; pre-defined scenarios for segment and key isolation; hybrid work organization with minimized analyst workload; and unification of control through "security profiles" with transparent mapping to cloud services. The obtained results define priorities for SOC resource allocation and further testing of countermeasures.*

Key words: *Security Operation Center, cyberwar, observability, occupation of IT-assets, staffing problems.*

Постановка проблеми. В умовах повномасштабної війни ландшафт кібербезпеки для України та світу суттєво змінюється. В період 2022–2025 рр. український кіберпростір став фактично одним із найбільш атакованих у світі – зокрема, у 2024 р. командою CERT-UA зафіксовано 4315 кібератак, що на 70 % перевищує показник попереднього року [1]. Близько 75 % усіх кібератак здійснених державними структурами російської федерації у 2022–2023 рр. були націлені саме на Україну [1]. Попри детальний розгляд проблематики кібервійни – від опису окремих кіберінцидентів до принципів ведення бойових дій в кіберпросторі – залишається менше висвітленою проблематика внутрішніх викликів забезпечення кібербезпеки: підтримка роботи операційних центрів кібербезпеки (тут і дали, Security Operation Center – SOC) комерційних компаній та державних установ в умовах кінетичних атак, відключень електропостачання, окупації територій та інших військових факторів. Актуальність дослідження зумовлена необхідністю ідентифікації конкретних проблем у функціонуванні SOC в умовах повномасштабної війни для визначення відповідних засобів для вирішення цієї проблематики, визначення методів розбудови SOC, що будуть ефективними з урахуванням реалій воєнного часу. Повномасштабні війни 21-го століття продемонстрували, що кіберскладова є надзвичайно важливим елементом національної обороноздатності [2], тому виявлення проблем та перешкод у функціонуванні SOC під час бойових дій є необхідним фактором забезпечення національної стійкості. В статті розглянуто ключові проблеми кібербезпеки, з якими стикнулися українські комерційні та державні SOC у період 2022–2025 рр.

Аналіз останніх досліджень та публікацій. У воєнний період питання кібербезпеки України висвітлюються у звітах, наукових дослідженнях та аналітичних оглядах: здебільшого відзначається кіберстійкість української держави, роль інноваційних рішень та залучення приватного сектору у протидії атакам, підкреслюється перехід після вторгнення 2022 р. від реактивної до проактивної моделі, інтегрованої у воєнну стратегію країни [3].

В розрізі проблематики розбудови SOC розглядається кадровий дефіцит, фрагментарність нормативної бази та недостатня координація між державним і приватним секторами у сфері кібербезпеки – відзначається необхідність інтеграції приватних та державних SOC у єдину інфраструктуру, підсилену автоматизацією, зокрема, на основі ШІ, задля зменшення часу реагування на інциденти [4]. В розглянутих роботах такі висновки зроблені на основі аналізу досвіду кібератак 2014–2023 рр. Оцінка кадрового дефіциту в домені кібербезпеки носить характер критичної проблеми в умовах воєнного стану [5, 6]. При цьому ж, дана тенденція також має і глобальний характер: глобальний розрив кадрів у сфері кібербезпеки сягнув 4 млн спеціалістів, а 59 % команд кібербезпеки у світі недоукомплектовані [6]. Криза в українських умовах посилюється змінами в структурі зайнятості і міграцією населення. Статистично ці факти відображаються в опитуванні Європейської Бізнес-Асоціації, які відображають, що 67 % українських компаній відчувають гострий дефіцит персоналу, дві третини роботодавців називають повномасштабну війну основною причиною даної проблеми.

Значна увага у літературі та наявних дослідженнях приділяється феномену кібервійни, який знаходиться поза рамок даного дослідження. Відзначається рекордний масштаб залучення Advanced Persistent Threat (APT-груп) державного рівня та кібер-криміналітету для досягнення військових цілей у кіберпросторі [1–7]. Російські хакерські угрупування використовують Україну як полігон для відпрацювання атак. Близько 75 % їхньої активності припадає на українські об'єкти, причому, схожі тактики та техніки застосовуються після цього глобально [1].

Існуюча література в основному не висвітлює впливів фізичного руйнування інфраструктури на функціонування підрозділів кібербезпеки, порушення доступності та нормального функціонування цивільної інфраструктури, зокрема, енергетичної та телекомунікаційної. Хоча дана проблематика є тісно пов'язаною з питаннями забезпечення життєстійкості критичної інфраструктури, специфіка даного питання розглядається лише в окремих дослідженнях [8]. Окрім питань нестачі персоналу внаслідок кадрової кризи обмежено розглядається питання географічного розподілення персоналу, необхідного часу адаптації кадрів та впливу морально-психологічного стану на ефективність роботи фахівців. В даній статті проводиться аналіз статистичних даних, що підтверджують кожен з факторів проблематики, визначається вплив на функціонування SOC під час повномасштабної війни.

Метою дослідження є визначення та оцінка впливу обмежень середовища на роботу SOC, визначення ключових особливостей загроз кібербезпеки в умовах сучасної повномасштабної війни для подальшого напрацювання методологій та підходів до розбудови SOC для приватних та державних підприємств у військовий час.

Виклад основного матеріалу дослідження. ***Руйнування енергетичної та телекомунікаційної інфраструктури.*** Фізичне знищення цивільної інфраструктури – одна з перших і найочевидніших проблем, з якими зіткнулися кібербезпекові підрозділи. Російські обстріли енергосистеми

спричинили масові та тривалі відключення електропостачання, що ускладнило роботу дата-центрів, мережевого обладнання, засобів моніторингу та створило навантаження на системи резервування. Внаслідок кінетичних атак на енергетику України наприкінці 2022 р. – на початку 2023 р. значна частина держави регулярно залишалася без електропостачання на періоди від декількох годин до декількох діб. Для забезпечення неперервності роботи SOC та IT-систем підприємства розгортають резервні джерела живлення і канали зв'язку, а також забезпечують моніторинг доступності. Як приклад, найбільший оператор телекомунікаційних послуг Kyivstar станом на 2024 р. розширив свою інфраструктуру резервування енергоживлення 2322 дизель-генераторами та понад 115 тисячами резервних батарей на базових станціях мобільного зв'язку по усій Україні [9]. Втрата електроживлення на значній частині території держави у розподілених IT-системах призводить до впливів на команди кібербезпеки у формі втрати каналів зв'язку для віддалених локацій інфраструктури та розривах у комунікації з сенсорами безпеки [10].

Для команд SOC пошкодження телекомунікаційної та енергетичної інфраструктури призводить до фрагментованості моніторингу. Відключення призводить до ізоляції окремих сегментів інфраструктури мережі на фізичному рівні, SOC втрачають видимість подій у цих сегментах. Аналіз інцидентів затримується, оскільки журнали подій недоступні в реальному часі, а сенсори (наприклад, мережеві Intrusion Detection System (IDS)) не мають зв'язку з центральними консолями керування. Зниження пропускної здатності мереж через аварійні схеми маршрутизації та супутникові канали призводить до збільшення часу реагування на інциденти. Ключовими контрзаходами є резервування енергоживлення критично важливих об'єктів на період визначений доцільним в рамках оцінки ризиків [11] та резервування каналів зв'язку за рахунок, зокрема, супутникових каналів [9]. Обов'язковим контрзаходом є впровадження детектуючих контролів безпеки для виявлення події зникнення електроживлення, вимкнення каналів зв'язку, зникнення зв'язності мережі. Також SOC повинен володіти знанням інфраструктури в розрізі динамічності та вразливості об'єктів до вимкнення електропостачання. Український досвід показав ефективність масового резервування джерел живлення, каналів зв'язку за рахунок супутникових терміналів, міграції сервісів у хмарні дата-центри та розподіл VPN-мереж. Порушення цілісності цивільної інфраструктури спричиняє низку проблем для SOC від фізичних перебоїв у роботі центрів обробки даних до деградації якості моніторингу та збільшення часу на реагування.

В рамках аналізу вибірки діючих SOC, досвід яких враховувався при написанні даної статті, було проаналізовано ключові ідентифікатори деградації параметрів роботи SOC в межах дії масових впливів відключення електропостачання, а саме: середній час реагування на інциденти до закриття (MTTR, Mean Time To Response), середній час ідентифікації інциденту (Mean Time To Detect) та середній відсоток доступності систем технологічної платформи SOC. Були розглянуті періоди грудень 2023 р. – лютий 2024 р. (Період 1, період відносної стабільності цивільної інфраструктури), червень – серпень 2024 р. (Період 2, період масових відключень електропостачання), грудень 2024 р. – лютий 2025 р. (Період 3, період відносної стабільності цивільної інфраструктури). Так, було відмічено збільшення MTTR в Період 2 у порівнянні з Періодом 1 на 24 % та в порівняння з Періодом 3 – на 26 %. Збільшення MTTD в Період 2 в порівнянні з Періодом 1 склало 425 %, в порівнянні з Періодом 3 – 412 %. Відсоток доступності ТП SOC для усіх інфраструктурних компонентів для Періоду 2 зменшився приблизно на 7 % в порівнянні з Періодами 1 та 3. Очевидна суттєва деградація ефективності виявлення інцидентів та інших показників кібербезпеки саме в період масових відключень електропостачання, що пояснюється саме кореляцією з періодами масових відключень електропостачання. Дана проблематика зумовлена саме аспектами середовища та доступності відповідних інфраструктурних сервісів, та, враховуючи взяття до уваги показників декількох різних SOC, менше стосується внутрішніх процесів окремих підрозділів кібербезпеки.

Окупація та втрата об'єктів IT-інфраструктури. Специфічною загрозою для кібербезпеки в рамках повномасштабної війни з динамічною лінією бойового зіткнення стає фактор окупації територій та об'єктів IT-інфраструктури. На відміну від кіберінцидентів мирного часу, коли об'єкти (дата-центри, вузли мережі, окремі кінцеві точки) переважно залишаються під фізичним контролем власників, під час війни певна частина IT-активів опиняється захопленою супротивником в непошкодженому стані або фізично знищується бойовими діями. Однією з загроз, які з'являються в даному сценарії, є захоплення фізичного контролю над фрагментом мережі для подальшого контролю та просування в мережевій інфраструктурі. Зокрема, російські сили на окупованих територіях України цілеспрямовано захоплювали вузли зв'язку та центри обробки даних з подальшим підключенням їх до російських мереж – подібна тактика фіксувалась у діях російських окупаційних військ починаючи з 2014 р. на території Донецької та Луганської областей та в Криму [7]. Після лютого 2022 р. ці ж методи застосовувалися і на новоокупованих територіях (Херсонщина, Запорізька область тощо), що призводило як до

інформаційної ізоляції окупованої території, так і до порушення цілісності мережевої інфраструктури не окупованої України. Протягом 2022 р. окупаційні війська здійснювали примусове переведення трафіку регіональних провайдерів через російські мережі (Khersontelecom – Miranda Media/Rostelecom, після первинного відключення і «переналаштування» каналів наприкінці травня 2022 р.), що фактично встановлювало цензуру та повний технічний контроль трафіку [7].

Окупація фізичних компонентів інфраструктури створює для SOC втрату видимості та системну кризу довіри: скомпрометованими вважаються всі канали зв'язку, ідентичності вузлів, журнали подій та засоби керування. Також регулярно може відбуватись часткове знищення або від'єднання обладнання українськими провайдерами, щоб не допустити його використання ворогом, та прискорене винесення державних реєстрів і критичних даних у хмарні середовища за межі регіонів ризику [12–15]. Модель загроз при окупації ІТ-активів представлено в (табл. 1).

Таблиця 1

Модель загроз при захопленні ІТ-активів

Загроза	Опис	Техніка Mitre Att&ck [16]
Компрометація мережевої маршрутизації	Примусовий BGP/AS-рерутинг через підконтрольні оператори створює стійкі умови для Adversary-in-the-Middle, DPI (Deep Packet Inspection), ін'єкції контенту та SSL-перехоплення, навіть без змін у кінцевих системах.	TA0001 Initial Access через T1557 Adversary-in-the-Middle
Фізичне втручання в обладнання	Доступ до фізичного майданчика дозволяє встановлювати інлайн-тапи, змінювати конфігурації L2/L3, підмінювати оптичні канали, переносити ключі з мережевих пристроїв, змінювати прошивки (ризики до рівня UEFI/Boot-ROM), активувати приховані облікові записи на BMC (iLO/iDRAC).	TA0001 Initial Access через T1557 Adversary-in-the-Middle
Викрадення секретів	Отримання супротивником доступу до резервних носіїв, вузлів з HSM, токенів доступу у системах, що можуть застосовуватись в іншій частині інфраструктури, конфігурацій, доступів до out-of-band (OOB).	TA0006 Credential Access – T1552 Credentials in Files TA0040 Impact – T1489 Service Stop/T1490 Inhibit System Recovery TA0008 Lateral Movement – T1021 Remote Services
Технічна деградація спостережності	Відключення або фальсифікація журналів, підміна NTP, ін'єкція фальшивих подій і телеметрії, перевидача сертифікатів підконтрольним CA (Certification Authority).	TA0005 Defense Evasion – T1562 Impair Defenses
Примусове «переведення» абонентів на окупаційні мережі	Розповсюдження SIM, що компрометує мережеві й телефонні канали.	TA0001 Initial Access через T1557 Adversary-in-the-Middle

Вплив на функціонування SOC спричиняє колапс домену довіри – усі активи в зоні окупації вважаються недовіреними незалежно від попередньої автентикації та MAC/IP, сертифікати, TPM чи стандартні IoC без постійного контролю фізичного стану локації втрачають необхідний рівень довіри. Дана проблема особливо актуальна в умовах динамічної лінії фронту російсько-української війни періоду лютого – червня 2022 р. Весь трафік у регіонах, в яких ведуться активні маневрені бойові дії, підлягає обробці за принципом treat-as-compromised. Також одним з найважливіших ризиків даної ситуації є ризик вторинної компрометації центру, тобто зміщення супротивника з окупованої частини ІТ-інфраструктури до центральної та довіреної частини. Site-to-site тунелі, SD-WAN мережі, об'єднанні системи аутентифікації, Single-Sign On, агенти моніторингу та реплікації резервних копій можуть стати шляхом зміщення супротивника всередину мережі.

Кадрова криза. Кадровий склад в умовах повномасштабної війни зазнає одного з найбільших негативних впливів. Українська ІТ-галузь і сфера кібербезпеки входили у повномасштабну війну у 2022 р. з певним кадровим дефіцитом, характерним для галузі у всьому світі, але військові дії суттєво загострили проблему. Зміна в структурі зайнятості населення, масова евакуація та еміграція фахівців за кордон, а також прямі втрати серед військовослужбовців-фахівців призвели до того, що багато центрів кібербезпеки втратили частину досвідчених співробітників. Станом на 2023–2024 рр. 67–74 % українських компаній повідомляли про нестачу кадрів зумовлену факторами повномасштабної війни. При цьому, статистично майже 60 % команд кібербезпеки у світі на момент 2025 р. недоукомплектовані [6]. Особливо відчутною є дефіцит висококваліфікованих кадрів (аналітиків рівня Senior, архітекторів безпеки тощо), пошук і підготовка яких потребують тривалого часу. Очевидно, що

для інших сучасних масштабних військових конфліктів проблематика впливу на доступність кадрового ресурсу буде актуальною [17].

Кадрова криза проявляється двома ключовими шляхами. Перший – прямий, тобто нестача спеціалістів для укомплектування команд SOC. На прикладі трьох референтних SOC було визначено проблематику хронічного дефіциту фахівців нічних змін та певний дефіцит фахівців другої лінії кіберзахисту. Середні показники комплектації команд відрізняються від довоєнних на 25 %. Другий аспект – розпорошення персоналу географічно. Через бойові дії значна частина IT-спеціалістів релокувалася у більш безпечні регіони держави або за кордон – статистично 20% українських IT-фахівців у 2023 р. працювали віддалено за межами України [18]. З урахуванням даних факторів суттєва частина SOC України переходять у віддалений або гібридний режими роботи, що є вимушеним та достатньо складним заходом з урахуванням специфічних вимог операційної безпеки. Окрім суто безпекової проблематики розподіленої роботи команд кібербезпеки є і операційна проблематика. Як вже розглядалось у одному з попередніх розділів, руйнування цивільної інфраструктури, такої як енергетична та телекомунікаційна, призводить до втрат зв'язку і розривів IT-систем. Для центрів обробки даних дана проблематика вирішується встановленням додаткових резервних джерел живлення та зв'язку, в той час як для індивідуальних фахівців подібні заходи вдома можуть бути недоступними – аналітик у віддаленому режимі може в будь-який момент втратити зв'язок з робочою інфраструктурою внаслідок зникнення електропостачання чи каналу зв'язку. Резервування на достатньому рівні робочих місць фахівців кібербезпеки в умовах віддаленої, розподіленої роботи неможливе в достатній мірі внаслідок їх високої вартості та необхідності великої кількості. В умовах відсутності зв'язку SOC залишається без відповідних фахівців, які, в свою чергу, можуть бути єдиними доступними на зміні на даний момент часу, зважаючи на проблему кадрової кризи. Другою проблемою є забезпечення безпеки доступу: забезпечення захищених каналів (Virtual Private Network – VPN, Virtual Desktop Infrastructure – VDI) для підключення співробітників з різних мереж зі збереженим рівнем довіри, що збільшує навантаження на IT-інфраструктуру [19].

Додатковою кадровою проблемою в розглянутих умовах є зниження оперативності обміну знаннями та навчання нових фахівців: У віддаленому режимі цей процес ускладнюється, також ускладнюється і питання введення в робочий режим нових співробітників. Навчання нових фахівців, знайомство їх з робочими процесами, колективом, ключовими принципами функціонування SOC у віддаленому режимі ускладнюється, збільшується тривалість – на прикладі розглянутих в дослідженні референсних SOC навчання та загальний онбордінг фахівців у повністю віддаленому режимі потребував в середньому на 7 % більше часу.

Ще один специфічний виклик кадрового характеру – психологічна стійкість і мотивація персоналу SOC. Робота аналітика кібербезпеки завжди містить високий рівень стресу, але під час військових дій рівень стресу співробітників суттєво зростає. До факторів стресу належать обстріли, зокрема й нічні, руйнування цивільної інфраструктури, острах за розвиток подій на фронті та близьких залучених до оборони, а також невизначеність майбутнього. Очевидно, що в умовах стресового та часто понаднормового реагування на інциденти рівень психологічної готовності команд кібербезпеки знижується, з ним знижується рівень мотивації та концентрації, що має очевидний вплив на ефективність виявлення та реагування на загрози. Впровадження ротацій фахівців, додаткових вихідних є дуже обмеженим за ефективністю заходом, однак фактичний вплив подібних підходів потребує додаткових досліджень. Очевидно, що надання додаткових вихідних в умовах кадрової кризи є не ефективним рішенням з огляду на хронічну нестачу персоналу.

Фактор кібервійни. Повномасштабне вторгнення російської федерації в Україну супроводжується інтенсивною кібервійною, у якій діє значна частина хакерських груп обох держав. З боку рф більшість таких загроз – це підрозділи спецслужб (так звані Advanced Persistent Threat, APT). Хоча кібератаки на українські держструктури відбувалися і раніше (з 2014 р. фіксувалися операції Sandworm, Armageddon/Gamaredon та інші), у 2022–2023 рр. їх масштаб зріс на порядки. З початку вторгнення лише державними регуляторами та незалежними компаніями з кібербезпеки зафіксовано кратне зростання кількості інцидентів різного рівня складності. Станом на 2025 р. Україна посідає 2-е місце у світі серед країн, що найбільше стикаються з кіберзлочинністю [1]. Для будь-якого SOC це означає істотне збільшення навантаження: обсяг інцидентів, що фіксуються SOC, та які потребують розслідування та реагування, зросла приблизно в 2,5 рази, що було визначено на прикладі референсних інфраструктур SOC.

Аналогічно до українського досвіду Ізраїль після початку активної фази війни з Хамас зазнав різкого збільшення тиску на інфраструктуру кіберзахисту. Після нападу Хамас 7 жовтня 2023 р. кількість звернень на гарячу лінію кібербезпеки державних кіберцентрів зросла в 10

разів – з близько 50 повідомлень до понад 500 на день, а число активних APT-груп, що атакували країну, подвоїлося [20]. Основними агентами загроз, що оперують у кібервійні проти Ізраїлю, є іранські державні структури та вільні угрупування, що корелює з українським досвідом.

Діяльність SOC ускладнюється комбінованими атаками, що синхронізуються з військовими діями: наприклад, 24 лютого 2022 р. паралельно з кінетичними обстрілами було виведено з ладу частину інфраструктури супутникової мережі ViaSat, атаковано інфраструктуру GigaTrans та на 15 год уражено частину мережі Укртелекому. Такі атаки здійснюються із застосуванням спеціалізованого шкідливого ПЗ (Industroyer2, CaddyWiper, Pterodo), а частка атак на критичну інфраструктуру України зросла з 20 % у 2021 р. до 40 % у 2022 р. Навантаження на функціонування SOC зростає і зумовлюється також фактором застосування супротивником постійно змінюваних та еволюціонуючих інструментів – майже кожен з використованих російськими APT-групами інструментів зазнає неперервного розвитку та еволюції. Це змушує команди SOC впроваджувати більше проактивних практик виявлення інцидентів, налагоджувати обмін розвідданими з розвідувальними джерелами та впроваджувати дані практики в операційні процеси. Очевидно, що успішне впровадження даних заходів потребує розширення виділених на функціонування SOC ресурсів, як людських так і фінансових, які є також значно обмеженими у військовий час.

Розрізненість нормативної бази. У березні 2022 р. Кабінет Міністрів України дозволив державним реєстрам та відповідним ІК-системам розміщуватися у хмарних/ЦОД (Центрах Обробки Даних) поза межами України на період дії воєнного стану з обов'язком припинити таке розміщення протягом шести місяців після його завершення. Це рішення легалізувало «цифрову евакуацію», але водночас поставило питання узгодженості з чинними галузевими та загальними вимогами захисту інформації. Паралельно держава формалізувала ринок хмар для публічного сектору: Закон України «Про хмарні послуги» [35] запровадив державний перелік провайдерів та регуляторний нагляд Держспецзв'язку. У 2025 р. регулятор актуалізував і деталізував порядок включення провайдерів до переліку, що створює основу для контрольованого використання хмар у державному секторі [21, 22] Системоутворюючими рамками стали також: Положення про організаційно-технічну модель кіберзахисту [23], загальні вимоги до кіберзахисту ОКІ (Об'єктів Критичної Інфраструктури) [26], Закон України «Про критичну інфраструктуру» (1882-IX, 2021 р.), Порядок реагування на події у кіберпросторі [30]. Сукупно вони задали «скелет» державної моделі кіберзахисту та координації реагування [23].

До 2024 р. основним операційним інструментом в державному регулюванні функціонування SOC були Нормативні Документи систем Технічного Захисту Інформації (НД ТЗІ) та атестація/експертиза Комплексних Систем Захисту Інформації (КСЗІ); у 2024–2025 рр. держава перейшла до ризик-орієнтованих «профілів безпеки» і запустила нову модель підтвердження відповідності: від експерименту з декларування [25] – до постійної моделі «авторизації з безпеки» та порядку профілювання [32]. Перехідний період спричинив накладання старих і нових процедур, різночитання щодо достатності контролів і дублювання оцінок [24; 25]. Загальнодержавні вимоги [23; 26; 30] співіснують із жорсткішими галузевими нормами, наприклад, Постанова НБУ № 95 [36], що веде до різних трактувань цільового рівня захисту, строків повідомлення про інциденти та змісту звітності. Для багатосекторних груп це породжує паралельні комплаєнс-процеси [26; 27]. Дозвіл Кабінету Міністрів України № 263 [33] на екстрене розміщення за кордоном змінив традиційну модель побудови КСЗІ, орієнтовану на національні процедури експертизи та локальне середовище. До появи оновлених профілів безпеки та порядку авторизації організаціям бракувало уніфікованих правил зіставлення «українських» вимог із практикою провайдерів Європейського Союзу, що ускладнювало підтвердження відповідності для гібридних/хмарних архітектур. Базовий Закон №2297-VI [28] залишається чинним (редакція 2025 р.), а оновлена редакція № 8153 [34], що гармонізує режим із GDPR (General Data Protection Regulation) (ухвалена у І читанні в листопаді 2024 р.), ще не набула чинності. Це створює невизначеність щодо підстав і гарантій транскордонної обробки даних, ролей контролера/процесора та санкційного режиму [28; 29].

Практично у функціонуванні SOC розрізненість нормативної бази має три ключові фактори впливу – задвоєння норм відповідності, нерівномірність процедур реагування та неоднозначності у трактовці вимог ризик-менеджменту на міждержавному рівні.

Організації, які одночасно є суб'єктами ОКІІ і піднаглядними галузевому регулятору, змушені підтримувати різні набори контролів, звітності та аудиторських процедур. Це підвищує операційні витрати та ризик виникнення явища формальної відповідності регуляторним вимогам [26; 27]. Загальний порядок реагування [30] і методичні рекомендації Держспецзв'язку задають єдину логіку етапів реагування на інциденти, але галузеві акти додають власні терміни та канали комунікації, що потребує узгоджених, багатовекторних планів оповіщення (CERT-UA, регулятор, клієнт) [30].

За умов розміщення даних/сервісів у ЄС вимагається явне відображення контролів НД ТЗІ/профілів безпеки на контрольні набори провайдерів і внутрішні політики провайдера, що є нетривіальною задачею [24; 28].

Проблематика розрізненості нормативної бази хоча і є актуальним викликом, натомість має тенденцію до вирішення протягом 2023–2025 рр. Зокрема, постановою КМУ № 712 [32] затверджено порядок розробки/затвердження профілів безпеки інформації та порядок авторизації з безпеки, Держспецзв'язку затвердив порядок моніторингу їх реалізації (наказ № 160, березень 2025 р.). Це уніфікує підходи до підтвердження відповідності й дозволяє будувати КСЗІ як ризик-орієнтовані конфігурації, сумісні з хмарними шаблонами [31]. НД ТЗІ зазнали оновлення – НД ТЗІ 3.6-006-24 і наказ Держспецзв'язку № 54 (січень 2025 р.) з «Базовими заходами» та методичними рекомендаціями фактично здійснюють відображення стандарту на сучасні домени [24]. Проблематика сприйняття хмарних інфраструктур в правовому полі вирішується в рамках закону № 2075-IX [35] – утворюються уніфіковані критерії допуску хмар до держсектору, що знижує правову невизначеність під час міграції інфраструктури [21, 22]. Прийняття нової редакції закону про персональні дані у другому читанні [34] має усунути недоліки регуляції транскордонності, прав суб'єктів та санкціях, що є важливим для хмарних операцій. У 2025 р. Національний банк України оновив Постанову № 95 [36], підвищивши вимоги до інформаційної безпеки у банківському секторі, Міністерство енергетики України готує секторальні вимоги до паливно-енергетичного комплексу з урахуванням державної моделі кіберзахисту. Таким чином, галузеві регуляторні вимоги наближуються до єдиного «ядра» контролів, що однозначно зменшує рівень розрізненості нормативної документації.

Висновки. Виконане дослідження показало, що на ефективність SOC в умовах сучасної повномасштабної війни впливає у значній мірі не лише внутрішня зрілість процесів, а й екзогенні обмеження середовища – насамперед деградація енергетичної й телекомунікаційної інфраструктури, втрата фізичного контролю над активами та зростання інтенсивності загроз. Емпіричні дані показали суттєве погіршення ключових метрик у періоди масових відключень електропостачання: приріст часу виявлення та реагування і зниження доступності технологічної платформи SOC.

Окремим важливим чинником виступає окупація й фізичне втручання в ІТ-активи, яке руйнує домен довіри й підвищує ймовірність вторинної компрометації через легітимні канали зв'язку. За цих умов раціональною стратегією є «treat-as-compromised» для прикордонних і близьких до лінії зіткнення сегментів, примусова сегментація та Zero Trust на міжсайтових з'єднаннях, попередньо авторизовані сценарії ізоляції, а також затримкостійкі ланцюги спостережності з буферизацією телеметрії та хмарними точками відновлення. Паралельно кадрова криза і географічне розпорошення персоналу зміщують оптимальну модель у бік гібридної організації праці з VDI/JIT-доступом, що потребує стандартизації онбордингу, підтримки психологічної стійкості та системного зменшення когнітивного навантаження для аналітиків.

Регуляторні зміни 2023–2025 рр. формують тренд до уніфікації вимог і ризик-орієнтованого підтвердження відповідності, однак перехідний період зберігає неоднозначність регуляторних вимог. Практичним наслідком для SOC є необхідність підтримувати «єдине ядро» контролів із чітким відображенням профілів безпеки на хмарні шаблони провайдерів і міжсекторально узгоджені процедури інцидент-менеджменту та звітності.

Життєздатний у сучасних воєнних умовах SOC – це ризик-орієнтована система, що поєднує мінімально необхідну спостережність за умов погіршеної видимості інфраструктури, превентивну сегментацію довіри, операційну автоматизацію та комплаєнс, гармонізований із хмарними архітектурами. Подальша робота ведеться над кількісною оцінкою ефективності контрзаходів під час блекаутів, верифікації метрик «мінімально життєздатної спостережності», порівняльному аналізі з іншими країнами, що діють в умовах кібервійни, а також на відстеженні впливу завершення регуляторного переходу на операційні витрати й якість реагування.

Список використаних джерел:

1. Microsoft Digital Defense Report 2024. URL: https://cdndynmedia1.microsoft.com/is/content/microsoftcorp/microsoft/final/en-us/microsoft-brand/documents/Microsoft%20Digital%20Defense%20Report%202024%20%281%29.pdf (дата звернення: 31.08.2025).

2. Cyber Digest. Огляд ключових подій у світі кібербезпеки за жовтень 2022. URL: https://www.rnbo.gov.ua/files/2022/NKCK/Cyber%20digest_Oct.pdf (дата звернення: 31.08.2025).

3. GMF – Ukraine's Cyber Defense: URL: https://www.gmfus.org/sites/default/files/2023-12/Kvartsiana%20-%20Ukraine%20Cyber%20-%20Report.pdf (дата звернення: 31.08.2025).

4. Prokopovych-Tkachenko D., Zvieriev V., Kozachenko I. Integration of Security Operations Centers (SOC) into Ukraine's national security system. *STATE SECURITY.* 2025. T. 1, № 5. С. 106–114. URL: https://doi.org/10.33405/2786-8613/2025/1/5/336736 (дата звернення: 01.09.2025).

5. Дефіцит кіберспеціалістів. Чому українські фахівці на вагу золота. URL: https://speka.media/deficit-kiberspecialistiv-comu-ukrayinski-faxivci-cinuyutsya-na-vagu-zolota-vzjr69 (дата звернення: 31.08.2025).

6. Ukraine's cyber specialists disrupt Russian industry backbone. URL: https://defence.nridigital.com/global_defence_technology_aug24/latest-news-ukraine_war_dominant_in_cyber_operations (дата звернення: 31.08.2025).

7. Локот Т. Russia's Networked Authoritarianism in Ukraine's Occupied Territories during the Full-Scale Invasion: Control and Resilience: URL: https://ppr.lse.ac.uk/articles/10.31389/lseppr.85 (дата звернення: 31.08.2025).

8. Drahuntsov R., Zubok V. Modeling of Cyber Threats Related To Massive Power Outages and Summary of Potential Countermeasures. Èlektronnoe *modelirovanie.* 2023. T. 45, № 3. С. 116–128. URL: https://doi.org/10.15407/emodel.45.03.116 (дата звернення: 01.09.2025).

9. Lipscombe P. Ukrainian telco Kyivstar to deploy more generators in fight against blackouts : URL: https://www.datacenterdynamics.com/en/news/ukrainian-telco-kyivstar-to-deploy-more-generators-in-fight-against-blackouts (дата звернення: 31.08.2025).

10. Зубок В. Ю., Драгунцов Р. І. Особливості розслідування та реагування на інциденти кібербезпеки в умовах масових відключень електропостачання. Матеріали V науково-практичної конференції «Безпека енергетики в епоху цифрової трансформації», м. Київ, 22 листопада 2023 р.

11. Драгунцов Р. І. Алгоритм управління ризиками кібербезпеки в умовах знищення енергетичної інфраструктури. Науково-практична конференція «Кібербезпека енергетики», м. Київ, 28 травня 2025 р.

12. Wired: Ukraine–Russia internet takeover: URL: https://www.wired.com/story/ukraine-russia-internet-takeover (дата звернення: 31.08.2025).

13. Russia descends iron curtain over occupied land, cuts people off in internet: URL: https://kyivindependent.com/russia-descends-iron-curtain-over-occupied-land-cuts-people-off-internet/ (дата звернення: 31.08.2025).

14. Safeguarding Ukraine's data to preserve its present and build its future (Amazon): URL: https://www.aboutamazon.com/news/aws/safeguarding-ukraines-data-to-preserve-its-present-and-build-its-future (дата звернення: 31.08.2025).

15. One year of war in Ukraine (Cloudflare blog): URL: https://blog.cloudflare.com/one-year-of-war-in-ukraine/ (дата звернення: 31.08.2025).

16. MITRE ATT&CK : URL: https://attack.mitre.org/ (дата звернення: 31.08.2025).

17. Кадрова криза в Україні: брак спеціалістів: URL: https://www.rbc.ua/rus/news/kadrova-kriza-ukrayini-k-brak-spetsialistiv-1727103044.html (дата звернення: 31.08.2025).

18. Kyiv IT Market Review (EchoGlobal.tech): URL: https://echoglobal.tech/kyiv-it-market-review-industry-research/ (дата звернення: 31.08.2025).

19. How technology helped Ukraine resist during wartime (Microsoft news) : URL: https://news.microsoft.com/en-cee/2023/01/20/how-technology-helped-ukraine-resist-during-wartime/ (дата звернення: 31.08.2025).

20. Israel stage3 cyber wars with Iran proxies: URL: https://www.darkreading.com/threat-intelligence/israel-stage-3-cyber-wars-with-iran-proxies (дата звернення: 31.08.2025).

21. Пономаренко Н. Тимчасове розміщення державних реєстрів за кордоном: мета та строки дії: електронний ресурс. URL: https://eba.com.ua/tymchasove-rozmishhennya-derzhavnyh-reyestriv-za-kordonom-meta-ta-stroky-diyi (дата звернення: 31.08.2025).

22. Про критичну інфраструктуру : Закон України від 16.11.2021 р. № 1882-IX: станом на 21 вересня 2024 р. URL: https://zakon.rada.gov.ua/laws/show/1882-20#Text (дата звернення: 01.09.2025).

23. Про затвердження Положення про організаційно-технічну модель кіберзахисту: Постанова Кабінету Міністрів України від 29.12.2021 р. № 1426: станом на 26 грудня 2024 р. URL: https://zakon.rada.gov.ua/laws/show/1426-2021-п (дата звернення: 01.09.2025).

24. Нормативні документи системи ТЗІ : URL: https://cip.gov.ua/ua/news/normativni-dokumenti-sistemi-tzi2024 (дата звернення: 31.08.2025).

25. Про реалізацію експериментального проекту з декларування відповідності комплексних систем захисту інформації в інформаційних, електронних комунікаційних та інформаційно-комунікаційних системах, створених з використанням профілів безпеки інформації: Постанова Кабінету Міністрів України від 30.05.2024 № 627: станом на 19 червня 2025 р. URL: https://zakon.rada.gov.ua/laws/show/627-2024-п#Text (дата звернення: 01.09.2025).

26. Про затвердження Загальних вимог до кіберзахисту об'єктів критичної інфраструктури: Постанова Кабінету Міністрів України від 19.06.2019 р. № 518: станом на 7 верес. 2022 р. URL: https://zakon.rada.gov.ua/laws/show/518-2019-п#Text (дата звернення: 01.09.2025).

27. Про затвердження Змін до деяких нормативно-правових актів Національного банку України з питань інформаційної безпеки та кіберзахисту: Постанова Національного Банку України від 25.02.2025 № 24.

28. Про захист персональних даних: Закон України від 01.06.2010 р. № 2297-VI: станом на 14 червня 2025 р. URL: https://zakon.rada.gov.ua/laws/show/2297-17#Text (дата звернення: 01.09.2025).

29. Про порядок денний тринадцятої сесії Верховної Ради України дев'ятого скликання: Постанова Верховної Ради України від 11.02.2025 р. № 4229-IX: станом на 21 серпня 2025 р. URL: https://zakon.rada.gov.ua/laws/show/4229-20#Text (дата звернення: 01.09.2025).

30. Деякі питання реагування суб'єктами забезпечення кібербезпеки на різні види подій у кіберпросторі: Постанова Кабінету Міністрів України від 04.04.2023 р. № 299. URL: https://zakon.rada.gov.ua/laws/show/299-2023-п#Text (дата звернення: 01.09.2025).

31. Про затвердження Порядку ведення Переліку надавачів хмарних послуг та/або послуг центру обробки даних та форми заяви про внесення надавача хмарних послуг та/або послуг центру обробки даних до Переліку надавачів хмарних послуг та/або послуг центру обробки даних: Наказ Адміністрації Державної служби спеціального зв'язку та захисту інформації України від 07.04.2025 р. № 231. URL: https://zakon.rada.gov.ua/laws/show/z0778-25#Text (дата звернення: 01.09.2025).

32. Деякі питання захисту інформаційних, електронних комунікаційних, інформаційно-комунікаційних, технологічних систем: Постанова Кабінету Міністрів України від 18.06.2025 р. № 712: станом на 19 червня 2025 р. URL: https://zakon.rada.gov.ua/laws/show/712-2025-%D0%BF#Text (дата звернення: 01.09.2025).

33. Деякі питання забезпечення функціонування інформаційно-комунікаційних систем, електронних комунікаційних систем, публічних електронних реєстрів в умовах воєнного стану: Постанова Кабінету Міністрів України від 12.03.2022 р. № 263: станом на 06 червня 2025 р. URL: https://zakon.rada.gov.ua/laws/show/263-2022-%D0%BF#Text (дата звернення: 01.09.2025).

34. Про критичну інфраструктуру: Закон України від 25.10.2022 р. № 8153: станом на 21 вересня 2024 р. URL: https://itd.rada.gov.ua/billinfo/Bills/Card/40707 (дата звернення: 01.09.2025).

35. Про хмарні послуги: Закон України від 28.06.2024 р. № 2075-IX: станом на 28 червня 2024 р. URL: https://zakon.rada.gov.ua/laws/show/2075-20#Text (дата звернення: 01.09.2025).

36. Про затвердження Положення про організацію заходів із забезпечення інформаційної безпеки в банківській системі України: Постанова Національного Банку України від 28.09.2017 р. № 95: станом на 01 березня 2018 р. URL: https://zakon.rada.gov.ua/laws/show/v0095500-17#Text (дата звернення: 01.09.2025).

Дата надходження статті: 18.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.93:004.94
DOI https://doi.org/10.32689/maup.it.2025.3.20

Сергій РЕВА
кандидат технічних наук, доцент кафедри комп'ютерних систем та робототехніки,
Харківський національний університет імені В. Н. Каразіна,
iec-lab@karazin.ua
ORCID: *0000-0002-2615-9226*

Денис ЦИБЛІЄВ
аспірант кафедри комп'ютерних систем та робототехніки,
Харківський національний університет імені В. Н. Каразіна,
dtsibliev@gmail.com
ORCID: *0009-0008-4373-8773*

РОЗРОБКА ПРОГРАМНОЇ ПЛАТФОРМИ ДЛЯ КОМП'ЮТЕРНОГО МОДЕЛЮВАННЯ, АНАЛІЗУ ТА ВЕРИФІКАЦІЇ ПАРАМЕТРІВ СПЕКТРОМЕТРИЧНИХ СИГНАЛІВ

Анотація. *Мета роботи. Стаття присвячена розробці програмної платформи, яка дозволяє комплексно досліджувати та використовувати методи комп'ютерного аналізу оцифрованих спектрометричних сигналів. Функціонал розробленого програмного засобу включає в себе моделювання цифрових образів сигналів з повністю відомими, регульованими параметрами, комп'ютерну обробку даних за допомогою існуючих або нових розроблених методів аналізу, а також програмну верифікацію та візуалізацію результатів роботи таких методів. Крім цього програма підтримує можливість завантаження даних, що були отримані під час реальних експериментів, та їх подальший аналіз для побудови спектрів.*

Методологія. *У статті наводиться детальний опис можливостей програмної платформи та її внутрішньої архітектури. Функціонал та графічний інтерфейс програми створені з використанням методів та технологій розробки програмного забезпечення на мові програмування C++ на основі фреймворку QT. Даний фреймворк є кросплатформним, що дозволяє компілювати та запускати розроблений додаток на різних операційних системах, таких як Windows та Linux. Для генерації цифрових образів спектрометричних сигналів застосовуються методи математичного та комп'ютерного моделювання. В процесі комп'ютерної обробки даних використовуються методи цифрової обробки сигналів, методи і алгоритми інтелектуального аналізу великих масивів даних. Наприкінці наводиться порівняльний аналіз результатів роботи декількох існуючих та нового методу комп'ютерного аналізу, що були отримані за допомогою створеного програмного засобу.*

Наукова новизна. *Вперше розроблено платформу (програмний засіб), яка надає можливості комплексного дослідження точності та швидкодії як відомих, так і нових розроблених методів комп'ютерного аналізу параметрів спектрометричних сигналів. Введено чіткі критерії оцінювання точності роботи (поняття верифікованої точності) того чи іншого методу комп'ютерної обробки на змодельованих даних, які перевіряються за допомогою програмно реалізованого алгоритму верифікації.*

Висновки. *Створений в ході дослідження програмний засіб дозволяє виконувати комп'ютерне моделювання спектрометричного сигналу із заданими, регульованими параметрами і здійснювати аналіз симульованих або завантажених з реальних експериментів даних за допомогою програмно реалізованих існуючих та запропонованих методів комп'ютерної обробки. Результати дослідження свідчать, що програма дозволяє обчислити та порівняти основні метрики роботи методів комп'ютерного аналізу, такі як швидкість обробки даних і точність розпізнавання основних параметрів імпульсів, а також візуалізувати результати. В перспективі функціональні можливості платформи можуть бути розширені шляхом додавання підтримки більшого числа методів комп'ютерного аналізу, що дозволить краще дослідити ефективність як відомих, так і нових методів комп'ютерної обробки спектрометричних сигналів.*

Ключові слова: *комп'ютерний аналіз спектрометричних сигналів, програмна платформа, комп'ютерне моделювання, алгоритми розпізнавання, комп'ютерна система, візуалізація даних, алгоритми верифікації.*

Sergiy REVA, Denys TSYBLIYEV. DEVELOPMENT OF A SOFTWARE PLATFORM FOR COMPUTER MODELING, ANALYSIS AND VERIFICATION OF SPECTROMETRIC SIGNALS

Abstract. Purpose of the work. *The article is devoted to the development of a software platform that allows for the comprehensive study and use of methods of computer analysis of digitized spectrometric signals. The functionality of the developed software application includes modeling of digital signal images with fully known, adjustable parameters, computer data processing using existing or newly developed analysis methods, as well as software verification and visualization of the results of such methods. In addition, the program supports the ability to load data obtained during real experiments and their further analysis to construct spectra.*

Methodology. *The article provides a detailed description of the capabilities of the software platform and its internal architecture. The functionality and graphical interface of the program are created using methods and technologies of software development in the C++ programming language based on the QT framework. This framework is cross-platform, allowing you to compile and run the developed application on different operating systems, such as Windows and Linux. Mathematical and computer modeling methods are used to generate digital images of spectrometric signals. In the process of computer data analysis, digital signal processing methods, methods and algorithms for intelligent analysis of large data sets are used. At the end, a comparative analysis of the results of several existing and new computer analysis methods obtained using the created software is presented.*

Scientific novelty. *For the first time, a platform (software tool) has been developed that provides the ability to comprehensively study the accuracy and speed of both known and newly developed methods of computer analysis of spectrometric signal parameters. Clear criteria for assessing the accuracy of work (the concept of verified accuracy) of a particular computer processing method on simulated data, which is verified using a software-implemented verification algorithm, have been introduced.*

Conclusions. *The software application created during the research allows for computer modeling of a spectrometric signal with specified, adjustable parameters, as well as analysis of simulated or loaded from real experiments data using software-implemented existing and proposed computer processing methods. The results of the study show that the program allows you to calculate and compare the main metrics of the work of computer analysis methods, such as data processing speed and accuracy of recognition of the main pulse parameters, as well as visualize the results. In the future, the platform's functionality can be expanded by adding support for a larger number of computer analysis methods, which will allow for better research into the effectiveness of both known and new methods of computer processing of spectrometric signals.*

Key words: *computer analysis of spectrometric signals, software platform, computer modeling, recognition algorithms, computer system, data visualization, verification algorithms.*

Поставка проблеми. Протягом довгого часу класичні методи аналізу спектрометричних сигналів базувалися на використанні аналогової електроніки. Але завдяки активному розвитку комп'ютерних технологій значного поширення набули комп'ютерні методи обробки даних та вимірювання спектрів. Під поняттям «спектрометричного сигналу» в цифровій спектрометрії іонізуючого випромінювання зазвичай розуміють послідовність імпульсів (рис. 1), які генеруються детекторами рентгенівського, гамма або іншого типу випромінювання, та оцифровуються за допомогою аналого-цифрових перетворювачів (АЦП) або діджітайзерів [1; 6]. Ці сигнали в оцифрованому вигляді являють собою великі масиви даних, які можуть бути оброблені методами комп'ютерного аналізу за певними алгоритмами з метою виявлення корисної інформацію про матеріали і процеси, що досліджуються. Проте існує проблема об'єктивного оцінювання ефективності того чи іншого методу комп'ютерної обробки, оскільки немає можливості отримання повністю достовірних вхідних даних через випадковість процесів на вході детектора. Ця проблема може бути віришена за допомогою комп'ютерного моделювання [4] цифрових образів сигналів із заздалегідь відомими параметрами та програмної верифікації результатів аналізу цих даних. Тому розробка програмної платформи, яка може здійснювати в комплексі комп'ютерне моделювання, аналіз та верифікацію параметрів спектрометричних сигналів є актуальним завданням.

Аналіз останніх досліджень і публікацій. Сучасні дослідження у галузі комп'ютерного аналізу спектрометричних сигналів часто підкреслюють важливість механізмів зменшення впливу електричного шуму в процесі обробки цифрових даних та спрямовані на покращення ефективності методів аналізу, особливо при частій суперпозиції імпульсів. Так, у роботі [12] було представлено метод аналізу оцифрованого спектрометричного сигналу отриманого зі сцинтиляційного детектора, одним з етапів роботи якого є фільтрація сигналу від шуму за допомогою застосування цифрових фільтрів (ковзне середнє, фільтр Бесселя, Чебишева або Баттерворта). У роботі [7] було запропоновано алгоритм

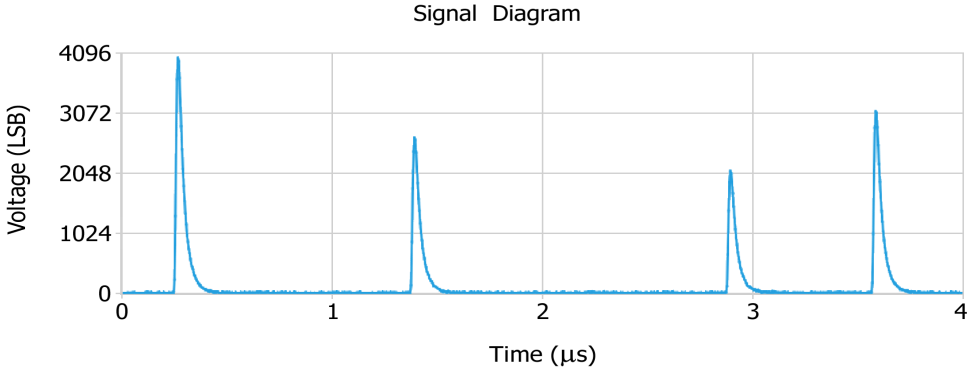


Рис. 1. Візуалізація оцифрованого спектрометричного сигналу

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

послаблення впливу суперпозиції імпульсів (pile-up ефекту) на результуючий спектр, що базується на розрідженій апроксимації сигналу для розділення накладених імпульсів та використанні методів регресійного аналізу Least Absolute Shrinkage and Selection Operator (LASSO). Представлений підхід дозволив зменшити вплив pile-up ефекту та покращити результуючий досліджений спектр. Проте оскільки у вищезазначених працях перевірка роботи запропонованих алгоритмів проводилася тільки на реальних спектрах, аналіз результатів їх роботи на змодельованих, повністю відомих вхідних даних з додаванням механізму верифікації дозволив би краще дослідити точність розпізнавання імпульсів даними методами.

У роботах [5; 11] авторами було представлено підхід до комп'ютерного моделювання спектрометричних сигналів, а також детально досліджено та отримано порівняльні характеристики одразу кількох методів аналізу таких сигналів: методів Максимуму (Maximum), Сум (Sum), Підбору (Fitting) та Деконволюції (Deconvolution). За допомогою розробленого авторами програмного забезпечення (DeGaSum) було отримано результати роботи цих методів на змодельованих цифрових даних і візуалізовано на діаграмах залежність розпізнавання кількості імпульсів від рівня завантаження детектора. Окрім цього, за допомогою кожного з підходів було отримано спектри та проаналізовано їх характеристики. Беручи до уваги той факт, що при високих рівнях завантаження (кількість імпульсів за одиницю часу) декілька імпульсів можуть накладатися, формуючи один з великою амплітудою, невирішеним питанням залишилася верифікація результатів розпізнавання кожним методом. Тобто, перевірка того, чи розпізнаний імпульс співпадає з тим, що був згенерований під час моделювання, що важливо для об'єктивного визначення точності розпізнавання.

Формулювання мети дослідження. Метою даної роботи є створення програмного засобу для комплекного дослідження та оцінки ефективності методів комп'ютерного аналізу оцифрованих спектрометричних сигналів. А також детальний опис функціоналу розробленого додатку, його внутрішньої архітектури, принципів роботи та отриманих експериментальних результатів.

Виклад основного матеріалу. Для комплексного аналізу точності та швидкодії методів комп'ютерної обробки оцифрованих спектрометричних сигналів в рамках дослідження була розроблена програмна платформа, яка реалізує наступні можливості:

1) комп'ютерне моделювання (симуляція) цифрових образів спектрометричних сигналів, які за форматом відповідають вихідним даним діджітайзера та наближені до реальних сигналів, що отримуються під час проведення експериментів;

2) розпізнавання та вимірювання параметрів імпульсних сигналів за допомогою програмно реалізованих існуючих та нових розроблених методів аналізу;

3) верифікація точності аналізу та отримання метрик ефективності роботи різних методів комп'ютерної обробки;

4) візуалізація отриманих результатів.

Програмний засіб являє собою додаток для персональних комп'ютерів, який був розроблений на мові програмування C++ з використанням бібліотеки QT [8] (версії 6.4.0). Дана бібліотека була обрана через те, що містить широкий набір графічних компонентів для візуалізації даних, а також дозволяє створювати кросплатформне програмне забезпечення. Тому програмний засіб може бути встановлений на декілька різних операційних систем, зокрема на Windows та Unix-подібні ОС.

На (рис. 2) наводиться вигляд основного інтерфейсу користувача розробленої програми.

Головне вікно програми містить наступні групи елементів інтерфейсу користувача відповідно до свого функціоналу:

A. Елементи налаштування та управління моделюванням (симуляцією) цифрового образу сигналу.

B. Елементи налаштування та управління аналізом спектрометричного сигналу.

C. Елементи виводу інформації про результати моделювання та аналізу.

D. Діаграма відображення розподілу амплітуд (спектру) змодельованих/розпізнаних/верифікованих імпульсів.

E. Діаграма відображення спектрометричного сигналу (симульованого або завантантаженого з діджітайзера).

Моделювання спектрометричного сигналу**.** Додаток надає можливість моделювання образу сигналу зі сталим (константним) значенням амплітуд імпульсів (з опцією збереження симульованих даних у файл) або моделювання згідно попередньо завантаженого спеціального файлу-шаблону, який задає розподіл амплітуд імпульсів. Для генерації цифрового образу сигналу з потрібним законом розподілу амплітуд та потрібними характеристиками було використано математичні моделі і алгоритми моделювання згідно файлів шаблонів та метод моделювання з підвищеною деталізацією, які були детально описані у працях [2; 3]. За необхідності можна також застосовувати і спеціально створені

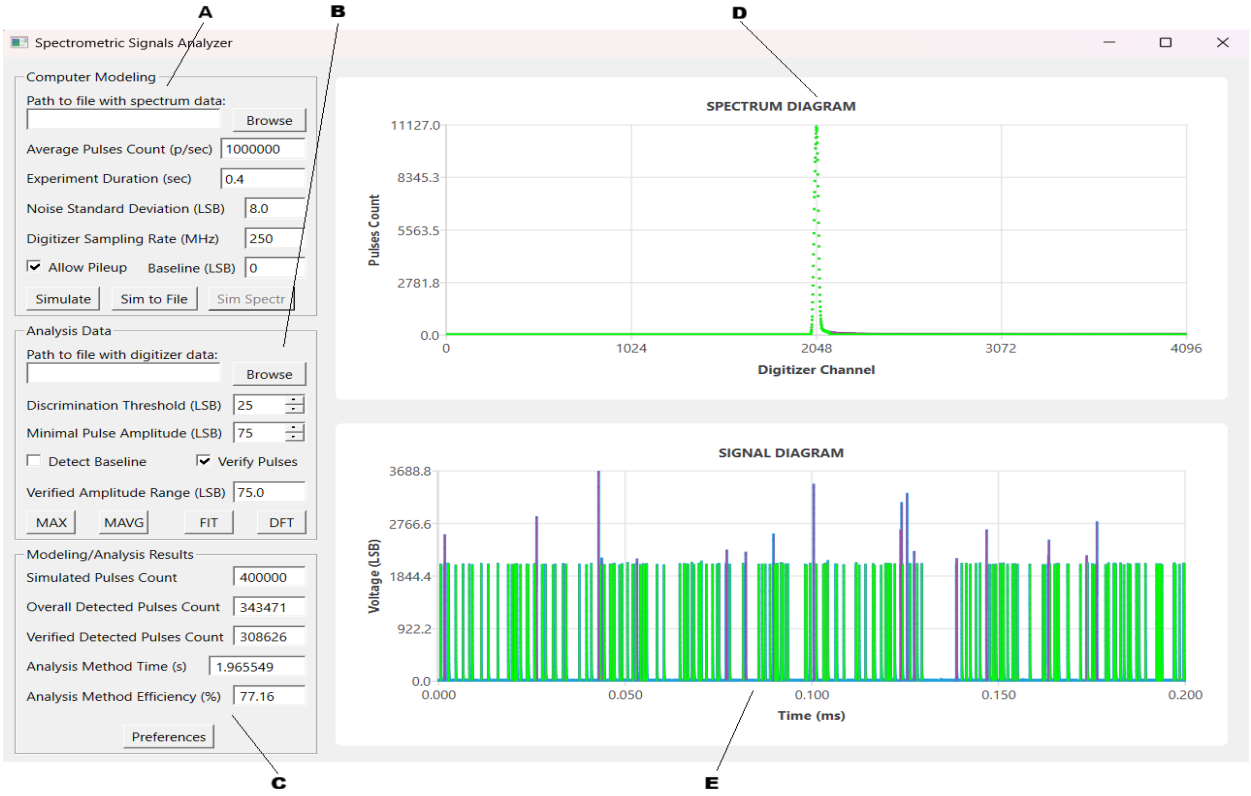


Рис. 2. Вигляд інтерфейсу користувача головного вікна розробленого програмного засобу

(ідеалізовані) функції розподілу. Миттєві значення сигналу відображаються в одиницях значення найменшого розряду діджітайзера (Least Significant Bit – LSB).

При симуляції цифрового образу сигналу можливо задати наступні параметри:

- Тривалість експерименту (в секундах)
- Рівень завантаження детектора (середня кількість імпульсів за секунду)
- Середньоквадратичне відхилення рівня електричного шуму, що накладається на сигнал згідно нормального розподілу (в LSB)
- Частота дискретизації сигналу (в МГц)
- Значення базової лінії сигналу (в LSB)
- Можливість суперпозиції імпульсів (pile-up ефекту) (так/ні)

Окрім моделювання, в програмі була реалізована можливість завантаження цифрових даних, які були отримані під час реальних експериментів та оцифровані за допомогою діджітайзера. Діаграма Е візуалізує змодельований імпульсний сигнал або сигнал, що був завантажений з файлу з даними, записаними під час реальних експериментів.

Комп'ютерний аналіз**,** верифікація та відображення результатів**.** Для подальшого комп'ютерного аналізу було реалізовано два існуючі методи по визначенню параметрів спектрометричних сигналів – Максимуму (Maximum) та Підбору (Fitting), основні принципи роботи яких описано в статті [5]. Також було програмно реалізовано вдосконалений метод аналізу з механізмом фільтрації сигналу від шуму за допомогою фільтра ковзне середнє [9] та розроблений метод аналізу під назвою Відстеження [10] з використанням власних алгоритмічних підходів. Результати роботи вищезазначених методів візуалізуються у вигляді числових параметрів, часових діаграм сигналів та гістограм спектрів, побудованих на основі проведеного аналізу.

Для конфігурації роботи методів аналізу можливо вказати наступні параметри:

- Поріг дискримінації (в LSB).
- Мінімальне (порогове) значення амплітуди розпізнаного імпульса (в LSB).
- Необхідність визначення базової лінії сигналу (так/ні)

Також надається можливість обрати опцію чи потрібно виконувати верифікацію розпізнаних імпульсів після роботи методу аналізу і якщо так, то вказати допустимий діапазон максимального відхилення розпізнаних імпульсів від змодельованих по амплітуді (в LSB).

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Після запуску обраного методу комп'ютерної обробки додаток дозволяє програмно виміряти та вивести інформацію про тривалість його роботи, загальну і верифіковану кількість розпізнаних імпульсів та точність методу. Для перевірки правильності розпізнавання при аналізі на симульованих даних з відомими параметрами було використано розроблений алгоритм верифікації представлений у роботі [10], який порівнює згенеровані та розпізнані імпульси на співпадіння з допустимими діапазонами відхилення, що можуть бути задані в інтерфейсі програми. Верифікована точність методу аналізу визначається співвідношенням кількості верифікованих імпульсів до загального числа змодельованих.

В групі елементів інтерфейсу відображення результатів моделювання/аналізу/верифікації виводяться наступні дані:

- Кількість змодельованих імпульсів.
- Кількість загалом розпізнаних імпульсів.
- Кількість верифікованих імпульсів.
- Тривалість роботи методу аналізу (в секундах).
- Верифікована точність методу аналізу (у відсотках)

Діаграма D (рис. 2) відображає розподіл амплітуд (результуючий спектр) змодельованих, розпізнаних та верифікованих імпульсів та дозволяє наглядно побачити наскільки ці графіки співпадають між собою.

Окрім описаних вище елементів, в програмі реалізовані додаткові налаштування, які дозволяють змінювати параметри діаграм, що відображають спектрометричний сигнал та отриманий результуючий спектр після роботи методів комп'ютерної обробки даних.

Архітектура програми**.** Оскільки програмний засіб створений на об'єктно-орієнтованій мові програмування C++ з використанням бібліотеки QT, то загалом весь функціонал реалізований за допомогою об'єктів-класів, кожен з яких виконує свої певні задачі. На рисунку 3 зображена UML діаграма класів, що візуалізує внутрішню архітектуру програми, а саме основні розроблені компоненти (C++ класи, їх основні методи) та зв'язки між ними. Нижче наводяться опис класів представлених цій UML діаграмі та задач, які вони виконують:

- SSAnalyzerWnd – клас, що наслідується від QWidget бібліотеки QT та реалізує функціонал/інтерфейс користувача головного вікна додатку, загальний вигляд якого був представлений на рисунку 2. Для реалізації комп'ютерного моделювання (симуляції) сигналу, програмного аналізу, відображення результатів та додаткових налаштувань даний клас агрегує інші об'єкти, що описані нижче.
- SignalSimulator – клас, що містить необхідні дані та методи для реалізації моделювання спектрометричного сигналу із заданими користувачем параметрами. Також даний клас реалізує методи для загрузки спеціальних файлів-шаблонів (*.dat, *.spr) для генерації цифрового образу сигналу згідно шаблонного розподілу амплітуд імпульсів використовуючи розроблені моделі і алгоритми моделювання описані в роботі [3].
- SignalAnalyzer – один з найбільших класів, який містить програмну реалізацію існуючих та розроблених методів аналізу спектрометричних сигналів. Методи AnalyzeMax, AnalyzeMAVRG, AnalyzeFitting, AnalyzeTracking відповідають за комп'ютерний аналіз цифрових даних методами Максимуму, Ковзне Середнє, Підбору, Відстеження відповідно. Окрім цього метод VerifyDetectedPulses цього класу містить програмну реалізацію алгоритму верифікації для співставлення згенерованих та розпізнаних імпульсів після аналізу з метою визначення верифікованої точності кожного з методів комп'ютерної обробки.
- PreferencesDialog – даний клас наслідується від QDialog і реалізує відображення та функціонал вікна додаткових налаштувань програми. Містить методи для відображення, зміни та збереження додаткових налаштувань.
- Chart, ChartView – класи, які наслідуються від QChart та QChartView бібліотеки QT і відповідають за відображення діаграми цифрового сигналу та діаграми розподілу амплітуд (спектру) розпізнаних імпульсів. Додатково до стандартного набору функцій класів з бібліотеки QT в цих компонентах була реалізована можливість масштабування (збільшення/зменшення) окремих участків діаграми для більш детального аналізу отриманих графіків.

Порівняльний аналіз точності та швидкодії декількох методів комп'ютерної обробки**.** Використовуючи розроблений програмний засіб було проведено дослідження точності та швидкодії відомих методів аналізу спектрометричних сигналів (Максимуму та Підбору) і нового розробленого методу під назвою Відстеження [10] на змодельованих даних. Цифровий образ сигналу для подальшого аналізу було симульовано з наступними параметрами: рівень завантаження детектора (інтенсивність генерації імпульсів) – 10^6 імпульсів за секунду, тривалість експерименту – 0.4 секунди, загальна кількість

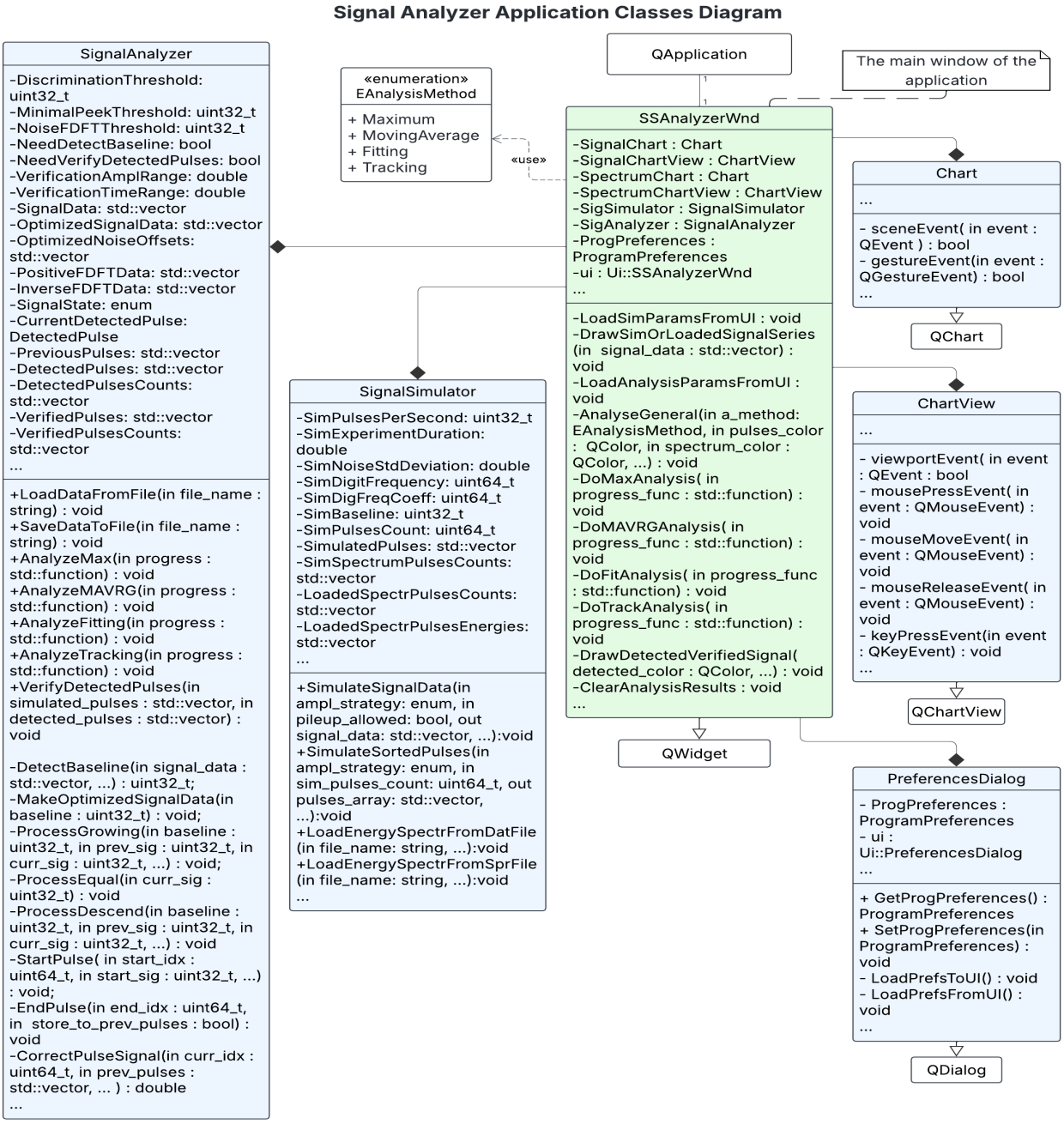


Рис. 3. UML діаграма класів, що реалізують функціонал додатку

змодельованих імпульсів – 400000, значення амплітуд кожного з імпульсів – 2048 LSB, середньоквадратичне відхилення рівня електричного шуму – 8 LSB, допустимий діапазон відхилення по амплітуді верифікованих імпульсів – 75 LSB. Порівняльні результати комп'ютерної обробки модельованих даних вищезазначеними методами аналізу наведені в (табл. 1).

Таблиця 1

Результати роботи методів аналізу на змодельованих вхідних даних

Назва методу	Кількість змодельованих імпульсів	Кількість загалом розпізнаних імпульсів	Кількість верифікованих імпульсів	Час роботи методу (сек)	Верифікована точність методу (%)
Максимуму	400000	343471	308626	1.965	77.16
Підбору	400000	390995	377920	5.919	94.48
Відстеження	400000	391349	378961	11.063	94.74

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Як можна побачити з (табл. 1) в досліджуваному сценарії простий метод Максимуму продемонстрував найкращу швидкодію, проте його верифікована точність є відносно низькою внаслідок того, що даний метод не здатний правильно розпізнавати імплітуди імпульсів при їх суперпозиції. Метод Відстеження показав на 18% кращу верифіковану точність аналізу, ніж метод Максимумів, та дещо кращу верифіковану точність, ніж існуючий метод Підбору.

Вис**н**овки**.** Розроблена в ході дослідження програмна платформа дозволяє комплексно оцінити та візуалізувати результати роботи методів комп'ютерної обробки спектрометричних сигналів. Додаток реалізує можливість комп'ютерного моделювання цифрових образів сигналів з необхідними параметрами або завантаження даних, що були записані під час реальних експериментів. Наведені результати підтверджують, що такі дані можуть бути програмно проаналізовані за допомогою реалізованих існуючих та розроблених методів аналізу з метою визначення ключових параметрів сигналів та побудови спектрів. Чіткі критерії точності розпізнавання параметрів імпульсів, які були введені в рамках дослідження, та механізм програмної верифікації дозволили більш об'єктивно оцінити і порівняти точність різних підходів спектрального аналізу.

В подальшому функціональні можливості програмного засобу можуть бути розширені шляхом додавання підтримки більшого числа методів обробки даних, що дозволить краще дослідити ефективність як відомих, так і нових розроблених методів комп'ютерного аналізу спектрометричних сигналів.

Список використаних джерел:

1. Грабовський В. А. Прикладна спектрометрія йонізуючих випромінювань: Навчальний посібник. Видавничий центр ЛНУ імені Івана Франка. 2008. 296 с.

2. Рева С. М., Цибліев Д. О. Комп'ютерне моделювання спектрометричних сигналів з підвищеною деталізацією. *Вісник Харківського національного університету імені В. Н. Каразіна, серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління»*. 2024. Том 65. С. 64–73. https://doi.org/10.26565/2304-6201-2025-65-06

3. Рева С. М., Цибліев Д. О. Математичні моделі та алгоритми комп'ютерного моделювання спектрометричних сигналів. *Вісник Харківського національного університету імені В.Н. Каразіна, сер. «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління».* 2023. Том 58. С.64–74. URL: https://periodicals.karazin.ua/mia/article/view/23502

4. Averill M. Law, W. David Kelton. Simulation Modeling and Analysis. Third edition. McGraw-Hill. 2000. 760 pages.

5. Khilkevitch E. M., Shevelev A. E., Chugunov I. N., Iliasova M. V., Doinikov, D. N., Gin D. B. et al. Advanced algorithms for signal processing scintillation gamma ray detectors at high counting rates. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*. 2020. Volume 977, 164309. URL: https://doi.org/10.1016/j.nima.2020.164309

6. Knoll G. F. Radiation Detection and Measurement. John Wiley & Sons. 2010. 864 pages.

7. Lopatin M., Moskovitch N., Trigano T., Sepulcre Y. Pileup attenuation for spectroscopic signals using a sparse reconstruction. IEEE 27th Convention of Electrical and Electronics Engineers in Israel. 2012. P. 1–5. URL: https://doi.org/10.1109/eeei.2012.6377045

8. QT Framework Official Website. URL: https://www.qt.io/product/framework

9. Reva S. M., Tsybliyev D. O. Computer methods of recognition and analysis of X-ray and gamma radiation parameters. *Bulletin of V. N. Karazin Kharkiv National University, series "Mathematical modeling. Information technology. Automated control systems".* 2022. Volume 55, pp.38–48. URL: https://periodicals.karazin.ua/mia/article/view/22593

10. Reva S. M., Tsybliyev D. O. Devising a computer method to recognize and analyze spectrometric signals parameters. *Eastern-European Journal of Enterprise Technologies.* 2024. 6(9 (132)), 86–96. URL: https://doi.org/10.15587/1729-4061.2024.318558

11. Shevelev A. E., Khilkevitch E. M., Lashkul S. I., Rozhdestvensky V. V., Altukhov A. B., Chugunov I. N. et al. High performance gamma-ray spectrometer for runaway electron studies on the FT-2 tokamak. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment.* 2016. Volume 830, pp. 102–108. URL: https://doi.org/10.1016/j.nima.2016.05.075

12. Wolszczak W., Dorenbos P. Time-resolved gamma spectroscopy of single events. Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment. 2018. Volume 886, pp. 30–35. URL: https://doi.org/10.1016/j.nima.2017.12.080

Дата надходження статті: 17.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 519.857
DOI https://doi.org/10.32689/maup.it.2025.3.21

Марія СЕМАНЬКІВ
кандидат технічних наук, доцент кафедри комп'ютерних наук та інформаційних систем,
Карпатський національний університет імені Василя Стефаника,
mariia.semankiv@pnu.edu.ua
***ORCID:** 0000-0002-1314-8923*

ВИКОРИСТАННЯ АЛГОРИТМУ ВЕЛЬЦЛЯ ДЛЯ РОЗВ'ЯЗАННЯ ЗАДАЧІ КОМІВОЯЖЕРА

Анотація. *Стаття присвячена вирішенню однієї з NP-задач, а саме пошуку оптимального маршруту комівояжера для відвідання кожного із n заданих міст. Ефективні алгоритми розв'язання даної задачі дозволяють знаходити оптимальні маршрути навіть для великих наборів міст, що зменшує витрати часу, пального та ресурсів. Поєднання високої точності з малою обчислювальною складністю робить такі методи придатними для використання в реальних системах, де рішення потрібно приймати швидко.*

Мета роботи *– вдосконалення методу гілок та меж для розв'язання задачі комівояжера за рахунок застосування алгоритму Вельцля та використання мінімального охоплюючого кола (Minimum Enclosing Circle, MEC) як евристики для підсилення відсікаючих правил у даному методі.*

Методологія. *Алгоритм Вельцля – це класичний приклад ефективного геометричного методу з відсіками та евристикою, який можна легко включати до більш складних алгоритмів. У задачах розміщення, пакування, колізій, кластеризації тощо, алгоритм Вельцля може бути використаний як частина оціночної або відсікаючої функції: обчислити мінімальне коло, що містить підмножину об'єктів-міст, щоб оцінити обсяг/простір і якщо коло з новою точкою виходить за дозволені межі – відсікати гілку.*

Наукова новизна. *Автором запропоновано використання алгоритму Вельцля для прискорення точного алгоритму розв'язання задачі комівояжера. Вигода у знаходженні мінімального кола, що охоплює множину точок-міст на площині, полягає у отриманні більш тісної нижньої межі гілки в методі гілок та меж, у збільшенні кількості гілок у дереві пошуку, які відсічуться раніше, і як наслідок метод запрацює швидше. MEC відсікає ті гілки, у яких невідвідані міста лежать на великій відстані одне від одного, і навіть найоптимальніший добудований маршрут буде занадто дорогим. Це зменшує простір пошуку і прискорює алгоритм.*

Висновки. *Метод гілок та меж із використанням мінімального охоплюючого кола MEC має широкий спектр практичного застосування у задачах, де необхідно швидко отримати якісний маршрут з мінімальними обчислювальними витратами, якщо допускається невелике відхилення від оптимального розв'язку. Таким чином, використання MEC у поєднанні з методом гілок та меж є універсальним підходом, що поєднує точність математичної оптимізації та швидкість евристичних методів і може бути впроваджене в будь-якій сфері, де маршрутизація має велике практичне значення.*

Ключові слова: *задача комівояжера (TSP), алгоритм Вельцля, метод гілок та меж.*

Mariia SEMANKIV. THE USE OF WELZL'S ALGORITHM FOR SOLVING THE TRAVELING SALESMAN PROBLEM

Abstract. *This article is devoted to solving one of the NP-hard problems, namely finding the optimal traveling salesman route for visiting each of the n given cities. Efficient algorithms for solving this problem make it possible to find optimal routes even for large sets of cities, which reduces time, fuel, and resource costs. Combining high accuracy with low computational complexity makes such methods suitable for real-world systems where decisions must be made quickly.*

The purpose of this work *is to improve the branch-and-bound method for solving the traveling salesman problem by applying Welzl's algorithm and using the Minimum Enclosing Circle (MEC) as a heuristic to enhance the pruning rules in the branch-and-bound method.*

Methodology. *Welzl's algorithm is a classical example of an efficient geometric method with pruning and heuristics, which can be easily integrated into more complex algorithms, including the branch-and-bound method. In problems of placement, packing, collision detection, clustering, and others, Welzl's algorithm can be used as part of an evaluation or pruning function: compute the minimum circle that contains a subset of city objects to estimate the space/volume, and if the circle with a new point goes beyond the allowed limits, prune the branch.*

Scientific novelty. *The author proposes using Welzl's algorithm to speed up the exact solution of the traveling salesman problem. The benefit of finding the minimum circle covering a set of city points on the plane lies in obtaining a tighter lower bound for a branch in the branch-and-bound method, increasing the number of branches in the search tree that are pruned earlier, and consequently making the branch-and-bound method faster. MEC prunes branches in which the unvisited cities lie far apart, and even the most optimal extended route would be too costly. This reduces the search space and accelerates the algorithm.*

Conclusions. *The branch-and-bound method with the use of the Minimum Enclosing Circle has a wide range of practical applications in problems where it is necessary to quickly obtain a high-quality route with minimal computational cost, even if a slight deviation from the optimal solution is acceptable. Thus, using MEC in combination with the branch-and-bound method is a universal approach that combines the precision of mathematical optimization with the speed of heuristic methods and can be implemented in any field where routing has significant practical importance.*

Key words: *traveling salesman problem (tsp), Welzl's algorithm, branch and bound method.*

Постановка проблеми. Задача комівояжера (Traveling Salesman Problem, TSP) залежно від формулювання відноситься до різних класів у сімействі *NP*-задач. TSP як задача розпізнавання є *NP*-повною задачею, оскільки будь-яке розв'язання можна перевірити за поліноміальний час (пройти по циклу, підрахувати суму ваг і перевірити ≤ *K*). TSP як оптимізаційна задача є *NP*-складною задачею, адже ця задача складніша, ніж клас *NP*, бо немає поліноміального алгоритму перевірки оптимальності (можна перевірити, що цикл має певну довжину, але довести, що він мінімальний – важче) [2; 10].

Задача комівояжера полягає у пошуку оптимального маршруту для відвідання кожного із *n* міст. Комівояжеру необхідно побувати у кожному місті рівно один раз, і повернутися у вихідне, з якого була розпочата мандрівка. Відомо що переміщення із міста *i* у місто *j* зазначається вартістю *c(i,j)* гривень. А також можливе опрацювання задачі відносно відстані між містами. Якщо згадати теорію графів, то можна так описати задачу: потрібно відшукати гамільтонів цикл у визначеному графі із найменшою вартістю (сума вартості кожного ребра циклу буде його загальною ціною). У відповідність можна поставити задачу вирішення, яка звучатиме так: чи є у графі *G* Гамільтонів цикл, вартістю меншою або такою ж як значення *k*. Мова даної проблеми математично описується таким чином: *G=(V,E)* – зв'язний неорієнтовний граф з множиною вершин *|V|=n*, кожному ребру *e* ∈ *E* приписана вага (вартість) *w(e)≥0*.

$$TSP_{\le k}=\{ (G, w, k) | \exists\ C \text{ Гамільтонів цикл: } f(C) \le k\},$$

де *f(C)* – це функція вартості Гамільтонового циклу *C*, *k*∈*Z*, у графі *G* наявний цикл Гамільтона із вартістю на більшою ніж *k* [2].

Більшість прийнятних по часу та кількості вхідних даних методів для розв'язку даної задачі є евристичними (дають не оптимальний, а наближений до нього результат) із доволі значною похибкою. Окрім них існують і точніші методи, проте хоч їх виконання і входить у клас *P*, проте все ж займає значний проміжок часу для достатньо великої вибірки.

Одним з найвідоміших точних методів розв'язання задачі комівояжера є метод гілок та меж, основою якого є пошук маршруту завдяки матриці коефіцієнтів. Даний метод має ряд переваг, а саме, він гарантовано знаходить оптимальний маршрут, тобто це точний метод, як і повний перебір, тільки містить умову відкидання частини розв'язків. За рахунок цього він швидший за повний перебір (завдяки нижнім межам і відсіченню «невигідних» гілок, часто розглядається лише фрагмент усіх можливих шляхів. У кращих випадках – у десятки або навіть тисячі разів швидше за повний перебір). Метод гілок і меж підходить для середніх розмірів задачі (у практиці до 12–14 міст може працювати в реальному часі, при хороших нижніх межах – навіть більше) [6].

Але метод гілок та меж не гарантує швидкий результат у найгіршому випадку, адже при поганому виборі меж або даних він працює майже як метод повного перебору, тобто *O(n!)*). Слід зауважити, що метод потребує складної реалізації, необхідна реалізація зниження матриці, обчислення нижньої межі, черга з пріоритетами, структура вузлів тощо. Він складніший в реалізації ніж жадібний алгоритм або метод повного перебору. Зокрема він чутливий до вибору нижньої межі (якщо нижня межа слабка, тобто погано наближує найменші можливі витрати – відсікання неефективне). Також слід згадати, що він не масштабується до великих задач (для 25+ міст час роботи може бути занадто довгим навіть при хороших оптимізаціях). Але незважаючи на вказані недоліки метод гілок та меж – золотий стандарт для точного вирішення TSP, коли задача невелика або середня за розміром, побудовано матрицю відстаней між містами [6].

Аналіз останніх досліджень і публікацій. Науковці намагалися покращити метод гілок і меж, зокрема шляхом розумнішого розгалуження (вибір вузлів з найкращою потенційною користю), використання сильніших і швидших меж через релаксації та евристики, впровадження жадібних і локальних алгоритмів для швидкого знаходження початкових розв'язків [4], скорочення дерева за допомогою відсікання вузлів і домінування, а також застосування паралельних та розподілених обчислень і комбінування з іншими методами оптимізації, такими як метод відсічних площин або врахування симетрії задачі [3–5]. В кожному з даних випадків покращення однієї характеристики призводило до ускладнення або погіршення інших параметрів [9; 10].

Метою роботи є вдосконалення методу гілок та меж для розв'язання задачі комівояжера за рахунок використання алгоритму Вельцля та обчислення мінімального охоплюючого міста кола для оцінки геометричних меж та побудови евристичного початкового маршруту.

Алгоритм Вельцля використовується для знаходження мінімального кола, що охоплює множину точок на площині. Алгоритм Вельцля – рекурсивний алгоритм для знаходження радіусу мінімального описуючого (охоплюючого) кола, яке накриває всі задані точки. Він працює у середньому за лінійний час. Алгоритм працює «знизу–вгору» рекурсивно: випадково переставляємо точки, потім по черзі «додаємо» точки і підтримуємо мінімальне коло для вже розглянутої підмножини. Якщо нова

точка лежить уже в поточному колі – нічого не змінюємо. Якщо не лежить – ця точка має бути однією з опорних для нового кола; тоді ми шукаємо мінімальне коло для попереднього підмножини з примусовим включенням цієї точки у множину опорних *R*. Оскільки опорних не більше трьох, рекурсія по *R* завершується швидко [7; 8].

Даний алгоритм можна використати як евристику для обмеження пошуку маршруту комівояжера. Мінімальне коло дає верхню межу діаметра області з точками-містами. Це може допомогти в методах відсікання методу гілок та меж чи оцінці початкових маршрутів.

Новизна. Алгоритм Вельцля швидко визначає мінімальне описане коло (MEC, Minimum Enclosing Circle – центр кола *C*, радіус кола *r*) для множини точок-міст. MEC сам по собі не дає жорсткої нижньої межі для довжини оптимального TSP-циклу, але дає корисну геометричну інформацію (просторовий масштаб, центр масового скупчення та діаметр $d = 2r$), яку можна використати як:

- джерело евристики для побудови якісного початкового розв'язку (щоб знизити глобальну верхню межу (найкраще знайдене на даний момент допустиме рішення для задачі). Це допоможе алгоритму не витрачати час на явно гірші гілки і поступово звужує область пошук, тобто посилити відсікання;
- для просторового розбиття (кластеризації) у методі гілок та меж (розгалуження за географічними кластерами). Це дозволить розбити простір можливих рішень на кластери або області, які можна оцінювати окремо, і визначати межі для кожного кластеру;
- для швидких геометричних перевірок/відсікань (наприклад, якщо поєднати з MST, нижнею межею у гілці);
- для прискорення обчислення оцінок, бо радіус *r* визначається лише опуклою оболонкою (внутрішні точки можна тимчасово відкинути).

Постановка задачі. Границі (або межі) у методі гілок та меж – це ключовий інструмент для ефективного відсікання невигідних підзадач. Якість і швидкість роботи методу напряму залежать від того, як добре побудовані ці межі. Розглянемо вузли в методі гілок та меж та оцінимо їх перспективність. Для цього знайдемо нижню межу (*LB*) вартості розширення маршруту. У вузлі є частковий маршрут *P*, що проходить через множину вже відвіданих вершин *V*, решта вершин – це *U*, ще не відвідані.

$$LB = \text{length}(P) + LB_{connect}(V,U),$$

де *length(P)* – вартість уже пройденого часткового маршруту,

$LB_{connect}$ – мінімальна додаткова довжина щоб "покрити" *U* і повернутися до маршруту.

Нижня межа вузла (MST bound) у задачі комівояжера ґрунтується на тому, що будь-який тур повинен утворювати зв'язну структуру: для оцінки будується мінімальне основне дерево на множині ще невідвіданих вершин, після чого до нього додають два найкоротші ребра, що з'єднують дерево з початком та кінцем часткового маршруту; сума вартості вже пройденого шляху, MST та цих двох ребер дає нижню межу, яка обчислюється швидко і є популярною релаксацією у методі гілок та меж.

Запропоновано перед побудовою MST запустити на виконання алгоритм Вельцля над *U*, отримати центр *C* і радіус *r* (MEC). Додавання MEC у нижню оцінку вузла в методі гілок і меж дає сильнішу (тіснішу) нижню межу.· MEC же одразу показує геометричний мінімальний діаметр підмножини точок *U* (рис.1). Якщо всі невідвідані вершини лежать у великому колі радіуса r, то для будь-якого туру, що їх відвідає, довжина маршруту не може бути меншою за *2r* (щоб хоча б перетнути діаметр). Таким чином, додаючи MEC-оцінку, «підтягуємо» нижню межу, змушуючи її враховувати просторове розташування точок.

Оцінка складності. Алгоритм методу гілок та меж із редукцією матриці відстаней на кожному кроці робить редукцію матриці ($O(n^2)$) та вставляє вузол у чергу ($O(\log k)$, де k- кількість вузлів). В гіршому випадку алгоритм пройде всі $n!$ перестановок. Отже, загальна складність становить $O(n!\cdot n^2)$, але в реальності значно менша за рахунок відсікання гілок [1].

Включене в алгоритм обчислення MEC працює в середньому лінійно, $O(m)$ для m точок, де $m = |U|$ – кількість ще не відвіданих вершин у вузлі. Тобто кожен вузол тепер додатково має $O(|U|)$. В гіршому випадку, коли MEC викликається у всіх вузлах, складність становить $O(n!\cdot(n^2+n))=O(n!\cdot n^2)$,·тобто той самий порядок, бо n^2 домінує над n. Теоретично порядок складності не змінюється: і без MEC, і з ним – це $O(n!\cdot n^2)$. Але на практиці MEC значно підсилює нижню межу, що дає більше вузлів на відсікання, а це, в свою чергу, призводить до того, що дерево пошуку суттєво скорочується. Отже, час виконання у середньому значно менший, особливо для «розкиданих» геометрично точок.

Практична реалізація. Для оцінки практичної реалізації запропонованого рішення обрано мову програмування Python та розроблено код для вирішення задачі комівояжера для вибірки міст України на основі їх координат з використанням методу гілок та меж в класичному вигляді та з відсіканням гілок на основі алгоритму Вельцля та обчислення MEC.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

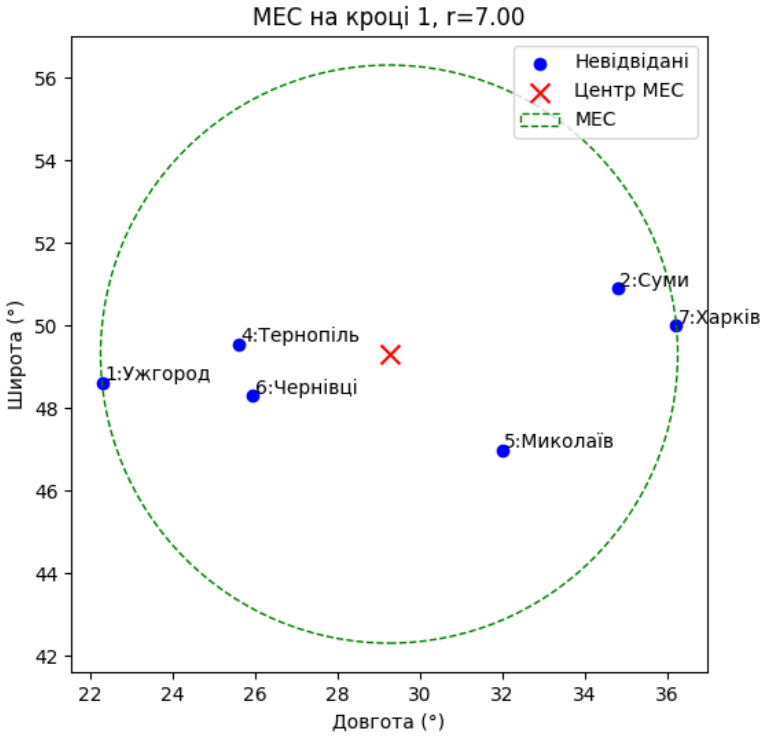


Рис. 1. Візуалізація використання мінімального охоплюючого кола для міст, що входять у нього при певному кроці обчислення на одному з вузлів

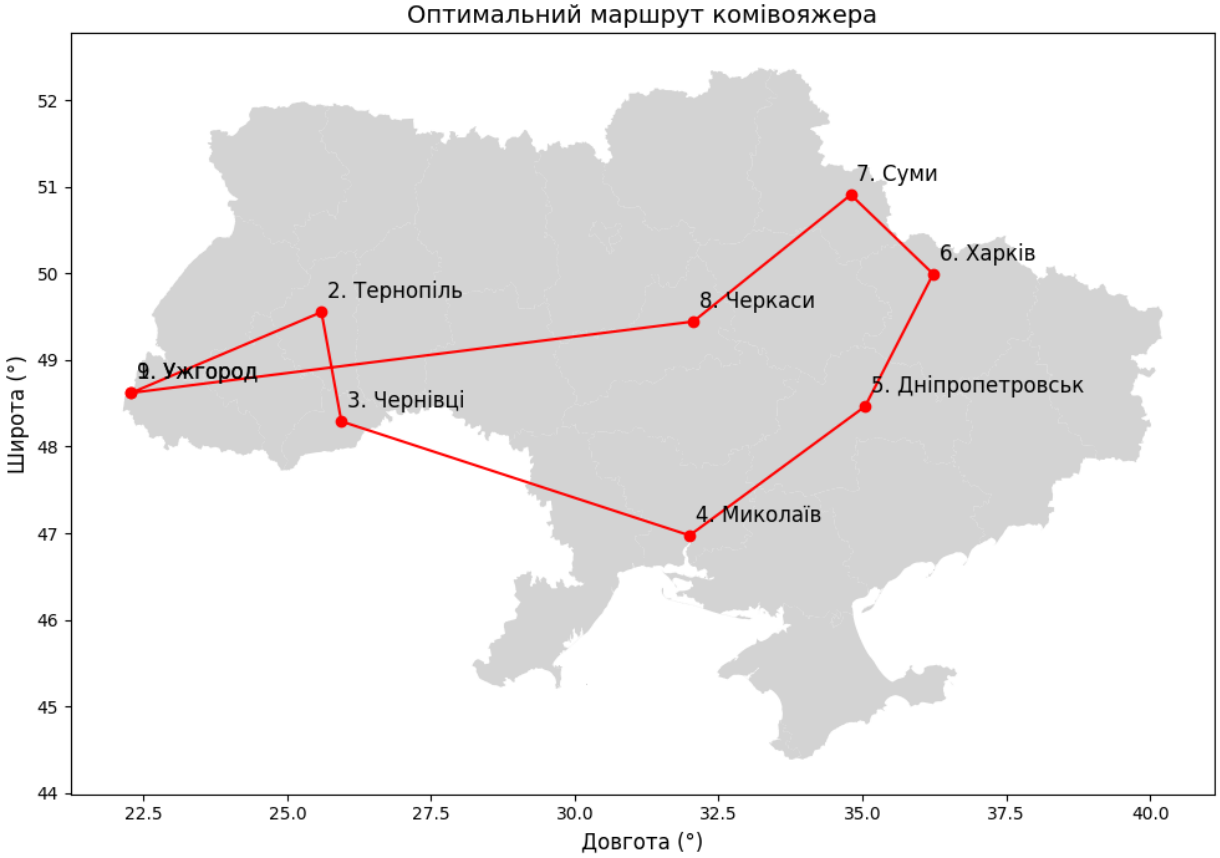


Рис. 2. Графічна візуалізації розв'язку задачі комівояжера для восьми міст України на основі методу гілок та меж із використанням мінімального охоплюючого кола

В (табл. 1) внесено час виконання програм на основі обох кодів: метод гілок і меж, метод з використанням мінімального охоплюючого кола.

Таблиця 1

Час виконання алгоритмів для визначеної кількості міст України

Кількість міст	Час виконання методом гілок та меж (с)	Час виконання із використанням мінімального охоплюючого кола (с)
5	1.40	1.42
6	1.52	1.46
7	1.58	1.56
8	2.31	1.72
9	9.20	3.42
10	74.08	7.62
11	1083.53	26.33

Експериментальні результати показують, що використання мінімального описаного кола (МЕС) у методі гілок та меж для задачі комівояжера практично не впливає на продуктивність при невеликій кількості міст (5–7), де накладні витрати на побудову МЕС співставні з виграшем від відсікання. Однак починаючи з 8–9 міст МЕС забезпечує суттєве скорочення часу роботи алгоритму завдяки ефективнішому відсіканню неперспективних гілок. Для більших задач (10–11 міст) спостерігається експоненційний виграш: час виконання з МЕС зменшується у 10–40 разів у порівнянні з класичним підходом. Це підтверджує доцільність інтеграції МЕС у схему оцінювання нижньої межі як засобу істотної оптимізації пошуку при зростанні розмірності задачі. На (рис. 3) побудовано графік залежності часу виконання програм від кількості міст.

Аналогічно були проаналізовані дані по кількості ітерацій, що здійснював кожен алгоритм для різної кількості міст. Результати показують, що використання МЕС у методі гілок та меж істотно впливає на кількість ітерацій пошуку. Для невеликих задач (5–6 міст) спостерігаються коливання – іноді кількість ітерацій навіть більша, ніж у класичному підході, що пояснюється накладними витратами на перевірку гілок. Однак починаючи з 7 міст МЕС різко зменшує кількість ітерацій: наприклад, при 9 містах від 109 601 зменшено до 2 325, а при 11 містах – від майже 10 млн до лише 17 620. Це демонструє, що МЕС забезпечує значно ефективніше відсікання неперспективних гілок, завдяки чому простір пошуку скорочується на порядки, і метод гілок та меж стає практично застосовним для більших розмірностей задачі.

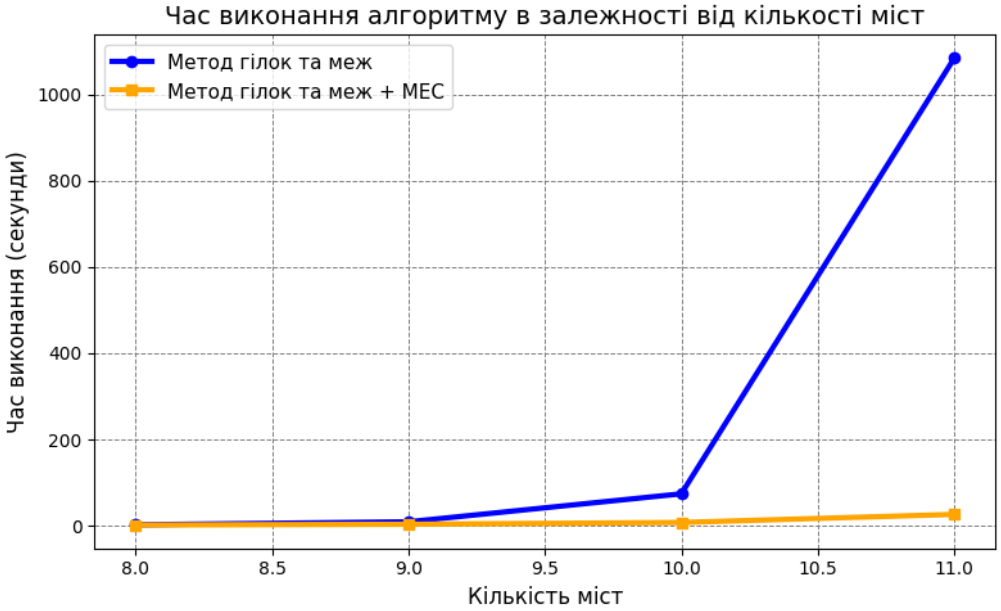


Рис. 3. Порівняльний графік залежності часу виконання від кількості міст методу гілок та меж без використання охоплюючого кола та з ним

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Але слід також відзначити, що алгоритм з MEC стає не абсолютно оптимальний маршрут, а наближений, оскільки додаткова умова відсікання (MEC) іноді «перерізає» гілки, які могли б привести до оптимального розв'язку. Водночас різниця між маршрутами невелика (у межах 3–5 %).·

Висновки. Алгоритм Вельцля є корисним допоміжним інструментом для оцінки геометричних меж або побудови евристичного початкового маршруту. Використання даного алгоритму при розв'язуванні задачі комівояжера полягає у застосуванні методу мінімального охоплюючого кола MEC як евристики для підсилення відсікаючих правил у методі гілок та меж. Алгоритм Вельцля дозволяє за лінійний час побудувати найменше коло, що містить множину невідвіданих міст, і цим сформувати тісну нижню межу для вартості продовження часткового маршруту. Якщо мінімальне коло виходить за межі допустимого діапазону або його радіус гарантує надто велику додаткову відстань, така гілка пошуку відкидається ще до обчислення точних оцінок. Це істотно зменшує розмір дерева пошуку, прискорює знаходження розв'язку та робить можливим обробку більших за розміром задач комівояжера. Починаючи з 8 міст спостерігається суттєве прискорення: при 8 містах час зменшується на ≈25 %, при 9 – на ≈63 %, при 10 – на ≈90 %, а при 11 – більш ніж на 97 %. Отже, MEC неефективний для малих розмірностей, але при збільшенні кількості міст він радикально скорочує час виконання алгоритму, демонструючи виграш на порядки. Також отримані результати свідчать, що використання MEC у методі гілок та меж призводить до зростання сумарної довжини маршруту порівняно з класичним підходом, тобто отримані розв'язки є не абсолютно оптимальними, а наближеними. Різниця у вартості маршрутів становить у середньому 3–5 %, що можна вважати прийнятним відхиленням з огляду на значне скорочення часу роботи алгоритму та кількості ітерацій. Таким чином, MEC виступає як ефективна евристика, яка дозволяє істотно зменшити обчислювальні витрати при розв'язанні задачі комівояжера за рахунок незначної втрати оптимальності.

Список використаних джерел:

1. Bang-Jensen J., Gutin G., Yeo A. When the greedy algorithm fails. *Discrete Optimization*. 2004. Vol. 1. P. 121–127.

2. Bendall G., Margot F. Greedy Type Resistance of Combinatorial Problems. *Discrete Optimization*. 2006. Vol. 3. P. 288–298.

3. Cormen T. H., Leiserson C. E., Rivest R. L., Stein C. Introduction to Algorithms. 2nd ed. MIT Press and McGraw-Hill, 2001. ISBN 0-262-53196-8.

4. Flemming J. A simple linear-time algorithm for the smallest enclosing circle, 2024. URL: https://arxiv.org/abs/2402.17853

5. Formella A., van Leeuwen E. A quasi-linear time heuristic to solve the Euclidean traveling salesman problem. 2024. arXiv:2401.12345. URL: https://arxiv.org/abs/2401.12345

6. Gutin A., Yeo A., Zverovich A. Traveling salesman should not be greedy: domination analysis of greedy-type heuristics for the TSP. *Discrete Applied Mathematics*. 2002. Vol. 117. P. 81–86.

7. Hopcroft J. E., Motwani R., Ullman J. D. Introduction to Automata Theory, Languages, and Computation. 2nd ed. Addison-Wesley, 2000. ISBN 81-7808-347-7.

8. Kiran M. S., Beskirli M. A new approach based on collective intelligence to solve traveling salesman problems. *Biomimetics.* 2024. Vol. 9, No. 3. P. 95. DOI: https://doi.org/10.3390/biomimetics9030095.

9. Puerto J., Valverde-Martín C. The hampered travelling salesman problem with neighbourhoods. *Computers & Industrial Engineering*. 2024. Vol. 189. P. 109120. DOI: https://doi.org/10.1016/j.cie.2024.109120.

10. Smolík M., Vondrák I., Král J. Preprocessing techniques for the smallest enclosing circle problem. *Lecture Notes in Computer Science*. 2022. Vol. 13277. P. 312–324. DOI: https://doi.org/10.1007/978-3-031-10599-0_24.

Дата надходження статті: 07.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

UDC 004.42
DOI https://doi.org/10.32689/maup.it.2025.3.22

Yurii STATYVKA
Candidate of Technical Sciences, Associate Professor,
Associate Professor at the Department of Software Engineering in Energy,
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute",
yu.statyvka@gmail.com
***ORCID:** 0000-0002-7734-953X*

Zhang MINGJUN
Postgraduate Student at the Department of Software Engineering in Energy,
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute",
dora_ZMJ@163.com
***ORCID:** 0000-0002-7189-2881*

SOFTWARE DESIGN TECHNOLOGY FOR AUTOMATING THE PROCESS OF EVALUATING THE LEVEL OF INTERNATIONALIZATION OF SCIENTIFIC INSTITUTION'S ACTIVITIES

Abstract. *The article is devoted to the development of software design technology for automating the process of evaluating the level of internationalization of a scientific institution's activities by developing the IRI-Frame architectural template, which differs from universal templates in that it is focused on the specifics of internationalization assessment.*

The aim of this paper. *Development of software design technology for automating the evaluation of the level of internationalization of scientific institutions' activities based on the IRI-Frame architectural template and system models that support the collection, processing and publication of data for expert decision-making.*

Methodology. *An analysis of the subject area and system requirements was conducted, and an IRI-Frame architectural template was created, focused on the specifics of internationalization evaluation. The static model describes the main artifacts (project, methodology, specification) and their interaction through the Designer, Specifier, Choicer, Estimator, Publisher classes, which inherit the IRIS functionality. The dynamic model reflects scenarios – from the creation and study of methodologies to the selection of the best one and the publication of results. The technology includes three stages: environment definition, requirements formulation, and design of universal and specialized components.*

Scientific novelty. *An IRI-Frame architectural template is proposed, which isolates universal components (support for system creation and evaluation) from special ones (methods of computation, data access, results publication). The combination with GoF patterns provides configuration changes without rewriting the code. The developed static and dynamic models detail the processes of choosing methods of normalization, aggregation, weighting, and statistical analysis.*

Conclusions. *The use of IRI-Frame simplifies the development of internationalization evaluation systems, ensures scalability and integration with various data sources. The technology is suitable for creating separate systems and integrated components. The models reflect the logic of work from project initiation to results publication, forming the basis for further expansion of evaluation methods and optimization of algorithms.*

Key words: *internationalization of scientific institutions, evaluation of the level of internationalization, architecture of the software system, software engineering.*

Юрій СТАТИВКА, Чжан МІНЦЗЮНЬ. ТЕХНОЛОГІЯ ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ АВТОМАТИЗАЦІЇ ПРОЦЕСУ ОЦІНКИ РІВНЯ ІНТЕРНАЦІОНАЛІЗАЦІЇ ДІЯЛЬНОСТІ НАУКОВОЇ УСТАНОВИ

Анотація. *Стаття присвячена розробленню технології проєктування програмного забезпечення для автоматизації процесу оцінки рівня інтернаціоналізації діяльності наукової установи за допомогою розробки архітектурного шаблону IRI-Frame, який відрізняється від універсальних шаблонів тим, що орієнтований на специфіку оцінювання інтернаціоналізації.*

Мета роботи. *Розроблення технології проєктування програмного забезпечення для автоматизації оцінювання рівня інтернаціоналізації діяльності наукових установ на основі архітектурного шаблону IRI-Frame та моделей системи, що підтримують збір, обробку й публікацію даних для прийняття експертних рішень.*

Методологія. *Проведено аналіз предметної області та вимог до системи, створено архітектурний шаблон IRI-Frame, орієнтований на специфіку оцінювання інтернаціоналізації. Статична модель описує основні артефакти (проєкт, методологія, специфікація) та їх взаємодію через класи Designer, Specifier, Choicer, Estimator, Publisher, що успадковують функціонал IRIS. Динамічна модель відображає сценарії – від створення та дослідження методологій до вибору найкращої та публікації результатів. Технологія охоплює три етапи: визначення середовища, формування вимог і проєктування універсальних та спеціалізованих компонентів.*

Наукова новизна. *Запропоновано архітектурний шаблон IRI-Frame, що ізолює універсальні компоненти (підтримка створення та оцінювання систем) від спеціальних (методи обчислень, доступу до даних, публікації*

результатів). Поєднання з патернами GoF забезпечує зміну конфігурацій без переписування коду. Розроблені статична та динамічна моделі деталізують процеси вибору методів нормалізації, агрегування, зважування та статистичного аналізу.

Висновки. *Використання IRI-Frame спрощує розробку систем оцінювання рівня інтернаціоналізації, забезпечує масштабованість і інтеграцію з різними джерелами даних. Технологія придатна для створення окремих систем та інтегрованих компонентів. Моделі відображають логіку роботи від ініціювання проєкту до публікації результатів, формуючи основу для подальшого розширення методів оцінювання й оптимізації алгоритмів.*

Ключові слова: *інтернаціоналізація діяльності наукових інституцій, оцінювання рівня інтернаціоналізації, архітектура програмної системи, інженерія програмного забезпечення.*

Introduction. The concept of internationalization of scientific activity emerged in the last decades of the last century as a result of the process of globalization of the world economy. The concepts of internationalization and globalization are not identical or even synonymous, although they are sometimes used interchangeably. A more balanced approach assumes that internationalization implies the presence of nations and nation-states and their movement towards interaction with other nations, cultures, etc. in the context of scientific activity, while globalization is simply "the flow of technologies, economy, knowledge, people, ideas across borders" [5].

Measuring the level of internationalization of scientific activity is a difficult problem due to the complexity and diversity of the phenomenon itself. This can be a global, national, regional, industry level or institutional.

Problem statement in general. Therefore, given the generally recognized importance of assessing the level of internationalization, the introduction of methods and tools for building reliable software systems for assessing the level of internationalization of scientific institutions is relevant and useful for both research and management applications. the development of a universal software design technology for automating the process of assessing the level of internationalization of the activities of a scientific institution will significantly simplify the process of developing software for the needs of managing the level of internationalization of scientific institutions of a specific community.

Analysis of recent research and publications. It is known [1; 2; 5; 6] that evaluating the level of internationalization of scientific institutions is a non-trivial task both due to the difficulty and complexity of the concept of internationalization itself, and due to its utilitarian nature, which leads to a multiplicity of possible utility functions (indicator systems) depending on the objective environmental conditions and its subjective vision by scientific communities.

At the same time, the analysis of the process of evaluating the level of internationalization makes it possible [9] to build its generalized model and identify functional requirements for a software system for its automation.

The analysis of the process of evaluating the level of internationalization highlights the fact that they are uncertain in advance and may change over time, in particular:

– the list and composition of data sources, formats and data access mode;
– methods:
• data normalization;
• data aggregation, indices and measurements;
• formal, in particular statistical, research of data and assessment results;
• ordering of scientific institutions;
• adoption of an expert decision on compliance with the established quality criteria;
– location, operating environment, presentation format, method of placement of materials intended for publication.

The aim of this paper is to determine the main stages of software design technology for automating the process of evaluating the level of internationalization of scientific institutions based on the proposed architectural template and software system models.

Presentation of the main material.

Structure of the architectural design template. Unlike universal design templates [3; 4; 7; 8; 10], which simplify the development of arbitrary software, the proposed architectural template [8; 11] IRI-Frame, the structure of which is presented in Figure 1, is aimed at solving the problem of choosing the architecture of a software system to automate the process of evaluating the level of internationalization of a scientific institution.

As can be seen from Figure 1, the IRI-Frame design template has the following structural components:

1. Expert is a class that provides user interaction with the IRIS class.

2. The IRIS class interacts simultaneously with the user, data provider, method provider, and publisher. It provides execution of all subprocesses of the internationalization level evaluation process by applying the

methods defined in it to the objects-properties of the DataProvider, MethodProvider, and PublicationMode types.

3. Data provider (DataProvider) is an interface for interacting with possible data sources. Its methods are available to the IRIS class and its descendants. The user of this system will be able to work with the data sources DataSource1, ProjectA and Specification25.

4. Concrete data providers (Concrete DataProvider) implements the DataProvider interface, providing interaction with individual concrete data sources.

5. Method provider (MethodProvider) is an interface that declares access to numerical and other computational methods. Its methods are available to the IRIS class and its descendants.

6. Concrete method providers (Concrete MethodProvider) implement the MethodProvider interface, providing access to concrete computational methods. Among the computational methods available to the user are the data normalization methods RankingNormalise, zScoreNormalise and MaxMinNormalise. The aggregation methods available are AdditiveAggregation, GeometricAggregation and MCAAggregation. The weighting methods are PCAWeighting. The statistical methods are CorrelationAnalysis.

7. Publisher (PublicationMode) is an interface that declares the publication of evaluation results. Its methods are available to the IRIS class and its descendants.

8. Concrete publishers (Concrete PublicationMode) implement the PublicationMode interface, providing publication in specific locations, operating environments, representation formats, etc. The user can publish the evaluation results in one of two modes – for viewing and with editing of test data (ViewMode and EditDataMode respectively).

Thus, the IRI-Frame architectural template provides work with different data sources, the use of various computational methods and the publication of results in different ways, and therefore allows you to use the same code base to create various system configurations. It should also be noted that the IRI-Frame architectural template can be implemented by combining universal GoF templates [1; 4], for example, the Bridge structural template to separate the abstraction of the index aggregation method from the aggregation algorithm, or the Adapter – eliminate the problem of processing heterogeneous data.

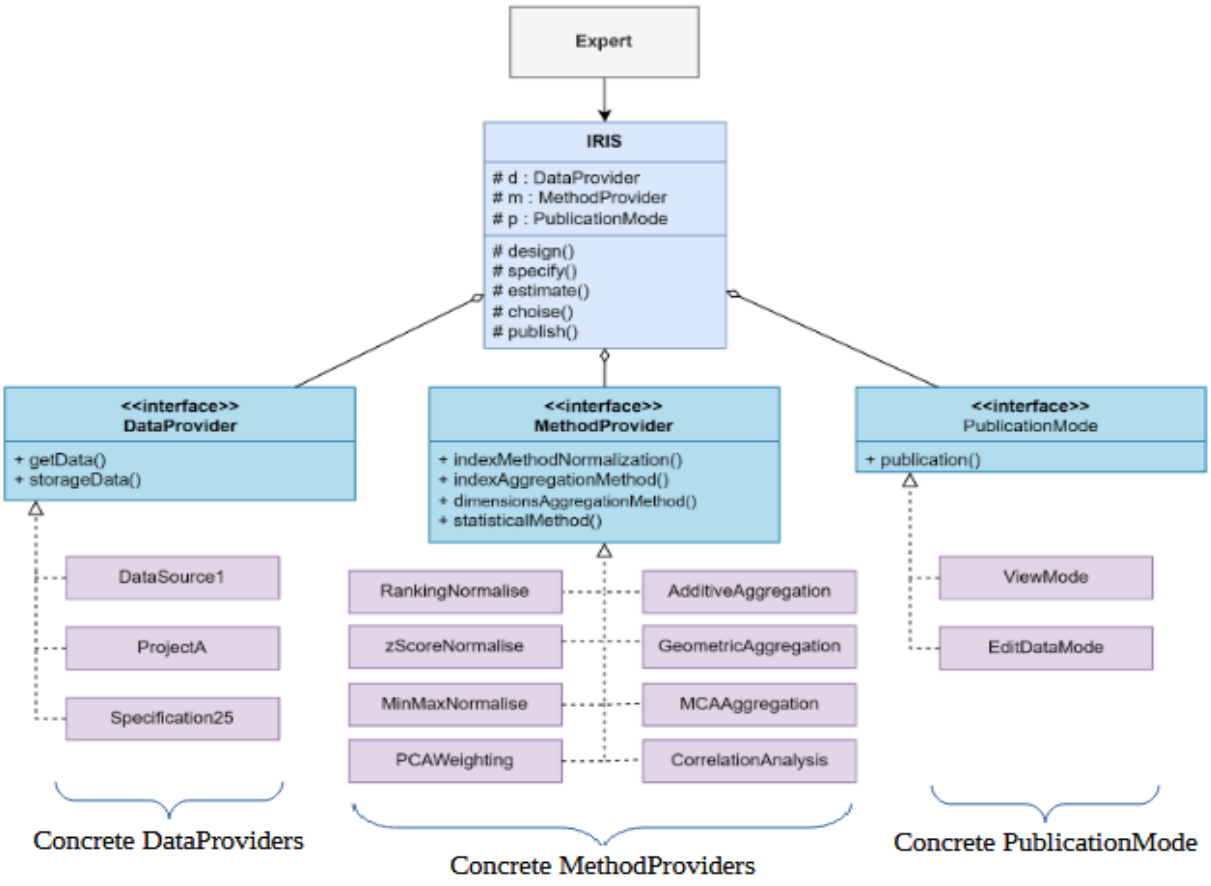


Fig. 1. Structure of the IRI-Frame design template

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Software system models for automating the process of evaluating the level of internationalization. The processes and boundaries of the software system for evaluating the internationalization of scientific institutions [9] identified as a result of the analysis of the subject area allowed us to formulate requirements for it and propose a basic architecture with the separation of the structure at the level of modules and components [4].

The developed architectural design template allows us to separate the general functionality of the use cases from specific data, specific calculation and publication methods.

Comparing the structure of the basic architecture and the structure of the architectural design template, we come to the conclusion that it is necessary to further detail the representations of the software system.

Static model of a software system. The main operational functionality of the software system, as shown in (Fig. 1), in the architectural design pattern is concentrated in the IRIS class. However, given the list of use cases, such a concentration of functions is excessive, which requires decomposition of the IRIS class.

Figure 2 presents a class diagram as a static software model. The IRIS base class aggregates instances of classes-implementations of the DataProvider, MethodProvider and PublicationMode interfaces and provides the constructors conDesigner(), conSpecifier(), conEstimator(), conChoicer(), conPublisher() to create objects, respectively, of the Designer, Specifier, Estimator, Choicer and Publisher classes. These classes are descendants of the IRIS class and provide the functionality of the software system for automating processes:

- Designer – designing a system for evaluating the level of internationalization of a scientific institution;
- Specifier – researching the evaluation system;
- Choicer – choosing the methodology that best meets the quality requirements;
- Estimator – using the approved specification to assess the level of internationalization;
- Publisher – publishing the evaluation results.

Descendant classes inherit members of the ancestor class and may have additional ones.

Thus, the Designer class defines, in particular, methods for creating a Project object, saving it in the data warehouse and reading from the warehouse – createProject(): Project, saveProject() and getProject(): Project, respectively.

In the Project class, the object of which is aggregated by an instance of the Designer class, the methodology constructor createMethodology() and the list of created methodologies: List<Methodology> are defined.

The Specifier and Choicer classes use Project to access the defined methodologies in order to, respectively, specify them (Specifier) and select the one that best meets the quality criteria (Choicer).

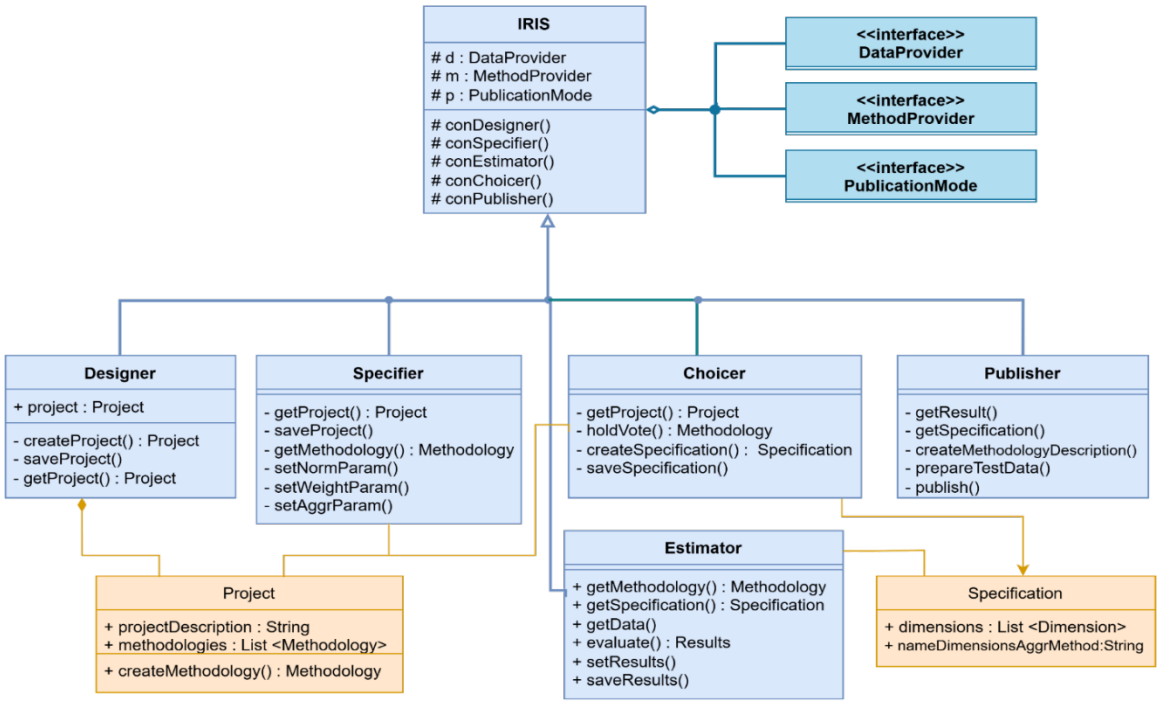


Fig. 2. Diagram of basic classes of the software system for evaluating the level of internationalization of a scientific institution

The methodology that was recognized as the best receives the status of a specification, becomes an independent artifact – Specification, the objects of which are stored in the data warehouse. The Specification class is structurally identical to the Methodology class. However, the Methodology instance, firstly, is not an independent artifact, but only a component of the Project composite, and secondly, it may be unspecified or partially specified. Instead, an instance of the Specification class is created as a clone of a fully specified methodology.

The classes-artifacts Project and Specification and their relationships with other classes are highlighted in beige on the diagram. The arrow on the association relationship from the Choicer class emphasizes the exclusive way to create a new instance of the Specification class.

The Specifier class contains, in particular, methods for reading and saving a possibly modified or newly defined project getProject() and saveProject(), methods for accessing project methodologies, in particular getMethodology(), and methods for setting parameters of normalization, weighting (weights) and aggregation methods – setNormParam(), setWeightParam() and setAggrParam(), respectively.

The Choicer class contains methods for reading the project getProject(), recognizing one of the studied methodologies as the best – holdVote(): Methodology, creating a specification and saving it in the data store – createSpecification(): Specification and saveSpecification() respectively.

The Estimator class defines methods for accessing the methodology, specification and data – getMethodology(), getSpecification() and getData() respectively, calculating the results evaluate(): Results and saving them setResults() in the methodology class or in the data store saveResults().

The Publisher class defines methods for accessing the results of the level evaluation getResult() and specification getSpecification(), as well as creating a methodology description, preparing a publication with test data and, in fact, publishing createMethodologyDescription(), prepareTestData() and publish().

Figure 3 presents the hierarchy of classes for representing Project and Specification artifacts. The diagram shows that the Project composite contains zero or more methodologies and a project description, which can be used to supplement the methodology description in an instance of the Specification class. The Methodology and Specification classes are identical in structure, as can be seen from the diagram.

The DataResult type depends on the structure of the internationalization level evaluation created during the operation of the software system, so it is not defined here. The same is true for the Parameters type, which depends on the chosen computational methods.

The Methodology and Specification classes contain a description of the methodology, a list of dimensions, names and parameters of the selected aggregation method for the dimensions.

The Dimension class contains the name and description of the dimension, a list of the measures that define it, and the method and parameters of the aggregation method for the measures.

The Index class contains the name and description of the measures, a list of paths to the data with the values of the measures, and the method and parameters of the normalization method for the measures.

Dynamic model of the software system for evaluating the level of internationalization of scientific institutions. To represent the temporal deployment of the functioning of the software system and the interaction of the selected classifiers, interaction diagrams are used, one of the varieties of which is a sequence diagram.

The diagram in (Fig. 4), presents a generalized software system sequence diagram. The diagram shows that, in the general case, the execution of the methods of each object is initiated by the system user. Thus, by contacting an object of the Designer class, the user can initiate the creation of a new project, and then proceed to the specification and study of each existing (created) methodology in it by contacting an instance of the Specifier class. After creating the project, the Designer and the Specifier, after specifying or studying the methodologies available in it, save the project in the data warehouse.

The Specifier specifies the methodologies created by the Designer and contacts the Estimator to perform calculations within the scenarios for determining the quality indicators of each of them.

An instance of the Choicer class, comparing the quality indicators of the specified project methodologies and, possibly, taking into account expert opinions on compliance with quality criteria, internationalization policy and strategy, chooses one of them, which is stored as a Specification. In the future, the Specification is used to calculate the level of internationalization of scientific institutions, until a decision is made to modify it or develop a new project.

Estimator, to determine the level of internationalization in accordance with the existing specification, loads the Specification, performs the evaluation and stores the results in the data warehouse, and references to them (with a time stamp) in the specification.

The Publisher class provides for loading the specification, and therefore the results of the internationalization level evaluation, generates a description of the evaluation methodology, and publishes the named materials.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

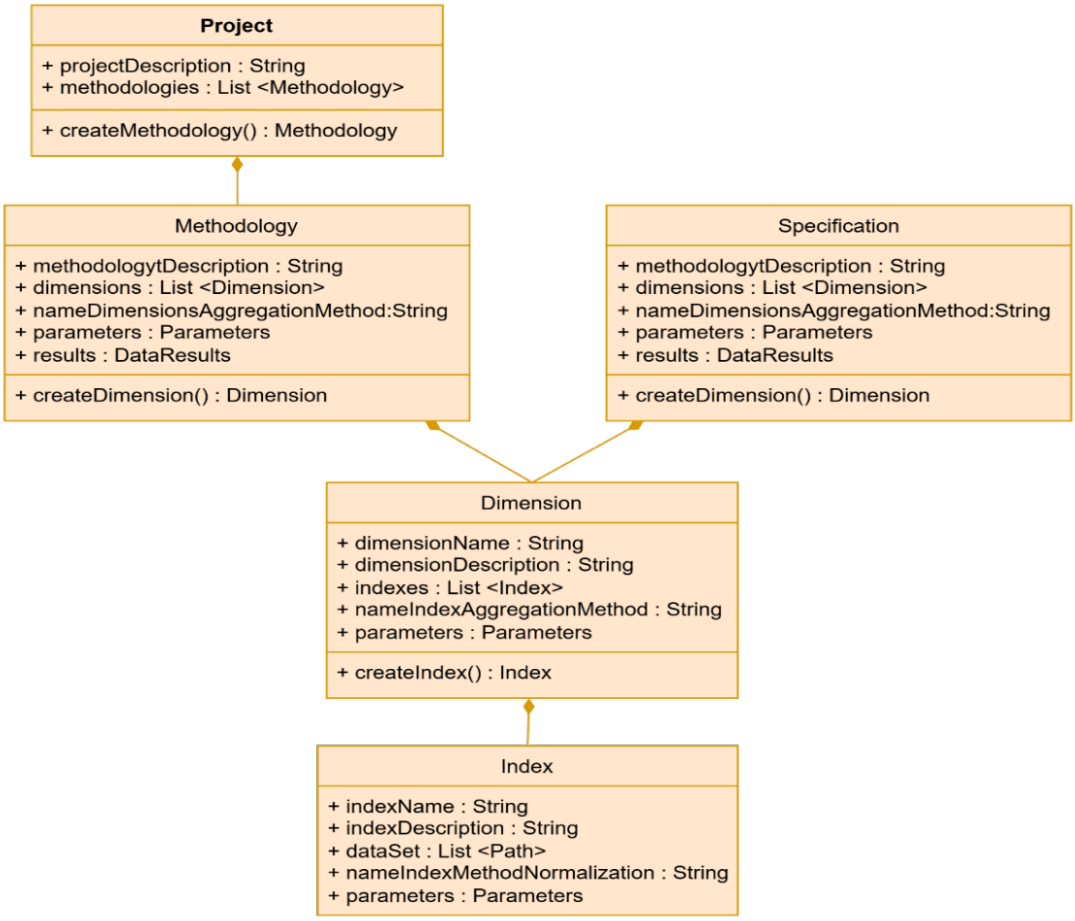


Fig. 3. Structure of the main artifact classes

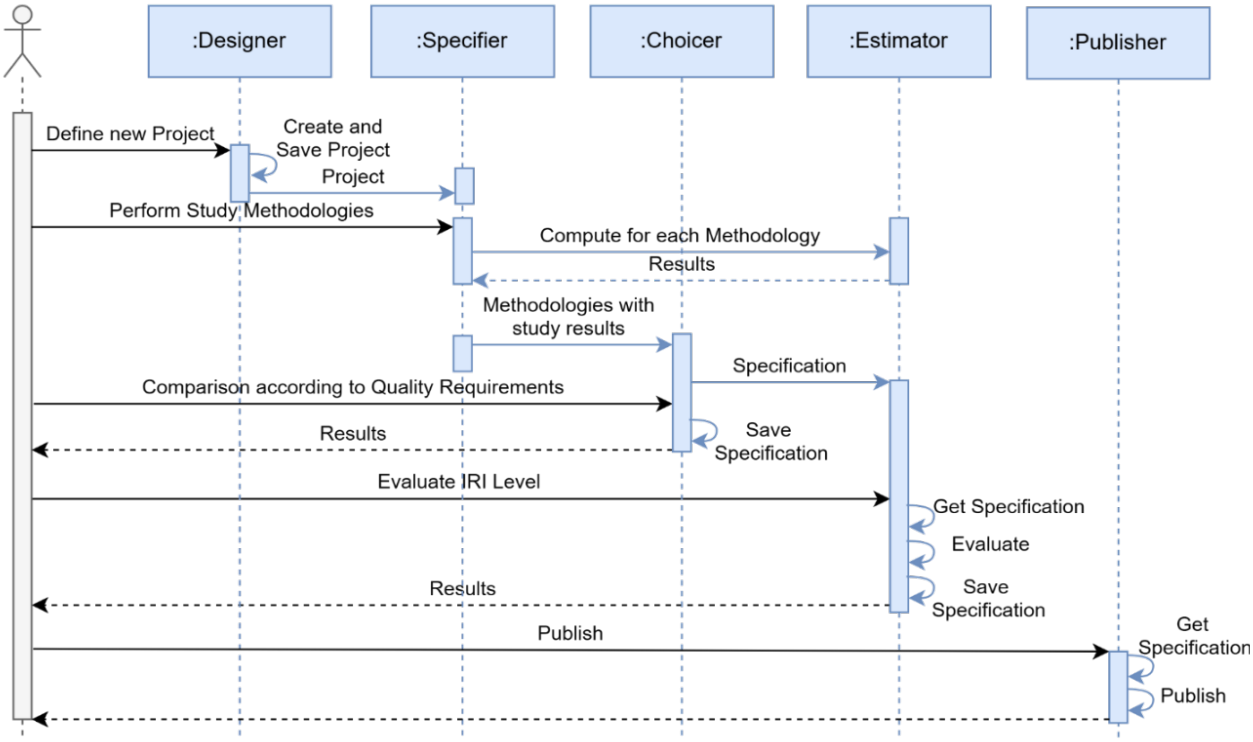


Fig. 4. Software system sequence diagram

Technology for designing and modifying software tools to automate the process of evaluating the level of internationalization. Software tools for automating the process of evaluating the level of internationalization of scientific institutions can be implemented both in the form of a separate system and in the form of a component of another system.

The life cycle of a software system, as is known, involves iterative processes of its development and modification, but does not exclude the identification of certain stages, each of which, in the general case, can be performed repeatedly.

Let us consider the main (enlarged) stages of designing a separate software system for automating the process of evaluating the level of internationalization of scientific institutions, provided that there is a formal or informal interested scientific community.

The first stage involves determining the organizational and operational environment of the software system, in particular:

– delineation of the boundaries of the system and its surroundings, as well as their (potential) representatives – experts (Experts), managers (IManager), public users and, possibly, other categories of actors;

– determination of the place of the software system in the system of activity of actors and the scientific community.

Determining these aspects allows us to formulate requirements for the system for evaluating the level of internationalization of scientific institutions in order to adapt the basic architecture proposed here to the needs of users.

The second stage involves identifying high-level representations of the substantive component of the system's functionality, in particular:

– the presence or formulation of documented interpretations of the concepts of internationalization of scientific institutions and the level of internationalization. Such documents are usually the Internationalization Policy and/or the Internationalization Strategy;

– the presence of a documented list and interpretations of the quality criteria for the system for evaluating the level of internationalization.

The specified high-level positions allow to define certain requirements for the software system and its information component:

– requirements for the necessary source information on the activities of scientific institutions. This allows to draw a conclusion about the possibility of using existing, possibly external, sources or the need to develop a separate subsystem for collecting relevant data;

– methods of evaluating the level of internationalization of scientific institutions. This allows to limit the set of methods of normalization, aggregation and ordering of data, as well as the necessary set of statistical methods.

– requirements for methods of presenting the results of the evaluation and description of the methodology, evaluation. This allows to limit the set of methods for presenting materials to be published.

The third stage involves designing the components of the software system:

– universal components of the software system that provide support for the processes of creating a system for evaluating the level of internationalization of scientific institutions and performing such an evaluation;

– special components that implement such computational methods, such as: methods of normalization and aggregation of data, weighting methods, ordering methods, statistical methods, etc. Special components should also include components that provide data access mechanisms and publication mechanisms.

When developing a software system to automate the process of evaluating the level of internationalization of scientific institutions, it is advisable to use the IRI-Frame architectural design template proposed in this study.

Conclusions. The results presented in this article on the research process of designing software for automating activities aimed at evaluating the level of internationalization of scientific institutions allow us to formulate the following conclusions:

1. The architectural design template has been developed that provides isolation of the code of universal components that provide support for the processes of creating a system for evaluating the level of internationalization from special components that implement computational methods, data access methods and methods related to the publication of materials.

2. The static model of a software system has been developed using the developed architectural design template, which contains behavioral classes, abstract classes (interfaces) and essential classes for the main artifacts of the software system.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

3. The dynamic model of interaction of software entities of the system for the main use cases of the system has been developed.

4. The main stages of the technology of designing software tools for automating the process of evaluating the level of internationalization of scientific institutions are proposed.

5. All the results presented in this section are aimed at simplifying the process of developing software for the needs of managing the level of internationalization of scientific institutions of a specific community.

Bibliography:

1. Bas M. C., Boquera M., Carot J. M. Measuring internationalization performance of higher education institutions through composite indicators, INTED, 2017 Proceedings, 2017. pp. 3149–3156. DOI: 10.21125/inted.2017.0815.

2. Brandenburg U., and G. Federkeil. "How to Measure Internationality and Internationalisation of Higher Education Institutions. Indicators and Key Figures." 2007. (Accessed February 09, 2024). URL: https://www.che.de/en/download/how_to_measure_internationality_ap_92-pdf (open in a new window).

3. Freeman E., Robson E. Head first design patterns. O'Reilly Media, 2020, p. 669.

4. Gamma E., Helm R., Johnson R., and Vlissides J. Design Patterns. Reading, MA.: Addison-Wesley, 1995.

5. Knight J. Monitoring the Quality and Progress of Internationalization. *Journal of Studies in International Education*, 2001. 5(3), 228–243. DOI: 10.1177/102831530153004.

6. Knight Jane. Internationalization Remodeled: Definition, Approaches, and Rationales. *Journal of Studies in International Education.* 2004. 8. 5–31. DOI: 10.1177/1028315303260832.

7. Larman C. Applying UML and patterns: an introduction to object-oriented analysis and design and iterative development. Pearson Education India, 2012. p. 703.

8. Mayvan B. B., Rasoolzadegan A., Yazdi Z. G. The state of the art on design patterns: A systematic mapping of the literature. *Journal of Systems and Software*, 2017. Vol. 125, P. 93–118.

9. Statyvka Y. I., Nedashkivskiy O. L., Mingjun Z. Model of the process for evaluating the level of internationalization of the scientific institution activities. *Connectivity,* No. 3 (175), 2025, pp. 42–51. DOI: 10.31673/2412-9070.2025.020915.

10. Wedyan F., Abufakher S. Impact of design patterns on software quality: a systematic literature review. *IET Software*, 2019. Vol. 14, Issue 1, P.1–17. DOI: 10.1049/iet-sen.2018.5446.

11. Дичка І. А., Сулема О. К., Крайносвіт А. А. Програмна система логістичного обліку на основі дворівневого штрихового коду. Системні технології, 2020. № 6, С. 28–38.

Дата надходження статті: 18.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.9: 303.732.4
DOI https://doi.org/10.32689/maup.it.2025.3.23

Олександр ТЕРЕНТЬЄВ
доктор технічних наук, доцент,
провідний науковий співробітник відділу прикладної інформатики,
Інститут телекомунікацій і глобального інформаційного простору НАНУ,
o.terentiev@gmail.com
***ORCID:** 0000-0002-4288-1753*

Кірілл БЕДЛІНСЬКИЙ
здобувач вищої освіти, Інститут прикладного системного аналізу,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»,
bedlinskyi.kirill@outlook.com
***ORCID:** 0009-0000-1630-3063*

Володимир ДУДА
аспірант,
Інститут телекомунікацій і глобального інформаційного простору НАН України,
dudavolodimir@gmail.com
***ORCID:** 0009-0002-4278-4635*

Михайло СТОЛЯР
аспірант, Інститут прикладного системного аналізу,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»,
misha.stolyar99@gamil.com
***ORCID:** 0009-0009-3624-3147*

МЕТОДИКА СИСТЕМНОГО АНАЛІЗУ ДЛЯ ТОРГІВЛІ ФІНАНСОВИМИ АКТИВАМИ ІЗ ВИКОРИСТАННЯМ ТЕХНІЧНИХ ІНДИКАТОРІВ У МОДЕЛЯХ МАШИННОГО НАВЧАННЯ

Анотація. *Стаття присвячена розробці методики системного аналізу, що складається з дев'яти кроків, для торгівлі фінансовими активами. Ця методика включає етапи підготовки даних для аналізу, побудови математичних моделей та аналізу результатів тестування. Особливістю методики є використання таких технічних індикаторів, як смуги Болліджера, стохастичний осцилятор та параболічний індикатор зупинки і розвороту.*

Мета статті. *Розробити методику системного аналізу для торгівлі фінансовими активами.*

Методологія. *На основі запропонованої методики системного аналізу, було реалізовано комп'ютерну програму. Із використанням цієї програми було проведено низку обчислювальних експериментів на реальних статистичних даних, що дозволило порівняти використання таких технічних індикаторів фінансового ринку, як смуги Болліджера, стохастичний осцилятор та параболічний індикатор зупинки і розвороту, при розробці моделей машинного навчання для прогнозування динаміки цін.*

Наукова новизна. *Представлено покрокову методику системного аналізу для торгівлі фінансовими активами. Запропоновану методику реалізовано у вигляді комп'ютерної програми. Виконано аналіз та порівняння використання різних технічних індикаторів фінансового ринку на реальних статистичних даних.*

Висновки. *Було з'ясовано, що при використанні різних технічних індикаторів для математичної моделі у вигляді випадкового лісу рішень, найкращі результати прогнозування показую стохастичний осцилятор, після нього за отриманими результатами моделювання йдуть смуги Болліджера, а найгірший результат надала модель із використанням індикатора зупинки і розвороту.*

Ключові слова: *машинне навчання, технічні індикатори, криптовалютний ринок, задачі класифікації.*

Oleksandr TERENTIEV, Kirill BEDLINSKYI, Volodymyr DUDA, Mykhailo STOLIAR. A SYSTEM ANALYSIS METHODOLOGY FOR TRADING FINANCIAL ASSETS, USING TECHNICAL INDICATORS IN MACHINE LEARNING MODELS

Abstract. *The article is devoted to the development of a nine-step system analysis methodology for trading financial assets. This methodology includes the stages of preparing data for analysis, building mathematical models, and analyzing test results. A feature of the methodology is the use of such technical indicators as Bollinger bands, the stochastic oscillator, and the parabolic stop and reversal Indicator.*

The goal. *Develop a systematic analysis methodology for trading financial assets.*

The methodology. *Based on the proposed system analysis methodology, a computer program was developed. Using this program, a number of computational experiments were conducted on real statistical data, which allowed us to compare the use of such technical financial market indicators as Bollinger bands, stochastic oscillator, and parabolic stop and reversal indicator in the development of machine learning models for forecasting price dynamics.*

The scientific novelty. *A step-by-step methodology for system analysis for trading financial assets is presented. The proposed methodology is implemented in the form of a computer program. The analysis and comparison of the use of various technical indicators of the financial market on real statistical data is performed.*

Conclusions. *It was found that when using various technical indicators for a mathematical model in the form of a random forest, the best forecasting results are shown by the stochastic oscillator, followed by the Bollinger Bands according to the obtained modeling results, and the worst result was provided by the model using the stop and reverse indicator.*

Key words: *machine learning, technical indicators, cryptocurrency market, classification tasks.*

Постановка проблеми. За даними агрегатору статистичних даних MACROMICRO (https://en.macromicro.me) [14], станом на липень 2025 року загальний обсяг ринку криптовалют становить 3,8 триліонів доларів США, згідно із даними спеціалізованих аналітичних сервісів 33CoinMarketCap (https://coinmarketcap.com) та CoinGecko (www.coingecko.com). Об'єм ринку криптовалют стрімко зростає та набуває популярності, і вже зрівнявся з об'ємом фондового ринку таких країн як Великобританія (3,54 трл. доларів США), Франція (3,42 трл. доларів США) та Німеччини (2,94 трл. доларів США), і у десять разів перевищив капіталізацію глобального ринку дорогоцінних металів (золото, срібло, платину та інші) який становить за оцінками експертів 300-500 млрд. доларів США [10].

Однією з ключових рис криптовалютного ринку є висока волатильність. Щоденні коливання цін криптовалют з найбільшою капіталізацією можуть перевищувати 5-10 %, а іноді становлять й двозначні значення. Це створює як додаткові можливості для інвесторів, так і додаткові ризики. Все це у сукупності і визначило тематику дослідження, щодо визначення надійних інструментів аналізу ринку.

Аналіз останніх досліджень і публікацій. В роботі [5] від 2025 року, авторами було виконано дослідження 88 технічних індикаторів та їхню корисність у прогнозуванні цін акцій. Зазначені технічні індикатори подавалися як вхідні дані для методів XGBoost, випадкового лісу рішень (Random Forest), машини опорних векторів (Support Vector Regression) та рекурентних нейронних мереж (LSTM). Для моделювання використовувалися дані фондового індексу S&P 500, а для зменшення розмірності простору вхідних даних, використовувався метод головних компонентів (PCA).

Окрім того, в роботі [3], також від 2025 року, іншою групою дослідників, було досліджено вплив наборів технічних індикаторів – смуги Болліндже́ра, ковзні середні, числа Фібоначі в задачах побудови моделей машинного навчання. На хвилинних даних індексу SPY було побудовано та оцінено 13 моделей. Аналіз важливості ознак показав, що первинні цінові ознаки часто перевершують технічні індикатори, що свідчить про їх обмежену корисність у контексті високочастотної торгівлі.

В статті [16] від 2024 року, зроблено великий огляд з 241 літературного джерела, що присвячені задачам глибокого навчання, саме для ринку криптовалют, включаючи використання різноманітних технічних індикаторів та додаткових ознак аналізу. Розглянуто також різні завдання моделювання, включаючи прогнозування цін, побудову портфеля, аналіз шахрайства, аномальну торгівлю, регулювання торгівлі та первинне розміщення монет у криптовалюті (ICO – Initial Coin Offering).

Методика дослідження. В рамках проведеного дослідження, пропонується оригінальна методика системного аналізу для торгівлі фінансовими активами, що складається з дев'яти кроків. Ця методика дозволяє будувати аналітичні моделі, а також впроваджувати їх в експлуатацію, із використанням технічних індикаторів. На основі цієї методики було реалізовано комп'ютерну програму [1] та виконано відповідні обчислювальні експерименти.

Крок 1. Збір та підготовка даних для аналізу.

В якості джерел даних можуть використовуватися різноманітні постачальники даних, власноруч написані api-програми, а також готові бази даних. Зазначимо, що вартість покупки готових наборів даних, у різних агрегаторів даних, може варіюватися від декількох сотень доларів до багатьох тисяч, в залежності від довжини історії, типу та повноти даних, саме тому написання власних api-програм – суттєво заощаджує бюджет, при проведені відповідних досліджень.

Цей крок також включає, вирішення задач попереднього аналізу щодо якості отриманих даних, за необхідності відновлення пропусків, аналіз аномальних значень та їх згладжування або взагалі видалення з аналізу [6].

Крок 2. Обчислення технічних індикаторів на основі даних, що було отримано з попереднього першого кроку. Для цього використовуються історичні дані про ціни – Open, High, Low, Close, Volume, на основі яких будуються різноманітні індикатори, як то:

1) індикатор волатильності – смуги Болліндже́ра [3; 5; 6];

2) стохастичний осцилятор [6];

3) параболічний індикатор зупинки і розвороту [6];

4) трендові індикатори (зазвичай їх отримують на основі ковзних середніх). Один з таких найбільш розповсюджених індикаторів – MACD (Moving Average Convergence/Divergence) [3; 5; 6];

5) індикатори об'єму торгів. Наприклад, OBV-індикатор (On Balance Volume) [6].

Крок 3. Розробка додаткових ознак аналізу [3; 6; 16], що будуть використовуватися як регресори в моделі, на основі даних та інформації яка вже є в наявності з попередніх кроків:

1) лагові значення цін та індикаторів (наприклад, зі зміщенням у часі на 1, 2, 3, 4 лагових значення);

2) різницеві значення індикаторами (різниця між поточним та попередніми значеннями);

3) бінарні змінні, що описують деякий стан або рівень ціни. Наприклад, змінна приймає значення одиниця, якщо обсяг торгів за останню годину більший за деяке задане порогове значення.

Процес розробки додаткових ознак аналізу, відомий серед фахівців під назвою – feature engineering.

Крок 4. Формалізація цільової змінної [6]. Це може бути як інтервальна змінна, для прогнозування значення ціни, на наступному кроці, так і номінальна або ординарна змінна, коли спочатку визначають кількість потенційних станів. Наприклад, можуть бути наступні три стани цільової змінної – ціна іде вгору, ціна іде вниз, або ціна залишається майже на тому самому рівні. Саме такий варіант формалізації цільової змінної було обрано в цьому дослідженні.

Крок 5. Побудова прогнозної моделі. На цьому кроці використовуються методи машинного навчання, це можуть бути логістична регресія, дерева рішень, XGBoost, машина опорних векторів або нейронна мережа. В проведеному дослідженні було використано досить популярний підхід математичного моделювання – випадковий ліс рішень (Random Forest) [5; 3; 16; 6].

Крок 6. Аналіз отриманих результатів та оцінка прогнозної точності моделі. В залежності від типу цільової змінної (інтервальна, номінальна або ординарна) можуть використовуватися різні метрики як то:

1) середньоквадратична похибка (RMSE) [5];

2) середня абсолютна похибка в процентах (MAPE) [5];

3) коефіцієнт детермінації (R2);

4) F1-оцінка (F1-score);

5) оцінка середньої точності (Average Precision score) [6];

6) частка прибуткових угод;

7) середній дохід з однієї прибуткової угоди;

8) середні втрати з однієї збиткової угоди.

Крок 7. Тестування моделі на нових даних, які не використовувалися для навчання. Фактично на цьому кроці імітується торгівля за розробленою моделлю, з метою визначення її прибутковості за історичними даними. Отримані результати порівнюються з якоюсь еталонною стратегією або прибутковістю портфелю акцій за індексами бірж, як то S&P 500 або NASDAQ. На цьому кроці, на основі отриманих результатів, обчислюється такі показники як [5]:

1) середня кількість угод за весь період, за місяць, протягом доби;

2) максимальний прибуток, за вказаний період аналізу, та періоді тестування моделі;

3) максимальне просідання портфеля, за весь період аналізу, та періоді тестування моделі [3];

4) максимальне просідання портфеля протягом доби [3];

5) коефіцієнт Шарпа [3].

Зауважимо, що коефіцієнт Шарпа [3] спирається на кілька ключових припущень, які в окремих випадках можуть обмежувати його корисність. Ця метрика оптимально інтерпретується за умови симетричного розподілу надлишкових дохідностей; при асиметрії, що часто спостерігається в аналізі цін на криптовалюти, використання стандартного відхилення може призводити до недооцінки або переоцінки ризиків [13]. Також коефіцієнт Шарпа трактує волатильність як лінійну міру ризику, однаково враховуючи підйоми й просідання вартості активів, тоді як інвестор зазвичай більше турбується саме про втрати.

Крок 8. Побудова аналітичних звітів для осіб, що приймають рішення (ОПР). Зазвичай такі звіти містять інформативні та зрозумілі графіки фактичного та прогнозного курсу ціни, сигналів відкриття та закриття угод на купівлю та продаж, статистичні характеристики впливовості ознак на цільову змінну.

Крок 9. Впровадження розробленої математичної моделі у вигляді коду програми або детально описаного технічного завдання, в існуючу технічну інфраструктуру компанії, інтеграція з торговими

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

платформами (для отримання даних в режимі реального часу та виставлення ордерів), налаштування аналітичних графіків та звітів.

На (рис. 1) наведена структурна схема запропонованої методики системного аналізу, що описана у вигляді дев'яти кроків аналітичного процесу.

Теоретична частина дослідження. В рамках проведеного дослідження, було розглянуто технічні індикатори – смуги Болліндже́ра [2; 3], стохастичний осцилятор та параболічний індикатор зупинки і розвороту (Parabolic SAR – Stop and Reverse). Всі ці індикатори належать до різних категорій і забезпечують різні аспекти аналізу крипторинку. Смуги Болліндже́ра відображають волатильність та можливі екстремальні відхилення ціни, стохастичний осцилятор вимірює відносну позицію ціни в діапазоні для прогнозування розворотів, а параболічний індикатор зупинки і розвороту сигналізує про напрям і зміну тренду [2].

Смуги Болліндже́ра. Смуги Болліндже́ра – це популярний індикатор волатильності, розроблений Джоном Болліндже́ром [2; 3]. Він складається з трьох смуг: середньої, верхньої та нижньої. Середня смуга зазвичай є простою ковзною середньою ціни за певний період. Верхня та нижня смуги відстають від середньої на певну задану кількість стандартних відхилень ціни (як правило, два стандартних відхилення) [4]. Формально центральна смуга визначається наступним чином:

$$M_t = SMA_n(P_t),$$

де *SMA* – проста ковзна середня ціни P за n періодів. SMA – це скорочення від Simple Moving Average. Нижня смуга визначається як [2]:

$$L_t = M_t - k*\sigma_t,$$

де σ_t – стандартне відхилення ціни за той самий період, k – коефіцієнт відступу. Аналогічним чином визначається і верхня смуга [2]:

$$U_t = M_t + k*\sigma_t,$$

де σ_t – стандартне відхилення ціни за той самий період, k – коефіцієнт відступу.

Рис. 1. Структурна схема запропонованої методики системного аналізу

Смуги Болліндже́ра формують діапазон, в межах якого коливається ціна; ширина цього діапазону динамічно відображає волатильність ринку. Якщо ринкова волатильність зростає, стандартне відхилення збільшується і відстань між смугами розширюється; при зниженні волатильності смуги зближуються [4]. З точки зору інтерпретації, коли ціна підходить до верхньої смуги або прориває її, можливий сигнал про перекупленість або скору корекцію вниз. В інших випадках, коли відбувається вихід ціни до нижньої смуги, це може сигналізувати про перепроданість та потенційний розворот вгору. Важливо зазначити, що смуги Болліндже́ра самі по собі не прогнозують напрям руху, а вказують на рівні аномальної волатильності [2; 3; 4].

При реалізації обчислювального алгоритму, у вигляді комп'ютерної програми [1], було обчислено наступні показники:

bb_high, bb_low, bb_mid – верхня, нижня та середня лінії Болліндже́ра. Ширина смуг обчислюється, як два стандартних відхилення, тобто параметр $k = 2$, на основі ковзного вікна у 20 свічок ($n = 20$). Зауважимо, що спочатку була видалена трендова складова, для того щоб ряд став більш стаціонарним.

bb_high_mean_1h, bb_low_mean_1h, bb_mid_mean_1h – це середні значення обчислені для bb_high, bb_low, bb_mid за останній час. Усереднення дозволяє згладжувати короткострокові коливання, одночасно зберігаючи динаміку зміни ціни в моделі, незважаючи на стаціонарність.

Стохастичний осцилятор. Стохастичний осцилятор – класичний індикатор імпульсу, який був розроблений Джорджем Лейном у 1950-х роках [1; 3; 8]. Він показує положення поточної ціни відносно її мінімуму та максимуму за вибраний період і таким чином оцінює, чи не наближається ціна до крайніх значень свого діапазону. Основна ідея полягає в тому, що при висхідному тренді ціни закриття мають тенденцію наближатися до верхньої межі недавнього діапазону, а при низхідному – до нижньої . Стохастичний осцилятор складається з двох ліній: %K – швидкої стохастичної лінії, та %D – повільної (сигнальної) лінії, яка є згладженим середнім %K [1; 8]. Формалізовано лінія %K визначається наступним чином:

$$\%\mathrm{K} = \frac{\mathrm{P}_{\mathrm{close}} - \min\left(\mathrm{P}_{\mathrm{n}}\right)}{\max\left(\mathrm{P}_{\mathrm{n}}\right) - \min\left(\mathrm{P}_{\mathrm{n}}\right)},$$

де $\mathrm{P}_{\mathrm{close}}$ – поточна ціна закриття свічки, $\min\left(\mathrm{P}_{\mathrm{n}}\right)$ – мінімум ціни за останні n періодів, $\max\left(\mathrm{P}_{\mathrm{n}}\right)$ – максимум ціни за останні n періодів. Лінія $\%D$ визначається як:

$$\%\mathrm{D} = \frac{\%\mathrm{K}_1 + \%\mathrm{K}_2 + \%\mathrm{K}_3 + ... + \%\mathrm{n}}{\mathrm{n}},$$

де $\%K$ – швидка стохастична лінія, n – вікно періодів. Стохастичний осцилятор набуває значень від 0 до 100. Високе значення означає, що ціна знаходиться близько до верхнього екстремуму періоду (сильний імпульс вгору), тоді як низьке – близько до нижнього екстремуму (сильний імпульс вниз). Традиційно рівень $\%K > 80\%$ розглядається як зона перекупленості ринку, а $\%K < 20\%$ – як зона перепроданості. Перетин швидкої лінії $\%K$ вниз через рівень 80 може інтерпретуватися як сигнал до продажу (ослаблення висхідного імпульсу), а перетин вгору через рівень 20 – сигнал до купівлі (закінчення спадного імпульсу). Також аналізують перетини ліній: коли $\%K$ перетинає $\%D$ зверху вниз, це може вказувати на початок зниження (сигнал "продавати"), а перетин знизу вгору – на можливий розворот вгору («купувати») [8].

Стохастичний осцилятор описують як індикатор, що функціонує на основі рівнів підтримки та опору, вимірюючи поточну ціну відносно діапазону за період. Його основна мета – прогнозувати розвороти тренду шляхом виявлення моментів, коли ціна досягла крайньої верхньої або нижньої межі свого недавнього діапазону.

При реалізації обчислювального алгоритму у вигляді комп'ютерної програми [1] для врахування індексу стохастичного осцилятора, було обчислено наступні показники:

Stoch_k – значення ліні %K стохастичного осциллятора, що відображає положення ціни закриття відносно діапазону максимумів та мінімумів за останні 60 хвилин.

Stoch_d – значення сигнальної лінії %D (ковзне середнє %K з вікном 13), використовується для згладжування сигналу.

Stoch_k_lag1, stoch_k_lag2 – значення лінії %K, зрушені на 1 та 2 періоди назад відповідно. Ці лінії дозволяють враховувати інерцію осцилятора

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Stoch_d_lag1, stoch_d_lag2 – значення лінії %D, зрушені на 1 та 2 періоди назад відповідно. Відображають недавні рівні сигнального індикатора.

Stoch_k_diff1, stoch_k_diff2 – різниці між поточним та попереднім значеннями лінії %K (перший та другий лаговий приріст), використовуються як індикатори імпульсу.

Stoch_d_diff1, stoch_d_diff2 – приріст лінії %D, що показує швидкість зміни сигнальної кривої осцилятора.

Параболічний індикатор зупинки і розвороту. Параболічний індикатор зупинки і розвороту (Parabolic SAR – Parabolic Stop and Revers) – трендовий індикатор, розроблений Дж. У. Веллсом Вайлдером [3; 9; 15]. На відміну від осциляторів, Parabolic SAR прив'язаний безпосередньо до цінового графіка і відображається у вигляді серії точок (парабол), що розташовуються по той чи інший бік від цінових барів. Принцип дії цього індикатора полягає в тому, що при висхідному тренді точки розміщуються нижче ціни і поступово підтягуються вгору, а при низхідному тренді – вище ціни і спускаються вниз слідом за нею. У моменти, коли тенденція змінюється на протилежну, точки перевертаються на інший бік графіка. Перехід точки через ціну інтерпретується як сигнал до виходу з позиції і одночасного відкриття протилежної позиції [9].

Формалізація індикатора базується на наступній рекурентній формулі:

$$SAR_{t+1} = SAR_t + \alpha\left(EP_t - SAR_t\right),$$

де EP_t – найвища ціна при висхідному або найнижча при висхідному тренді, α – коефіцієнт прискорення SAR. Дж. У. Вайлдер, автор цього індикатору, рекомендував ініціалізувати $\alpha = 0.02$ і збільшувати цей фактор на 0.02 при кожному оновленні екстремуму EP тренду, встановивши максимальне його значення 0.20. Таким чином, чим довше триває тренд і чим більше нових екстремумів він встановлює, тим швидше параболічний індикатор зупинки і розвороту наздоганяє ціну по параболічній траєкторії. Якщо в наступному періоді, розраховане значення SAR, виявляється по інший бік від цінового бару (тобто перевищує попередній мінімум при висхідному русі чи опускається нижче попереднього максимуму при спадному русі), то це слугує ознакою зміни тренду – індикатор перестрибує на протилежний бік цінового графіка, і розрахунок продовжується вже для нового тренду. [3; 9; 15]

При реалізації обчислювального алгоритму у вигляді комп'ютерної програми [1] для врахування індексу Parabolic SAR, було обчислено наступні показники:

Psar – поточні значення лінії індикатора PSAR, визначає потенційні рівні зупинки чи розвороту тренду.

Psar_dir – бінарна змінна направлення тренду: 1, якщо PSAR під ціною (висхідний тренд), 0 – якщо над ціною (низхідний тренд).

Psar_lag1, psar_lag2, psar_lag3, psar_lag4 – значення PSAR, зрушені на 1–4 кроки назад, дозволяють враховувати історію тренду.

Psar_dir_lag1, psar_dir_lag2, psar_dir_lag3, psar_dir_lag4 – історія бінарного направлення тренду на 1–4 періоди назад.

Psar_delta1, psar_delta2 – зміна значення PSAR за останні періоди (прирости), відображають швидкість зміни рівня зупинки.

Psar_dir_delta1, psar_dir_delta2 – зміна напряму (psar_dir) між поточним та попередніми кроками, фіксують потенційні точки розвороту.

Метод математичного моделювання – випадковий ліс рішень (Random Forest). Отримані показники-регресори для кожного виду індикаторів було використано для моделювання цільової змінної сигнал методом випадкового лісу рішень (Random Forest) [3; 12]. Цей метод належить до ансамблевих методів машинного навчання, що базуються на сукупності великої кількості звичайних дерев рішень. Формально ансамбль моделей можна визначити як:

$$F(x) = \frac{1}{N}\sum_{i=1}^{N} f_i(x),$$

де $f_i(x)$ – результат окремого дерева рішень, а N – кількість дерев у ансамблі [3; 12].

Ідея методу полягає у формуванні множини дерев рішень, кожне з яких будується на основі повної підмножини даних та змінних, після чого результат прогнозу визначається на основі агрегування результатів цих дерев [3; 12]. Дерева рішень є одним з базових алгоритмів машинного навчання, які застосовуються для задач класифікації та регресії. Структуру дерева рішень можна формалізувати як: $G = (V, E)$, де V – множина вершин вузлів, а E – множина ребер (гілок). Кожен внутрішній вузол відповідає перевірці певної ознаки , гілки представляють можливі результати такої перевірки, а листові вузли – кінцеві рішення або прогнози $f(x)$. Основною перевагою дерев рішень є їхня інтерпретованість, прозорість у прийнятті рішень та простота реалізації.

Розглянемо принципи, на яких базується алгоритм випадкового лісу дерев рішень:

1) Метод Bagging – випадкове повторне відбирання підмножин навчальної вибірки із поверненням. Кожне дерево рішень навчається на власній підвибірці даних, що забезпечує різноманітність дерев та знижує ймовірність перенавчання [12].

2) Випадковий вибір ознак – на кожному етапі побудови дерева випадковим чином обирається обмежена кількість ознак із загального набору змінних, серед яких визначається найкращий критерій поділу. Це дозволяє додатково посилити різноманітність дерев рішень у складі ансамблю [12].

3) Для визначення найкращого критерію поділу часто використовується індекс Джині (Gini impurity), який формалізується наступним чином:

$$\mathrm{Gini}(\mathrm{D}) = 1 - \sum_{c=1}^{C} p_c^2,$$

де p_c – ймовірність належності об'єкта вибірки до класу c, а C – кількість класів [3; 12].

У задачах класифікації остаточний результат методу випадкового лісу дерев рішень визначається через голосування більшості:

$$\mathrm{F}(\mathrm{x}) = \mathrm{argmax}_c \sum_{i=1}^{N} \mathrm{I}(\mathrm{f}_i(\mathrm{x}) = \mathrm{c}),$$

де I – індикаторна функція, яка визначається наступним чином:

$$\mathrm{I}(\mathrm{z}) = \begin{cases} 1, \textit{якщо умова z виконується} \\ 0, \textit{якщо умова z не виконується} \end{cases}$$

Завдяки таким особливостям, метод випадкового лісу дерев рішень є одним із найбільш ефективних, стійких до шуму та популярних алгоритмів машинного навчання, який широко застосовується у вирішенні задач аналізу та прогнозування [3; 12].

Аналіз результатів моделювання на реальних даних. Запропонована методика системного аналізу для торгівлі фінансовими активами, була реалізована у вигляді комп'ютерної програми [1]. Розроблена програма [1] реалізовує індикатор волатильності – смуги Боллінджера, стохастичний осцилятор та параболічний індикатор зупинки і розвороту.

Для проведення обчислювальних експериментів, щодо навчання та варіації математичних моделей, було зібрано дані про криптовалюту Біткоїн, у торгівельній парі з Tether USDT, що є аналогом долару США для світового крипторинку. Ці дані для аналізу було вивантажено з відкритого ресурсу Yahoo Finance (finance.yahoo.com) [11], що надає безкоштовний доступ до ринкових даних різних видів активів, в тому числі криптоактиви. Дані було отримано у щохвилинному інтервалі у виді набору даних, що включає стовпчики-показники – початкова ціна відкриття (Open), максимальна ціна (High), мінімальна ціна (Low), ціна закриття (Close), обсяг торгів (Volume). Для отримання даних було використано python-бібліотеку YFINANCE (https://pypi.org/project/yfinance), що дозволяє за допомогою мови програмування Python вивантажити ці дані для аналізу. Для проведення дослідження було вивантажено дані за період з 01.01.2022 по 08.05.2025 [11]. За посиланням [1], разом із кодом програми на мові програмування Python, можна вивантажити і самі дані для аналізу, на яких проводилися відповідні обчислювальні експерименти.

Отримані дані було розділено на тренувальну та тестову вибірки у співвідношенні 4 до 1. Таким чином, у тренувальному наборі знаходяться дані у періоді з 01.01.2022 до 06.09.2024, а в тестовому з 06.09.2024 до 08.05.2025.

Формалізація цільової змінної аналізу. Цільова змінна позначається як Signal, в реалізованій компресній програмі [1], та приймає наступні три значення-рекомендації для ОПР:

0 – нічого не робити (none);

1 – досягається максимум ціни, в цьому випадку виставляється угода на продаж фінансового активу (sell);

2 – досягається мінімум ціни, в цьому випадку виставляється угода на купівлю фінансового активу (buy).

Зауважимо, що через незбалансованість класів цільової змінної, для сигналів на покупку та продаж, попередні результати моделювання за метрикою середньої точності (Average Precision score) виявилися дуже низькими. Тому для покращення результатів моделювання, було накладено додаткові умови на цільову змінну – розглядаються тільки прогнози-рішення, що мають ймовірність приналежності до класу buy або sell більше аніж 80%. В інших випадках прогноз-рішення класифікується як – нічого не робити (none).

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

В (табл. 1) наведено результати оптимізації гіперпараметрів, які було оптимізовано в рамках виконаного дослідження із використанням комп'ютерної програми [1]. Для вирішення задачі оптимізації було використано python-бібліотеку optuna, з якою більш детальніше можна ознайомитися за посиланням [7].

Наведемо опис гіперпараметрів, які було використано в алгоритмі розробленої комп'ютерної програми [1]:

1) max_depth – обмеження максимальної глибини кожного дерева рішень;
2) n_estimators – максимальна кількість дерев у ансамблі моделей;
3) min_samples_split – мінімальна кількість зразків у вузлі, які необхідні для його розбиття;
4) min_samples_leaf – мінімальну кількість зразків, які будуть міститись у листовому вузлі;
5) max_features – кількість ознак, які потрібно обрати для побудови кожного розбиття вузла;
6) criterion – критерій якості розбиття, зазвичай використовуються індекс Джині (Gini) або ентропія.

Таблиця 1

Оптимальні гіперпараметри, підібрані алгоритмом програми

№ з/п	Назва гіперпараметра	Смуги Болліндже́ра	Стохастичний осцилятор	Параболічний індикатор зупинки та розвороту
1	max_depth	25	9	2
2	n_estimators	129	311	411
3	min_samples_split	15	13	5
4	min_samples_leaf	16	16	10
5	max_features	sqrt	log2	None
6	criterion	Gini	Gini	Gini

Таблиця 2

Статистичні характеристики побудованих моделей

	Смуги Болліндже́ра	Стохастичний осцилятор	Параболічний індикатор зупинки та розвороту
Збалансована точність (Balanced Accuracy)	0.74	0.67	0.59
F1-оцінка (F1-score)	0.24	0.14	0.25
Оцінка середньої точності (Average Precision score)	0.36	0.37	0.34
Максимальний прибуток за період	6,9%	34,64%	-0,12%
Максимальне просідання портфеля протягом доби	-11,02%	-3,65%	-3,63%
Середній дохід з однієї прибуткової угоди	0.22%	0,21%	0,12%
Середні втрати з однієї збиткової угоди	-0,23%	-0,21%	-0,11%
Частка виграшних угод	53,56%	53.94%	48,67%
Коефіцієнт Шарпа	0.32	0.67	0,11
Середня кількість угод протягом доби	2,9	7,3	1,1

З отриманих результатів моделювання, що наведені у (табл. 2) можна зробити наступні висновки.

За значеннями метрики збалансованої точності (Balanced Accuracy), модель на основі смуг Болліндже́ра є кращою, тому що в середньому правильно виявляє 74% прикладів кожного класу, в той час як моделі стохастичного осцилятору та параболічного індикатору 67% і 59% відповідно.

Значення F1-оцінки для усіх моделей є досить низьким, тобто моделі не в змозі одночасно підтримувати високу повноту (Recall) та точність (Precision) для кожного з класів. Зауважимо, що метрика якості моделі Recall оцінює здатність моделі виявляти всі наявні позитивні випадки в даних, але вона також може спотворювати результати, так як у торгівлі активами краще пропустити вигідний момент аніж отримати хибний сигнал.

Проведені тестування на ринкових даних, що симулюють угоди покупки й продажу показали, що максимальний прибуток у 34,64% за період дослідження забезпечує модель на основі стохастичного осцилятору, а той час як модель із використанням смуг Болліндже́ра дає прибуток у 6,9%, а індекс параболічного індикатору взагалі надає невеликий але ж таки збиток у -0,12%.

Максимальне просідання портфеля протягом доби у -11,02% спостерігалося для смуг Болліндже́ра, в той час як інші моделі давали у два с половиною рази менше.

Найбільша кількість угод у протягом дня, у кількості 7,3 здійснювалося для моделі стохастичного осцилятору, в той час як смуг Болінджера 2.9 угоди на день, а індекс параболічного індикатору лише одну угоду на день.

В таблиці 3 наведені значення помісячної кумулятивної дохідності для розроблених моделей із використанням різних індикаторів, а на (рис. 2) відповідні графіки, побудовані на основі значень з цієї таблиці.

Найгірші значення коефіцієнтів Шарпа спостерігалися для моделей на основі смуг Болінджера та параболічного індикатору, що говорить про дуже низьке співвідношення прибутковості до ризику.

Таблиця 3

Значення помісячної кумулятивної дохідності розроблених моделей із використанням різних індикаторів

Рік-Місяць	Смуги Болінджера	Стохастичний осцилятор	Параболічний індикатор зупинки та развороту
2024-09	2,3	3,5	0,3
2024-10	4,1	6,9	0,5
2024-11	5,03	9,7	1,4
2024-12	7,6	14,8	0,1
2025-01	12,4	18,3	-1,2
2025-02	12,2	20,1	-0,2
2025-03	17,6	24,5	-1,5
2025-04	11,9	33,2	-1,7
2025-05	9,8	34,1	-0,4

Висновки. В статті запропоновано оригінальну методика системного аналізу, яка реалізована у вигляді комп'ютерної програми, складається з дев'яти кроків, та призначена для торгівлі фінансовими активами. При побудові моделей за цією методикою можуть використовуватися різноманітні індикатори та додаткові змінні. В рамках проведеного дослідження, на реальних статистичних даних про курс криптовалюти Біткоїн було з'ясовано, що найкращі результати за кумулятивною прибутковістю за весь період, кількістю угод на день, значенням максимальної просадки моделі та значенням коефіцієнту Шарпа, показують моделі випадкового лісу рішень, в яких застосовується в якості регресора індикатор стохастичного осцилятора. В той час як смуги Болінджера дають другий за значимістю результат, а параболічний індикатор зупинки та розвороту показують найгірші результати.

Перспективи щодо подальших досліджень. Окрім методу випадкового лісу рішень також планується в подальшому виконати дослідження інших популярних методів машинного навчання, а саме XGBoost та нейронні мережі.

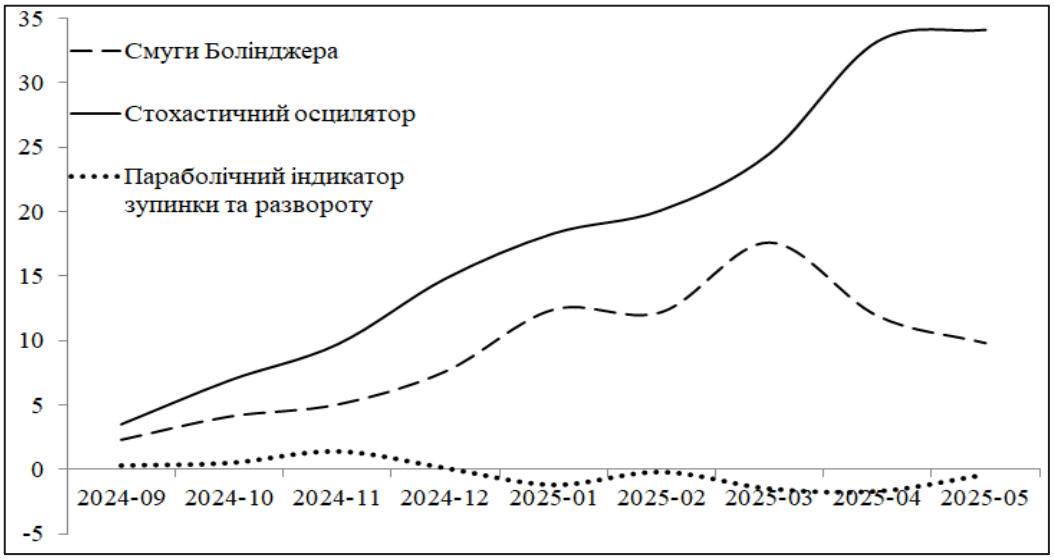


Рис. 2. Графіки кумулятивних дохідностей розроблених моделей із використанням різних індикаторів

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Список використаних джерел:

1. Комп'ютерна програма "ML technical indicators for Crypto". URL: https://github.com/oterentiev/ml-techical-indicators-for-crypto (date of access: 12.08.2025).

2. Bollinger J. A. Bollinger on Bollinger Bands. McGraw Hill, 2001. 227 p. ISBN-13: 978-0071373685.

3. Deep A., Monico C., Shirvani A., Rachev S., Fabozzi F. Assessing the Impact of Technical Indicators on Machine Learning Models for Stock Price Prediction. 2024. 22 p. URL: https://arxiv.org/html/2412.15448v1 (date of access: 12.08.2025).

4. Lakhwan D., Dave A. Determining the most efficient technical indicator of investing in financial markets based on trends, volume, momentum and volatility. *Myśl Ekonomiczna i Polityczna*. No. 3 Vol. 70. 2020 P. 64–137. URL: https://bazekon.uek.krakow.pl/rekord/171628292 (date of access: 14.08.2025).

5. Mostafavi S. M., Hooman A. R. Key technical indicators for stock market prediction. Machine Learning with Applications. Volume 20, June 2025. 16 p. Online ISSN: 2666-8270.

6. Murphy J. J. Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications. New York Institute of Finance, 1999. 576 p. ISBN-13: 978-0735200661

7. Optuna: A hyperparameter optimization framework. URL: https://optuna.readthedocs.io/en/stable/ (date of access: 15.08.2025).

8. Paik C., Choi J., Vaquero I. U. Algorithm-based low-frequency trading using a stochastic oscillator, Williams %R, and trading volume for the S&P 500. *Journal of Risk and Financial Management*. Vol. 17 No.11 Art. 501. 2024. 20 p. DOI: http://dx.doi.org/10.3390/jrfm17110501

9. Prasetijo A. B., Saputro T. A., Windasari I. P., Windarto Y. E. Buy/sell signal detection in stock trading with Bollinger Bands and Parabolic SAR: With web application for proofing trading strategy. Proceedings of the 2017 4th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE), Piscataway NJ, 2017. P. 41–44. DOI: http://dx.doi.org/10.1109/ICITACEE.2017.8257672

10. Precious Metal Market Summary. URL: https://www.grandviewresearch.com/industry-analysis/precious-metals-market (date of access: 16.07.2025).

11. Roussi R. YFinance – Yahoo! Finance Market Data Downloader. URL: https://ranaroussi.github.io/yfinance/ (date of access: 16.07.2025).

12. Scornet E., Biau G., Vert J.-P. Consistency of Random Forests. *Annals of Statistics*. Vol. 43, No. 4. 2015. 1716–1741 p. DOI: http://dx.doi.org/10.1214/15-AOS1321

13. Sharpe W. F. The Sharpe Ratio. *Journal of Portfolio Management*. Vol. 21, No. 1. 1994. P. 49–58. DOI: http://dx.doi.org/10.3905/jpm.1994.409501

14. UK – London Stock Exchange Market Capitalization. URL: https://en.macromicro.me/series/4227/uk-london-market-cap?utm_source=chatgpt.com (date of access: 16.07.2025).

15. Wilder W. New Concepts in Technical Trading Systems Hardcover. Trend Research, 1978. 141 p. ISBN-13 : 978-0894590276

16. Zhang J., Cai K., Wen J. A survey of deep learning applications in cryptocurrency. *iScience*. vol. 27, 2024. 40 p. URL: https://doi.org/10.1016/j.isci.2023.108509 (date of access: 12.08.2025).

Дата надходження статті: 09.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.5
DOI https://doi.org/10.32689/maup.it.2025.3.24

Олександр ХОМЕНКО
аспірант кафедри інженерії програмного забезпечення в енергетиці,
Навчально-науковий інститут атомної та теплової енергетики,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»,
khomenkosasha99@gmail.com
***ORCID:** 0000-0003-1964-1097*

Олександр КОВАЛЬ
доктор технічних наук, професор, завідувач кафедри інженерії програмного забезпечення в енергетиці,
Навчально-науковий інститут атомної та теплової енергетики,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»,
avkoval@gmail.com
***ORCID:** 0000-0003-0991-6405*

АНАЛІЗ ТА ПОРІВНЯННЯ СЦЕНАРІЇВ КАСКАДНИХ ЕФЕКТІВ В КРИТИЧНІЙ ІНФРАСТРУКТУРІ

Анотація. *Дослідження сценаріїв каскадних ефектів в критичній інфраструктурі відіграє важливу роль для прийняття рішень, щоб зменшити негативні наслідки. Дані про роботу критичної інфраструктури є закритими або обмеженими, що ускладнює процес аналізу каскадних ефектів. Для генерування та аналізу сценаріїв каскадних ефектів використовуються різні підходи: графові моделі, моделі потоку потужності, гібридні підходи, які використовуються відповідно до поставлених задач. Розвиток машинного навчання супроводжується появою нових перспективних підходів, що використовуються для дослідження властивостей роботи електромереж в різних сценаріях.*

Метою статті *є дослідження каскадних ефектів в критичній інфраструктурі та створення методу для аналізу та порівняння сценаріїв каскадних ефектів в електромережі, використовуючи графову нейронну мережу та коефіцієнт подібності.*

Методологія. *У статті описано процес створення даних в сценаріях роботи електромережі при виведенні компонентів системи, що потенційно можуть призвести до каскадного ефекту. Розроблено модель автоенкодера на основі графової нейронної мережі, що використовується для формування представлення про крок сценарію (стан електромережі). Косинус подібності використано для порівняння кроків в різних сценаріях та пошуку подібних станів мережі. На основі подібності сценаріїв про стани електромережі можливо зробити висновки про можливий розвиток каскадного ефекту в сценарії.*

Наукова новизна *роботи полягає у розробці методу, що покращує процес аналізу сценаріїв каскадних ефектів, порівняння послідовностей подій в сценаріях, визначення подібних ситуацій для прийняття рішень на основі існуючого досвіду. Визначено можливості для розширення методу, використовуючи поєднання графової нейронної мережі та LSTM для формування комплексного представлення послідовності кроків в сценаріях.*

Висновки. *Проведено дослідження підходів для аналізу каскадних ефектів в електромережах. На основі проведеного дослідження було визначено перспективні напрямки, які потенційно можуть покращити процес порівняння сценаріїв каскадних ефектів. Для аналізу та порівняння сценаріїв каскадних ефектів в критичній інфраструктурі (електромережі) було розроблено метод, що використовує модель автоенкодера, що містить розроблений шар графової нейронної мережі, який покращує точність роботи моделі при вивченні зв'язків, впливу параметрів компонентів в електромережі та формує представлення стану електромережі в кроці сценарію. Використано косинус подібності для пошуку схожих сценаріїв, що потенційно можуть доповнити інформацію про стан електромережі в наступних кроках сценарію. Розроблений метод може працювати з різним рівнем деталізації сценаріїв, що забезпечує його адаптивність до вхідних даних.*

Ключові слова: *критична інфраструктура, машинне навчання, нейронна мережа, коефіцієнт подібності, графи, каскадний ефект, моделювання подій.*

Oleksandr KHOMENKO, Oleksandr KOVAL. ANALYSIS AND COMPARISON OF CASCADE EFFECT SCENARIOS IN CRITICAL INFRASTRUCTURE

Abstract. *The study of cascading effects scenarios in critical infrastructure plays an important role in decision-making to reduce negative consequences. Data on the operation of critical infrastructure is closed or limited, which complicates the process of analyzing cascading effects. Various approaches are used to generate and analyze cascading effects scenarios: graph models, power flow models, hybrid approaches that are used in accordance with the tasks. The development of machine learning is accompanied by the emergence of new promising approaches used to study the properties of power grids in various scenarios.*

The aim of the article *is to investigate cascading effects in critical infrastructure and to create a method for analyzing and comparing cascading effects scenarios in the power grid using a graph neural network and similarity coefficient.*

Methodology. *The article describes the process of generating data in power grid scenarios when deriving system components that can potentially lead to a cascade effect. An autoencoder model based on a graph neural network is developed, which is used to form a representation of the scenario step (power grid state). The cosine of similarity is used to compare steps in different scenarios and search for similar network states. Based on the similarity of scenarios about power grid states, it is possible to draw conclusions about the possible development of a cascade effect in the scenario.*

The scientific novelty of the work *lies in the development of a method that improves the process of analyzing cascading effects scenarios, comparing sequences of events in scenarios, and identifying similar situations for decision-making based on existing experience. Possibilities for expanding the method are identified, using a combination of a graph neural network and LSTM to form a complex representation of the sequence of steps in scenarios.*

Conclusions. *A study of approaches to the analysis of cascading effects in power grids was conducted. Based on the conducted study, promising directions were identified that could potentially improve the process of comparing cascading effect scenarios. To analyze and compare cascading effect scenarios in critical infrastructure (power grid), a method was developed that uses an autoencoder model containing a developed graph neural network layer, which improves the accuracy of the model when studying connections, the influence of component parameters in the power grid, and forms a representation of the power grid state in the scenario step. The cosine of similarity was used to search for similar scenarios that could potentially supplement information about the power grid state in subsequent steps of the scenario. The developed method can work with different levels of scenario detail, which ensures its adaptability to input data.*

Key words: *critical infrastructure, machine learning, neural network, similarity coefficient, graphs, cascading effect, event modeling.*

Постановка проблеми. Критична інфраструктура – комплексна система, яка складається з множини об'єків, зв'язків між ними та відіграє важливу роль для забезпечення надання послуг в сучасному суспільстві. Протягом останніх років складність критичної інфраструктури збільшується, а збій в роботі пов'язаних системах може призвести до небезпечного явища – каскадного ефекту. Важливим об'єктом критичної інфраструктури є електромережі, які забезпечують електроенергією побутових споживачів та промислові об'єкти. Електромережі є комплексними об'єктами, де використовуються інструменти моніторингу, захисту та стабілізації мережі при виникненні потенційних збоїв. Але попри існуючі підходи системи є вразливими до каскадних ефектів, що виникають внаслідок збоїв та призводять до повного чи часткового знеструмлення мережі (блекаут). На виникнення та розвиток каскадних ефектів впливають різні чинники, наприклад: природні умови, несправність в обладнанні, помилки операторів. Знеструмлення були в різних країнах світу та мали різні передумови та наслідки, наприклад:

– 30 і 31 липня 2012 року в Індії знеструмлення вплинуло на 400 мільйонів людей та 620 мільйонів людей відповідно [28].

– 14–16 серпня 2003 року в США та Канаді знеструмлення вплинуло приблизно на 55 мільйонів людей [20].

– 28 вересня 2003 року в Італії знеструмлення вплинуло на приблизно 56 мільйонів людей [27].

– 28 вересня 2016 року в Південній Австралії знеструмлення сталося у результаті шторму та вплинуло на 850 тисяч споживачів [29].

Пошкодження об'єктів критичної інфраструктури в Україні вплинуло на забезпечення користувачів електроенергією:

– 20 травня 2022 року без електропостачання залишилося приблизно 635,8 тисячі споживачів [2].

– 1 червня 2022 року без електропостачання залишилося приблизно 632 тисячі споживачів [3].

– 10 червня 2022 року без електропостачання залишилося приблизно 640,75 тисяч споживачів [4].

Відключення електроенергії зазвичай негативно впливають на залежні інфраструктури. Наприклад, транспортна інфраструктура: виникають проблеми з логістикою, утворюються затори із транспортних засобів на дорогах у населених пунктах, що ускладнює переміщення населення. Затори на дорогах ускладнюють екстреним службам виконувати свою роботу, що може бути критичним у певних обставинах. Крім того перебої в логістиці негативно впливають на залежні сектори економіки. Зменшення потужності роботи електромережі може вплинути на роботу водопостачання. Зниження тиску води або її відсутність в системі, потенційно може погіршити якість води. Цей процес є прикладом каскадного ефекту, що був породжений відключенням компонентів в електромережі, а наслідки з'явилися в інших секторах.

Аналіз останніх досліджень та публікацій. Об'єкти критичної інфраструктури та зв'язки між ними описуються за допомогою графу $G=(V,E)$, де $|V|=n$ – кількість вершин, $|E|=m$ – кількість ребер (дуг). Взаємозалежності в критичні інфраструктурі класифікуються як [22]:

– Фізичний. Наприклад, збій енергосистем впливає на роботу світлофорів для регулювання перехресть.

– Кібернетичний. Наприклад, відсутність електроенергії призводить до перебоїв зв'язку, погіршується комунікація між об'єктами системи.

– Географічний. Наприклад, при виникненні сильних поривів вітру можуть бути пошкоджені лінії електропередачі.

– Логічний. Наприклад, при обмеженні руху по деяким маршрутах зростає завантаженість доріг на альтернативних маршрутах.

Розвиток каскадних ефектів поділяється на фази [18]:

1. Попередня фаза. Відбувається повільний прогрес збоїв.
2. Фаза ескалації. Збої стрімко поширюються, запобігання відключень стає складнішим.
3. Фаза каскадного поетапного припинення роботи. Швидкість розповсюдження збоїв сповільнюється, значна кількість компонентів системи вже вийшли з ладу.

Оскільки негативний вплив каскадних ефектів може бути зменшений на початковому етапі, тому доцільно досліджувати існуючі та потенційні сценарії роботи інфраструктури при виникненні та розвитку каскадних ефектів [5–7].

Для дослідження властивостей графів (міцності мережі) використовуються метрики, які згруповано в шість категорій [21]:

– Кластеризація (Clustering).
– Зв'язність (Connectivity).
– Відстань (Distance).
– Пропускна здатність (Throughput).
– Спектральні методи (Spectral Methods).
– Географічні метрики (Geographical Metrics).

Графові спектральні методи (Graph Spectral Techniques) застосовуються для управління водопровідною мережею та можуть бути використані для [10]:

– Виявлення вузьких місць (bottlenecks) за допомогою значень спектрального розриву (spectral gap). Чим більше значення спектрального розриву, тим стійкішою є мережа.

– Вимірювання міцності (strength) мережі для розбиття на підмережі. Алгебраїчна зв'язність (Algebraic connectivity) визначає міцність з'єднань мережі і стійкість до збоїв: чим більше значення алгебраїчної зв'язності, тим стійкішою є мережа. Спектральний радіус або індекс (Spectral radius or Index) може використовуватися для оцінки рівня зв'язності мережі [10].

Теорія графів використовується для аналізу вразливостей транспортних мереж [17]:

1. Середня відстань між усіма парами вершин графу (Average distance): чим нижче значення метрики, тим сильніша зв'язність [12].
2. Ефективність (Efficiency): чим вище значення метрики, тим більша міцність мережі [12].

Для оцінки міцності мережі можливо використовувати Average Edge Betweenness та Average Vertex Betweenness: чим менше значення метрик, тим більша міцність мережі [12].

Топологічний та гібридний підхід на основі теорії складних мереж використовується для визначення стійкості та вразливості в електромережах [9]. Для оцінки характеристик об'єктів використовуються різні метрики, наприклад: average path length, node degree distribution, betweenness, size of the largest component та ефективність мережі [9]. В гібридному підході використовуються концепції з електротехніки для покращення топологічного підходу. При виникненні збоїв у функціонуванні компонентів електромережі електроенергія перенаправляється відповідно до законів Кірхгофа та Ома, які зазвичай ігноруються в топологічних моделях, що може призвести до помилкових висновків [15; 16], оскільки каскадні збої в електричних мережах поширюються не локально, що ускладнює процес аналізу каскадних ефектів.

Аналіз потоку потужності (Power Flow Analysis) використовується для аналізу роботи енергосистеми, відіграє важливу роль при проектуванні мережі, дослідженні роботи компонентів при різних вхідних даних. Електромережа моделюється за допомогою графу, який складається з: вузлів (nodes), що представляють навантаження (споживачами можуть бути пов'язані критичні інфраструктури), генератори, шунти; гілок (branches), що представляють трансформатори та лінії електропередачі. Для кожної шини обчислюються величини (задані дві з цих чотирьох величин, а інші дві є невідомими):

1. величини напруги (voltage magnitude, $|V|$);
2. кута напруги (voltage phase angle, δ);
3. реальної потужності (injected real power, P);
4. реактивної потужності (injected reactive power, Q).

На основі відомих величин кожна із шин може бути класифікована як [31]: Slack bus, PQ bus, PV bus.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Таблиця 1

Категорії шин в електромережі при моделюванні

Тип шини	P	Q	|V|	δ
Slack	Не задано	Не задано	Задано	Задано
PQ	Задано	Задано	Не задано	Не задано
PV	Задано	Не задано	Задано	Не задано

Рівняння для активної та реактивної потужності визначаються як:

$$P_i = \sum_{k=1}^{N} |V_i||V_k|\left(G_{ik}cos\delta_{ik} + B_{ik}sin\delta_{ik}\right) \quad (1)$$

$$Q_i = \sum_{k=1}^{N} |V_i||V_k|\left(G_{ik}sin\delta_{ik} - B_{ik}cos\delta_{ik}\right) \quad (2)$$

де $\delta_{ik} = \delta_i - \delta_k$ – різниця фазових кутів між вузлом *i* та *k*.

Рівняння потоку потужності можуть бути розв'язані за допомогою чисельних методів. При збільшенні розмірності, складності мережі обчислювальна складність моделі зростає, що ускладнює процес аналізу сценаріїв.

В останні роки машинне навчання (machine learning) та глибоке навчання (deep learning) стрімко розвиваються, розроблюються нові підходи для аналізу каскадних збоїв в енергосистемах [18]. Одним із підходів для прогнозування каскадних збоїв є використання графових нейронних мережі (Graph Neural Networks) [8], які також можуть використовуватися для онлайн-прогнозування каскадних збоїв в електромережах [24]. Дослідження показують перспективи використання графових нейронних мереж для аналізу каскадних ефектів в електромережах та можливості для покращення результатів відповідно до поставлених задач. Для тренування та валідації роботи нейронних мереж потрібні набори даних, що описують сценарії каскадних ефектів в електромережах. Оскільки дані про роботу критичної інфраструктури (електромереж) є обмеженими або недоступними, то виникає необхідність у створенні штучних сценаріїв розвитку каскадних ефектів в електромережах. Для симуляції сценаріїв роботи електромережі (каскадних збоїв та їх розвиток) використовуються фізичні моделі та ймовірнісні моделі [14]. Для аналізу стійкості електромереж в роботі може використовуватися модель каскадних відмов змінного струму, яка включає динамічні явища, механізми захисту [19]. Ця модель може бути використана для генерації сценаріїв роботи електромережі при виникненні несправності в компонентах мережі, що потенційно призводить до каскадного ефекту. При аналізі сценаріїв каскадних ефектів в електромережах виникає необхідність в методах, що забезпечують аналіз та порівняння сценаріїв з допустимою похибкою для формування кроків для можливого зменшення наслідків каскаду на основі існуючих знань.

Метою статті є розробка методу для аналізу та порівняння сценаріїв каскадних ефектів в критичній інфраструктурі на прикладі електромережі, використовуючи графову нейронну мережу та коефіцієнт подібності. Розроблений метод покращує процес аналізу сценаріїв каскадних ефектів, порівняння послідовностей подій в сценаріях, визначення подібних ситуацій для прийняття рішень на основі існуючого досвіду.

Виклад основного матеріалу дослідження. Оскільки дані про роботу електромережі є обмеженими, то виникає потреба у створенні синтетичних сценаріїв роботи електромережі. Для генерації сценаріїв розвитку каскадних ефектів використовується алгоритм, що використовує модель [19] та складається з кроків:

1. Визначити початкові параметри електромережі.
2. Визначити множину об'єктів, що будуть виведені з ладу.
3. Запустити модель потоку потужності.
4. Запустити захисні механізми для компонентів.
5. Визначити статуси та параметри компонентів електромережі.
6. Зберегти результати сценарію.

Нейронні мережі використовуються для виявлення складних закономірностей в даних та складаються з множини нейронів, що утворюють шари: вхідний шар, приховані шари, вихідний шар. При навчанні нейронної мережі використовується пряме поширення (forward propagation) та зворотне поширення (backpropagation): на кожній ітерації ваги та зміщення мережі оновлюються, щоб мінімізувати похибку. При проектуванні нейронної мережі вибір функції активації відіграє важливу роль, оскільки кожна з них має свої особливості та використовується в залежності від задачі та архітектури [11].

Графові нейронні мережі використовуються для вивчення зв'язків в даних на основі графів. Завдання на основі графових даних можуть бути класифіковані у наступні категорії [25]: Node Level Task (Завдання рівня вузла), Edge Level Task (Завдання рівня ребра), Graph Level Task (Завдання рівня графа).

Завдання рівня вузла діляться на чотири категорії:

– Класифікація вузлів (Node classification) – визначення класів вузлів на основі їхніх ознак та зв'язків.

– Регресія вузлів (Node regression) – прогнозування числових значень для вузлів.

– Кластеризація вузлів (Node clustering) – поділ вузлів на класи та групування вузлів з подібними характеристиками.

– Виявлення аномалій вузлів (Node anomaly detection) – ідентифікація вузлів графа, які відрізняються від визначених характеристик.

Завдання рівня ребра поділяються на категорії:

– Прогнозування зв'язків (Link prediction) – визначення ймовірності створення ребра між двома вузлами на основі структури графа та характеристик вузлів.

– Класифікація ребер (Edge classification) – прогнозування класу для ребер в графі (наприклад, категорія або статус ребра).

– Регресія ребер (Edge regression) – завдання прогнозування числових значень для ребер у графі.

– Кластеризація ребер (Edge clustering) – групування подібних ребер у графі в кластери на основі їхніх характеристик.

– Виявлення аномальних ребер (Edge outlier detection) – визначення ребер у графі, які відрізняються від характеристик у графі (наприклад, незвичайні ознаки або зв'язки).

Завдання рівня графа визначаються на рівні цілих графів (регресія, класифікація, висновки для графу або підграфу).

Основою графових нейронних мереж (GNN) є фреймворк передачі повідомлень (message-passing neural network) – процес ітеративної передачі повідомлень між вузлами та агрегації інформації від їхніх сусідів, що дозволяє моделі розуміти комплексні зв'язки та залежності в даних на основі графів [13].

Фреймворк передачі повідомлень складається з наступних кроків:

1. Агрегація повідомлень (Aggregation of messages).

Кожен вузол використовує функцію для створення повідомлення для кожного сусіднього вузла. Для кожного вузла v збирається інформація від його сусідніх вузлів $N(v)$. Агреговане повідомлення m_v для вузла v обчислюється як агрегування інформації від сусідніх вузлів.

$$m_v = aggregate\left(\left\{h_u, u \in N(v)\right\}\right) \tag{3}$$

2. Оновлення (Update).

Оновлення представлення вузла h_v на основі агрегованого повідомлення m_v та поточного представлення вузла h_v виконується за допомогою функції:

$$h_v^{'} = update\left(h_v, m_v\right) \tag{4}$$

Вибір функції aggregate, update та кількості ітерацій залежить від архітектури GNN. Існують різні архітектури графових нейронних мереж, які мають свої особливості [26]. При значному збільшенні глибини графової нейронної мережі (кількості шарів) виникає надмірне згладжування (oversmoothing) [23], яке негативно впливає на роботу моделі.

Метод визначення подібних ситуацій в електромережі для пошуку можливих сценаріїв розвитку каскадного процесу для прийняття рішень складається з наступних кроків:

1. Визначити вхідні дані мережі.

2. Використати модель автоенкодера для формування представлення електромережі в момент часу в сценарії.

3. Знайти подібні стани електромережі в сценаріях: порівняти дані за допомогою коефіцієнту подібності.

Вхідні дані. Кожна шина i має ознаки:

– Чиста активна потужність (P_i) визначається як різниця між реальною потужністю (P_{gi}), що генерується на цій шині, та реальною потужністю (P_{di}), що споживається навантаженнями, які підключені до неї.

– Чиста реактивна потужність (Q_i) визначається як різниця між реактивною потужністю (Q_{gi}), що генерується на цій шині, та реактивною потужністю (Q_{di}), що споживається навантаженнями, які підключені до неї.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

– Величина напруги (Vm_i).
– Кут напруги (Va_i).

Кожна гілка, що з'єднує шини i та j має ознаки:
– Pf: реальна потужність, яка походить від шини i на початку гілки.
– Pt: реальна потужність, яка надходить на шину j в кінці гілки.
– Qf: реактивна потужність, яка походить від шини i на початку гілки.
– Qt: реактивна потужність, яка надходить на шину j в кінці гілки.

Додатково для аналізу можуть додаватися ознака, що відображає стан компонента системи.

Модель автоенкодера для формування представлення кроку сценарію. Для формування представлення про стан роботи електромережі в момент часу сценарію використовується модель автоенкодера. Енкодер складається з двох входів, що оброблюють дані про вершини та ребра графа: $X \in R^D$ – інформація про ознаки вершин та $E \in R^D$ – інформація про ознаки ребер:

– Інформація про вершини $z_{ex} \in R^M, M < D$

$$z_{ex} = \sigma_{ex}\left(W_{ex}X + b_{ex}\right) \tag{5}$$

– Інформація про ребра $z_{ee} \in R^M, M < D$

$$z_{ee} = \sigma_{ee}\left(W_{ee}E + b_{ee}\right) \tag{6}$$

Після попередньої обробки та зменшення даних дані про вершини та ребра об'єднуються для обробки в спільному шарі та вивчення закономірностей:

$$z_{shared} = z_{ex} \,||\, z_{ee} \tag{7}$$

Дані поступово стискаються до визначеного розміру z_{emb} за допомогою послідовного зменшення розмірності шарів.

Декодер відновлює стиснене представлення z_{emb} до початкових даних через поступове збільшення розмірності даних:

– Інформація про вершини $\hat{X}$:

$$\hat{X} = \sigma_{dx}\left(W_{dx}z_{shared} + b_{dx}\right) \tag{8}$$

– Інформація про ребра $\hat{E}$:

$$\hat{E} = \sigma_{de}\left(W_{de}z_{shared} + b_{de}\right) \tag{9}$$

Результати оцінюються за допомогою функції втрат (loss fucntion), яка складається з двох функцій втрат:

Помилка при відновленні ознак вершин та ребер:

$$Mean\,Squared\,Error = \frac{1}{N}\sum_{i=1}^{N}\left(x_i - y_i\right)^2 \tag{10}$$

Помилка при відновленні статусів вершин та ребер:

$$Binary\,Cross\,Entropy\,Loss = -\left[y_i * \log\left(x_i\right) + \left(1 - y_i\right) * \log\left(1 - x_i\right)\right] \tag{11}$$

Модель автоенкодера, яка складається з лінійних шарів, може мати складнощі при вивченні зв'язків та взаємовпливу компонентів електромережі. Для навчання, валідації та тестування моделі були згенеровані сценарії роботи електромережі, що складається з 9 шин.

При встановлених значеннях параметрів: розмір представлення Embedding_Size = 80, Epoch = 500 помилка на даних для тренування (Train Loss) = 0.2530, помилка на даних для валідації (Val Loss) = 0.2699, помилка на даних для тестування (Test Loss) = 0.2702. Коефіцієнт детермінації для вершин $R^2\left(nodes\right)$ = 0.7830, коефіцієнт детермінації для ребер $R^2\left(edges\right)$ = 0.6755.

Для роботи з даними у вигляді графу використовуються графові мережі, які потенційно можуть краще оброблювати ознаки графів. В моделі використовується комбінація із лінійних шарів та шарів для обробки графів.

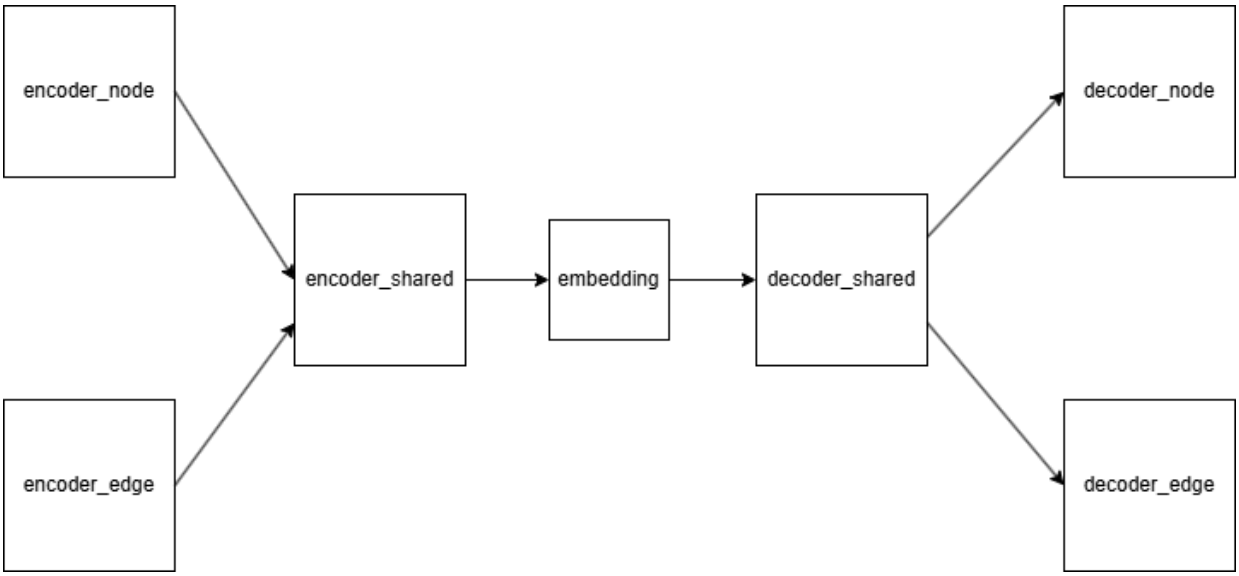


Рис. 1. Архітектура автоенкодера, що обробляє вершини та ребра електромережі

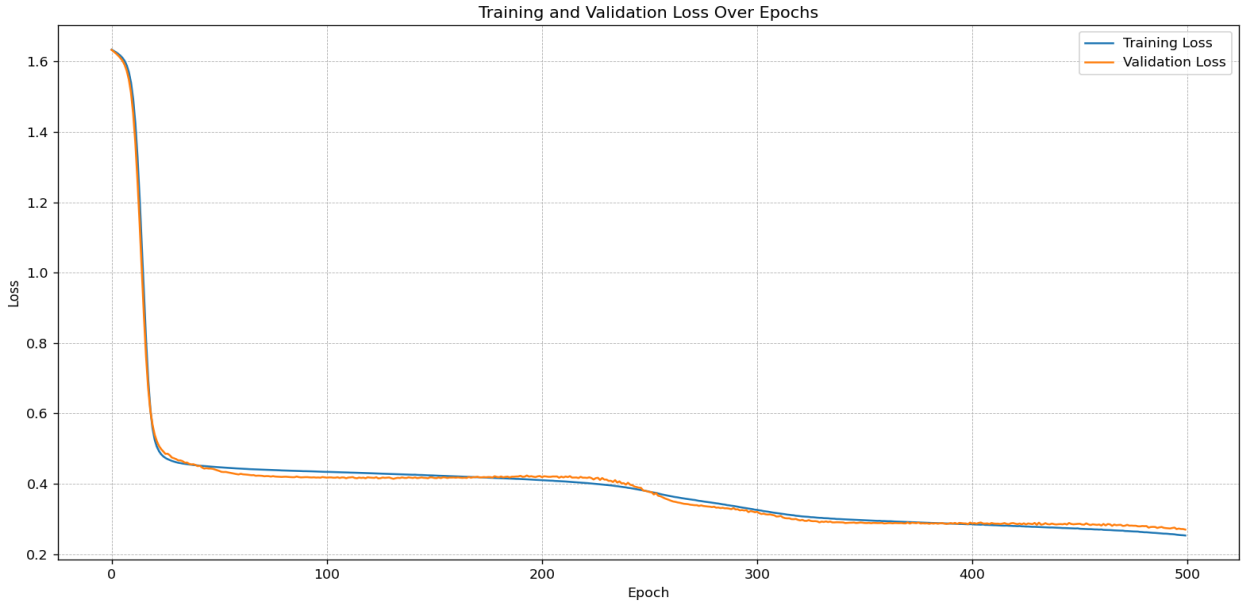


Рис. 2. Графік зміни помилки під час тренування моделі на даних для тренування та валідації (Embedding_Size=80)

На вхід автоенкодера подається граф, що характеризується:

1. Ознаками вершин, що визначається матрицею.
2. Ознаками ребер, що визначається матрицею.
3. Список ребер, які з'єднують вершини.

Дані проходять через шари, які адаптовані для роботи з графами. В результаті отримане представлення даних через декодер відтворює граф, що складається з ознак вершин, зв'язків між ребрами, ознак ребер. При дослідженні роботи параметрів об'єктів електромережі доцільно створити шар, який може поєднати інформацію про взаємодію вершин та ребер в графі. Розроблений графовий шар складається з наступних кроків:

Крок 1. Визначити множину ознак x_j, які будуть передані до цільових вершин .

Крок 2. Визначити множину ознак ребер $edge\_attr$, які з'єднують вершини j та i.

Крок 3. Застосувати лінійне перетворення до множини ознак вершин та ребер.

$$lin\_node = X_{nodes} * W_{nodes}^{T} + b_{nodes} \quad (12)$$

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

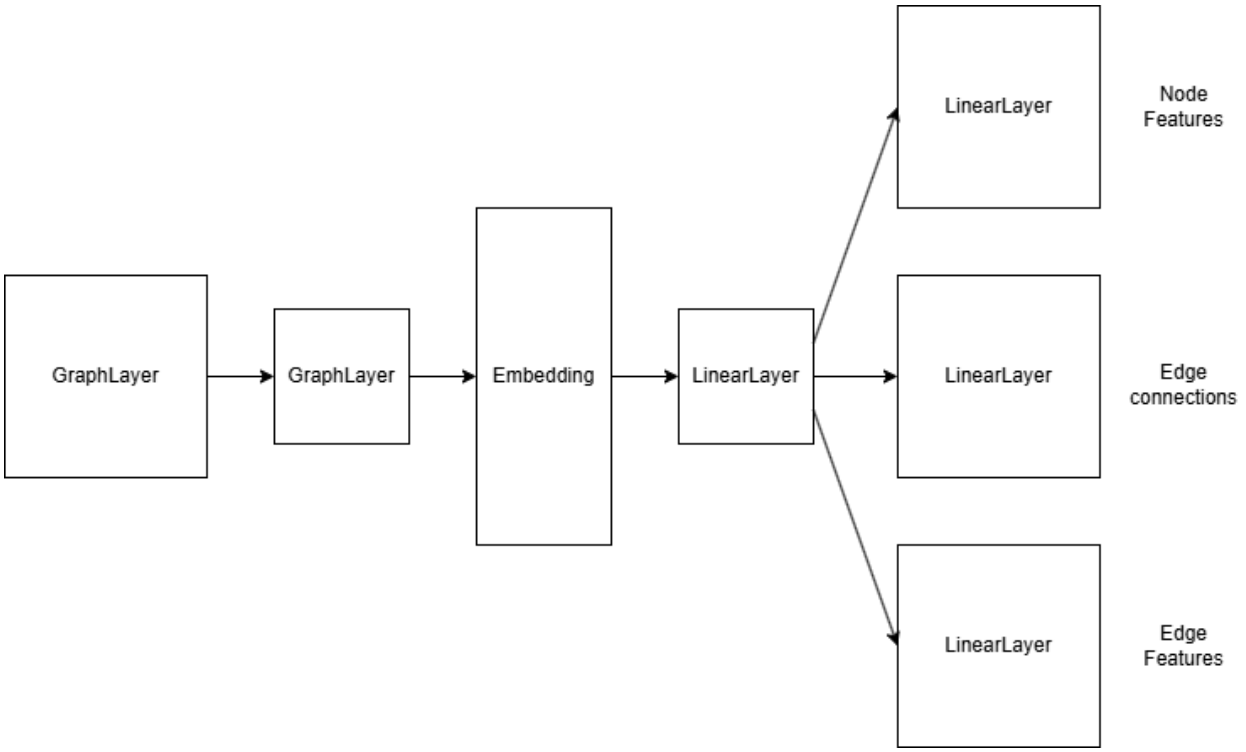


Рис. 3. Архітектура автоенкодера з шарами для обробки інформації про електромережу, яка визначена за допомогою графа

де lin_node – матриця лінійного перетворення ознак вершин графу, X_{nodes} – вхідна матриця ознак вершин, W_{nodes} – вагова матриця із параметрами для навчання, b_{nodes} – вектор зміщення.

$$lin\_edge = X_{edges} * W_{edges}^{T} + b_{edges} \tag{13}$$

де lin_edge – матриця лінійного перетворення ознак вершин графу, X_{edges} – вхідна матриця ознак вершин, W_{edges} – вагова матриця із параметрами для навчання, b_{edges} – вектор зміщення.

Крок 4. Сформувати повідомлення, які об'єднують перетворені ознаки вершин та ребер.

$$Messages = X_{nodes} * W_{nodes}^{T} + b_{nodes} + X_{edges} * W_{edges}^{T} + b_{edges} \tag{14}$$

Крок 5. Агрегація повідомлень для кожної вершини і від сусідніх вершин.

$$Aggregate_i\left(Messages\right) = \sum Messages_i \tag{15}$$

Крок 6. Об'єднати агреговані повідомлення з початковими ознаками вершини.

$$Concat\_data = X_{nodes} || Aggregate_i\left(Messages\right) \tag{16}$$

Крок 7. Застосувати лінійне перетворення для формування результуючого представлення.

$$Embedding\_Nodes = Concat\_data * W_{Embedding\_Nodes}^{T} + b_{Embedding\_Nodes} \tag{17}$$

При необхідності із вихідного представлення можливо виокремити представлення ребер для передачі та обробки в наступному шарі:

$$Edges\_Features = Embedding\_Nodes\left[source\_index\right] || Embedding\_Nodes\left[destination\_index\right] \tag{18}$$

$$Embedding\_Edges = Edges\_Features * W_{Edges\_Features}^{T} + b_{Edges\_Features} \tag{19}$$

При використанні комплексного представлення інформації про вершини та ребрах у вершинах та використанні в наступному шарі при значенні Epoch = 1000 помилка на даних для тренування (Train Loss) = 0.0003, помилка на даних для валідації (Val Loss) = 0.0001, помилка на даних для тестування (Test Loss) = 0.0001. Для вершин $R^2\left(nodes\right)$ = 0.9981, для ребер $R^2\left(edges\right)$ = 0.9719.

Для обчислення представлення про граф в момент сценарію застосовується механізм пулінгу (graph-level pooling) для агрегації ознак вершин, що були отримані в Embedding шарі. Наприклад, global_mean_pool обчислює усередненні значення ознак вершин графів в пакеті (batch).

Порівняння стану електромережі в моментах сценаріїв. Для визначення подібності представлень графів використовується косинус подібності, який обчислюється як косинус кута між ненульовими векторами. При значенні кута 0 градусів косинус подібності дорівнює 1 – однаковий напрямок, максимальна подібність; при значенні кута 90 градусів косинус подібності дорівнює 0 – вектори ортогональні, відсутність подібності; діаметрально направлені вектори мають подібність -1 [1].

$$cos(\theta) = \frac{A \cdot B}{\|A\| * \|B\|} \tag{20}$$

де $A \cdot B$ – добуток векторів, $\|A\|$, $\|B\|$ – довжина векторів A, B відповідно.

При визначенні подібності станів електромережі в сценаріях можливо ввести граничні значення для визначення рівня подібності.

При виявленні подібних станів електромережі в існуючих сценаріях експерт може зрозуміти можливий розвиток каскадного ефекту, подивитися, які дії були виконані, характеристики компонентів та прийняти рішення. Представлення про характеристики електромережі можуть бути використані для формування бази знань, що полегшить процес розуміння предметної області та створить можливість для подальшого аналізу.

Обробка послідовності станів електромережі. При формалізації сценаріїв каскадних ефектів в електромережі у вигляді послідовностей графів з часовою складовою використовуються статичні графи – структура графу не змінюється із часом, але ознаки змінюються. При аналізі послідовностей станів електромережі до графової нейронної мережі додається блок LSTM (Long Short-Term Memory). Оскільки сценарії можуть складатися з різної кількості кроків, то доцільно використовувати маску при вирівнюванні довжини послідовностей та ігнорувати доповнені дані при роботі моделі. Це доповнення дозволяє формувати представлення про послідовність станів електромережі.

Висновки. В результаті проведення дослідження було розроблено метод для аналізу та порівняння сценаріїв каскадних ефектів в критичній інфраструктурі на прикладі електромережі. Для вивчення зв'язків та впливу параметрів компонентів в електромережі було розроблено шар графової нейронної мережі, що покращує точність роботи моделі автоенкодера формування представлення про стан роботи електромережі. Пошук схожих сценаріїв, використовуючи косинус подібності, потенційно дозволяє покращити процес прийняття рішень на основі існуючого досвіду. Розроблений метод може працювати з різним рівнем деталізації сценаріїв: порівнювати один конкретний крок сценаріїв чи послідовності кроків в сценаріях, що забезпечує його адаптивність до вхідних даних в залежності від цілей користувача.

Рис. 4. Графік зміни помилки під час тренування моделі (Embedding_Size = 45) на даних для тренування та валідації, використовуючи розроблений шар для графової мережі в енкодері

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Список використаних джерел:

1. Косинус подібності. URL: https://uk.wikipedia.org/wiki/Косинус_подібності
2. Робота енергосистеми України станом на 20 травня 2022 року. URL: https://www.kmu.gov.ua/news/robota-energosistemi-ukrayini-stanom-na-20-travnya-2022-roku
3. Робота енергосистеми України на 1 червня 2022 року. URL: https://www.kmu.gov.ua/news/robota-energosistemi-ukrayini-na-1-chervnya-2022-roku
4. Робота енергосистеми України на 10 червня 2022 року. URL: https://mspu.gov.ua/news/robota-energosistemi-ukrayini-na-10-chervnya-2022-roku
5. Сенченко В. Р., Бойченко А. В., Коваль О. В., Бисько Р. М., Хоменко О. М. Огляд методів і технологій сценарного аналізу каскадних ефектів. Реєстрація, зберігання і обробка даних, 2024. Том 26. № 1. С. 24–54. DOI 10.35681/1560-9189.2024.26.2.316908
6. Сенченко В. Р., Бойченко А. В., Коваль О. В., Хоменко О. М. Методологія і архітектура платформи моделювання взаємозалежностей у критичних інфраструктурах при виникненні каскадних ефектів. Реєстрація, зберігання і обробка даних, ІПРІ НАН України, 2025. Том 27. № 1. С. 28–41. DOI: 10.35681/1560-9189.2025.27.1.335624
7. Хоменко О. М., Сенченко В. Р., Коваль О. В. Мережевий підхід при дослідженні каскадних ефектів критичних інфраструктур. Реєстрація, зберігання і обробка даних, 2024. Том 26. № 2. С. 44–72. DOI 10.35681/1560-9189.2024.26.2.316908
8. Chadaga S., Wu X., Modiano E. Power Failure Cascade Prediction using Graph Neural Networks. https://doi.org/10.48550/arXiv.2404.16134
9. Cuadra L, Salcedo-Sanz S, Del Ser J, Jiménez-Fernández S, Geem ZW. A Critical Review of Robustness in Power Grids Using Complex Networks Concepts. *Energies*. 2015. 8(9), 9211–9265. https://doi.org/10.3390/en8099211
10. Di Nardo A, Giudicianni C, Greco R, Herrera M, Santonastaso GF. Applications of Graph Spectral Techniques to Water Distribution Network Management. *Water*. 2018. 10(1), 45. https://doi.org/10.3390/w10010045
11. Dubey S. R., Singh S. K., Chaudhuri B. B. Activation Functions in Deep Learning: A Comprehensive Survey and Benchmark. https://doi.org/10.48550/arXiv.2109.14545
12. Ellens W., Kooij R. E., Graph measures and network robustness. https://doi.org/10.48550/arXiv.1311.5064
13. Gilmer J., Schoenholz S. S., Riley P. F., Vinyals O., Dahl G. E. Neural Message Passing for Quantum Chemistry. https://doi.org/10.48550/arXiv.1704.01212
14. Guo Z., Sun K., Su X., Simunovic S., "A review on simulation models of cascading failures in power systems," *in iEnergy*, vol. 2, no. 4, pp. 284–296, December 2023, doi: 10.23919/IEN.2023.0039
15. Hines P., Cotilla-Sanchez E., Blumsack S., Do topological models provide good information about electricity infrastructure vulnerability? https://doi.org/10.48550/arXiv.1002.2268
16. Korkali M., Veneman J. G., Tivnan B. F., Hines P. D. H. Reducing Cascading Failure Risk by Increasing Infrastructure Network Interdependency. https://doi.org/10.48550/arXiv.1410.6836
17. Mattsson L.-G., Jenelius E. Vulnerability and resilience of transport systems a discussion of recent research. *Transportation Research Part A: Policy and Practice*, vol. 81, pp. 16–34, 2015. https://doi.org/10.1016/j.tra.2015.06.002
18. Naeem Md Sami, Mia Naeini. Machine Learning Applications in Cascading Failure Analysis in Power Systems: A Review. https://doi.org/10.48550/arXiv.2305.19390
19. Noebels M., Preece R., Panteli M., "AC Cascading Failure Model for Resilience Analysis in Power Networks," *in IEEE Systems Journal*, vol. 16, no. 1, pp. 374–385, March 2022, doi: 10.1109/JSYST.2020.3037400
20. Northeast blackout of 2003. URL: https://en.wikipedia.org/wiki/Northeast_blackout_of_2003
21. Oehlers M, Fabian B. Graph Metrics for Network Robustness—A Survey. Mathematics. 2021. 9(8), 895. https://doi.org/10.3390/math9080895
22. Rinaldi S. M., Peerenboom J. P., Kelly T. K., "Identifying, understanding, and analyzing critical infrastructure interdependencies", *in IEEE Control Systems Magazine*, vol. 21, no. 6, pp. 11–25, Dec. 2001, doi: 10.1109/37.969131
23. Rusch T. K., Bronstein M. M., Mishra S. A Survey on Oversmoothing in Graph Neural Networks. https://doi.org/10.48550/arXiv.2303.10993
24. Varbella A., Gjorgiev B., Sansavini G. Geometric deep learning for online prediction of cascading failures in power grids. *Reliability Engineering & System Safety*, Volume 237, 2023. https://doi.org/10.1016/j.ress.2023.109341
25. Waikhom L., Patgiri R. Graph Neural Networks: Methods, Applications, and Opportunities. https://doi.org/10.48550/arXiv.2108.10733
26. Wu Z., Pan S., Chen F., Long G., Zhang C., Yu P. S. A Comprehensive Survey on Graph Neural Networks. https://doi.org/10.48550/arXiv.1901.00596
27. 2003 Italy blackout. URL: https://en.wikipedia.org/wiki/2003_Italy_blackout
28. 2012 India blackouts. URL: https://en.wikipedia.org/wiki/2012_India_blackouts
29. 2016 South Australian blackout. URL: https://en.wikipedia.org/wiki/2016_South_Australian_blackout

Дата надходження статті: 25.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.415.5:004.8
DOI https://doi.org/10.32689/maup.it.2025.3.25

Михайло ХОМЧАК
аспірант факультету комп'ютерних наук та технологій,
Державний університет «Київський авіаційний інститут»,
mykhailo.khomchak@gmail.com
***ORCID:** 0009-0000-4127-556X*

Сергій ГНАТЮК
доктор технічних наук, професор,
Державний університет «Київський авіаційний інститут»,
s.gnatyuk@kai.edu.ua
***ORCID:** 0000-0003-4992-0564*

МЕТОД СТРУКТУРОВАНОГО ВПРОВАДЖЕННЯ ХМАРНОЇ ІНФРАСТРУКТУРИ

Анотація. *Українські підприємства дедалі активніше впроваджують хмарні сервіси (ХС) як основу бізнес-процесів. Проте вибір оптимальної конфігурації ХС має стратегічне значення, оскільки у підсумку визначить рівень витрат на ІТ-інфраструктуру, продуктивність і масштабованість ІТ систем. А також, додатково на безпеку даних й відповідність регуляторним вимогам. Проте процес прийняття рішень у цій сфері є надзвичайно складним. Це пов'язано багатокритеріальним характером задачі. Особи яка приймає рішення у процесі вибору ХС зазвичай оцінює суперечливі вимоги. Класичні методи багатокритеріальної оптимізації не завжди здатні врахувати суб'єктивні пріоритети та нечітко сформульовані вимоги («прийнятна вартість», «висока безпека», «достатня масштабованість»). Відповідно це знижує практичну придатність такої оптимізації.*

***Метою даної роботи** є синтез моделей, які здатні поєднати формальні кількісні показники та експертні судження, забезпечуючи збалансований вибір.*

***Методологія.** У статті запропоновано інтегровану модель, яка об'єднує апарат нечіткої логіки (НЛ) з сучасними еволюційними алгоритмами багатокритеріальної оптимізації (NSGA-III та MOEA/D). Для оцінювання альтернатив застосовано систему нечітких функцій належності та механізм агрегації за методом Мамдані. Це дозволило адекватно формалізувати якісні та нечіткі критерії. На етапі оптимізації формуємо множину Парето-оптимальних рішень, яка відобразить компроміс між різними вимогами. Для підвищення інтерпретованості та зручності вибору в ході дослідження отримано Парето-множину додатково ранжовано за допомогою методу нечіткого аналізу ієрархій (Fuzzy AHP – FAHP) та функції бажаності. Запропонована модель в цілому забезпечує комплексне врахування як техніко-економічних параметрів ХС, так і суб'єктивних пріоритетів особи, яка приймає рішення. Ефективність моделі підтверджено обчислювальним експериментом (ОЕ).*

***Наукова новизна.** Результати ОЕ продемонстрували покращене покриття Парето-фронту та вищу якість інтерпретації рішень у порівнянні з традиційними методами оптимізаціх структури ХС для підприємств.*

***Висновки.** Представлена модель може бути використана як інструмент підтримки прийняття рішень у сфері управління ІТ-інфраструктурою підприємств, сприяючи підвищенню обґрунтованості та адаптивності вибору ХС.*

***Ключові слова:** хмарні сервіси, багатокритеріальна оптимізація, функції належності, NSGA-III, MOEA/D, Парето-оптимальність, еволюційні алгоритми, модель вибору.*

Mykhailo KHOMCHAK, Sergiy GNATYUK. STRUCTURED METHOD OF CLOUD INFRASTRUCTURE IMPLEMENTATION

Abstract. *Ukrainian enterprises are increasingly adopting cloud services (CS) as the foundation of their business processes. However, selecting the optimal CS configuration has strategic importance, as it ultimately determines the level of IT infrastructure costs, performance, and scalability of IT systems, as well as data security and regulatory compliance. The decision-making process in this field is extremely complex due to its multi-criteria nature. A decision-maker involved in the selection of CS must usually evaluate conflicting requirements. Classical methods of multi-criteria optimization are not always capable of accounting for subjective priorities and vaguely defined requirements (e.g., "acceptable cost," "high security," "sufficient scalability"). Consequently, this reduces the practical applicability of such optimization approaches.*

***The purpose of this study** is to synthesize models capable of combining formal quantitative indicators and expert judgments, ensuring a balanced decision-making process.*

***Methodology.** The article proposes an integrated model that combines fuzzy logic (FL) with modern evolutionary algorithms for multi-objective optimization (NSGA-III and MOEA/D). To evaluate alternatives, a system of fuzzy membership functions and an aggregation mechanism based on the Mamdani method were applied. This approach enabled the formalization of qualitative and vague criteria. At the optimization stage, a set of Pareto-optimal solutions was generated to represent the trade-offs between different requirements. To enhance interpretability and convenience of selection, the obtained Pareto set was additionally ranked using the Fuzzy Analytic Hierarchy Process (Fuzzy AHP – FAHP) and the desirability function. Overall,*

the proposed model provides a comprehensive consideration of both the technical-economic parameters of cloud services and the subjective preferences of the decision-maker. The effectiveness of the model was validated through a computational experiment (CE).

Scientific novelty. *The results of the CE demonstrated improved Pareto front coverage and higher interpretability of decisions compared to traditional optimization methods for CS structure selection in enterprises.*

Conclusions. *The presented model can be used as a decision-support tool in enterprise IT infrastructure management, enhancing the justification and adaptability of cloud service selection.*

Key words: *cloud services, multi-objective optimization, membership functions, NSGA-III, MOEA/D, Pareto optimality, evolutionary algorithms, selection model.*

Постановка проблеми. В умовах цифрової трансформації компаній та підприємств хмарні сервіси (далі – ХС) все частіше розглядають як головну складову корпоративної ІТ-інфраструктури. Вибір оптимальної конфігурації ХС має стратегічне значення. Це пов'язано з тим, що архітектура ХС впливає на вартість володіння ІТ-ресурсами, продуктивність бізнес-процесів, рівень безпеки даних, відповідність регуляторним вимогам та гнучкість масштабування. Однак прийняття таких рішень є складним через наявність численних суперечливих критеріїв. Також на вибір впливають невизначеність зовнішнього середовища та суб'єктивний характер експертних оцінок доцільності компонентів архітектури ХС [4].

Попередні результати дослідження, спрямовані на формалізацію математичної моделі вибору ХС на основі інтеграції нечіткої логіки та методів багатокритеріальної оптимізації, були представлені у [3]. У тій роботі було зосереджено увагу на методологічних аспектах побудови моделі та описі процедур оптимізації. У даній статті ми розвиваємо запропонований підхід, роблячи акцент на розширеному статистичному аналізі отриманих результатів, емпіричній апробації методу та інтерпретації практичних наслідків для підприємств.

Новизна проведеного дослідження полягає у тому, що на відміну від попередньої роботи [3], зосередженої на побудові та формалізації моделі, у цій статті вперше подано результати її прикладної апробації. Виконано комплексний статистичний аналіз рішень, отриманих із застосуванням алгоритмів NSGA-III та MOEA/D, що дало змогу виявити відмінності у рівномірності розподілу Парето-фронту та чутливості моделі до варіації вагових коефіцієнтів. Практична значущість дослідження підтверджується можливістю використання методу для підтримки рішень у сфері вибору та налаштування ХС підприємств, де необхідно забезпечити баланс між вартістю, надійністю та масштабованістю системи.

Постановка проблеми. Підприємства, які впроваджують ХС, стикаються з необхідністю вибору оптимальної конфігурації серед великої кількості альтернативних пропозицій. Цей вибір ускладнено багатокритеріальністю задачі. Тобто, одночасно потрібно враховувати вартість, продуктивність, масштабованість, рівень безпеки, надійність та інші параметри, які часто мають суперечливий характер. Додаткову складність створює невизначеність вхідних даних і суб'єктивність експертних оцінок, які виражено у нечіткій формі вимог до ХС. Традиційні методи оптимізації повною мірою не забезпечують достатньої гнучкості для роботи з такими умовами. Отже, постає проблема розробки інтегрованої моделі, здатної поєднати нечітку формалізацію критеріїв з еволюційними методами багатокритеріальної оптимізації для формування множини обґрунтованих альтернатив та подальшого ранжування рішень з урахуванням пріоритетів особи, що приймає рішення.

Огляд попередніх досліджень. За останні роки проблематика вибору ХС набула значної уваги у дослідженнях науковців різних країн. Це зокрема зумовлено багатокритеріальним характером прийняття рішень у цій задачі. У низці робіт здійснено систематизацію наявних підходів. Зокрема, у [4] проведено огляд методів від класичних моделей багатокритеріальної оптимізації до евристичних алгоритмів і нечітких методів. Автори відзначили, що поєднання MCDM та еволюційних алгоритмів все ще залишилися недостатньо дослідженим напрямом, зокрема для задач високої розмірності. Окрема група досліджень присвячена застосуванню нечіткої логіки (НЛ) для формалізації якісних та експертних критеріїв. Так у [3] запропоновано нечітку модель оцінки довіри до постачальників ХС, де враховано показники продуктивності та безпеки. Подібні ідеї використані у [19], де автори запропонували нечітко-ентропійний підхід для оцінки надійності та стану довіри до ХС. У [16] розглянуто інтеграцію НЛ з елементами комп'ютерного інтелекту для адаптивного налаштування правил оцінювання. Перевагою таких підходів, на думку авторів, є висока інтерпретованість та можливість врахування суб'єктивних оцінок. Проте їхнім недоліком залишається обмеженість у пошуку множини оптимальних альтернатив у складному критеріальному просторі. Для подолання цього обмеження дослідники активно застосовують еволюційні багатокритеріальні алгоритми. У [13] використано гібридний NSGA-III-GKM++, який забезпечив рівномірне покриття фронту Парето у задачах брокерської оптимізації ресурсів. У [9] досліджено застосування NSGA-III у задачі розміщення віртуальних машин

з урахуванням енергоспоживання та затримок. Схожі результати подані у [6], де NSGA-III автори використали для оптимізації композиції ХС з урахуванням асоціаційних витрат. У [7] подано узагальнений огляд методології NSGA-III та показано її сильні сторони й обмеження.

Поряд із NSGA-III дослідники активно розвивають алгоритми сімейства MOEA/D. У [17] описано двоступеневу еволюційну стратегію, яка підвищує збіжність алгоритму. В [8] представлено удосконалений варіант IMOEA/D з адаптивними операторами мутації. Обидва підходи продемонстрували високу придатність для багатовимірних задач оптимізації. Проте вони потребують адаптації до випадків вибору ХС із нечіткими критеріями. Окремий напрям становлять гібридні моделі. Такі моделі поєднують нечітку логіку з метаевристиками. Зокрема у [15] запропоновано метод вибору постачальників CSP. Метод поєднав нечіткі оцінки з алгоритмом firefly. У [10] наголошено на важливості адаптивного налаштування нечітких систем із залученням алгоритмів комп'ютерного інтелекту. Дослідження [15], [10] підтвердили доцільність інтеграції НЛ та еволюційних підходів у єдину модель. Ефективним інструментом підтримки прийняття рішень є класичні методи MCDM. Зокрема це Fuzzy AHP (FAHP). У [5] і [18] запропоновано комбінації FAHP з TOPSIS для ранжування ХС. На думку авторів, такий підхід дозволить інтегрувати експертні оцінки в моделі на основі НЛ. У [14] представлено застосування сучасних MCDM-методик для пріоритизації CSP. Сильними сторонами цих підходів є простота та прозорість. Проте, вони не завжди забезпечують достатню різноманітність альтернатив у складних задачах.

Проведений аналіз показав, що сучасні дослідження зосереджені на трьох головних напрямах: формалізація критеріїв через нечіткі функції та ентропійні оцінки; застосування еволюційних алгоритмів (NSGA-III, MOEA/D) для знаходження Парето-оптимальних множин; використання MCDM (зокрема FAHP) як інтерпретаційних інструментів для експертної підтримки. Проте комплексні моделі, які інтегрують усі ці підходи одночасно, залишилися малодослідженими. Саме тому у даній роботі запропоновано інтеграцію нечіткої логіки з еволюційними методами NSGA-III та MOEA/D, а також застосування FAHP для валідації й пояснення результатів.

Мета дослідження – розробити інтегровану модель вибору ХС для підприємства (далі ХС), що поєднує нечітку логіку та еволюційні методи багатокритеріальної оптимізації з подальшим ранжуванням рішень для підтримки обґрунтованого прийняття рішень..

Методи та моделі. Задача вибору оптимального набору ХСП формалізована як задача багатокритеріальної оптимізації з нечіткими функціями цілі.

Нехай $X = \{x_1, x_2, \ldots x_n\}$ – множина потенційних конфігурацій ХС, де $\{x_i\}$ – кожна конфігурація x_i являє собою комбінацію ХС з різних підкатегорій (обчислення, зберігання, безпека тощо). Потрібно знайти таке $x^* \in X$, яке оптимізує векторну функцію цілі з урахуванням нечіткості критеріїв:

$$x^* \in X : \mathbf{F}(x^*) = \left(f_1(x), f_2(x), \ldots f_k(x)\right) - \text{Pareto} - \text{оптимум}, \quad (1)$$

де $f_i(x)$ – функція оцінки за j-м критерієм; $j = 1, \ldots, k, k = 7$ – кількість критеріїв у задачі.

Розглянемо наступні сім критеріїв для задачі вибору оптимального набору ХС для великого підприємства (або далі приймаємо наступну абревіатуру ХСП – Хмарні Сервісні Платформи): $f_1(x)$ – вартість ХСП (мінімізуємо (або)); $f_2(x)$ – масштабованість ХСП (максимізуємо); $f_3(x)$ – продуктивність ХСП (максимізуємо – ()); $f_4(x)$ – рівень безпеки (або безпека) ХСП (максимізуємо); $f_5(x)$ – надійність ХСП (максимізуємо); $f_6(x)$ – відповідність ХСП стандартам (максимізуємо); $f_7(x)$ – інтеграція ХСП з іншими ХС (максимізуємо).

Зазначимо, що набір критеріїв $\{f_1(x), \ldots, f_7(x)\}$ не є довільним. Він є результатом так званої таксономії ознак, про що йшлося у [2]. Або факторного аналізу домену (в нашому випадку – характеристик ХС, релевантних для підприємства). З погляду математичного моделювання, цей процес можна описати як побудову простору критеріїв $\mathbb{R}^k$, у якому кожен критерій відповідає окремому виміру (осьовому напряму) простору рішень.

Запишемо це формально. Нехай $X \in \mathbb{R}^k$ – простір допустимих конфігурацій ХСП.

Тоді критерій $f_i : X \to \mathbb{R}$ – це функція, яка проектує кожне рішення $x \in X$ на шкалу оцінки за j-м параметром. І, відповідно, тоді таксономія тут – це процедура класифікації та структурування характеристик системи за семантичними або функціональними групами, яка в нашій оптимізаційній моделі подана у вигляді вектора оцінювання.

Зазначмо, що в рамках дослідження саме з математичної точки зору ми прагнемо, щоби:

– множина критеріїв є повною (тобто покриває всі визначальні аспекти ефективності ХСП);

– критерії є незалежними або слабо корельованими (тобто кожен критерій вносив у модель унікальну інформацію);

– додавання нових критеріїв не змінювало істотно структуру Парето-фронту (тобто не призводило до домінування одного критерію над іншими).

Зазначимо, що у реальних умовах прийняття рішень у сфері ІТ, приміром при виборі ХСП, рідко можна знайти чітко визначені та однозначні критерії ефективності. Більшість вимог замовників – таких як «висока безпека», «достатня масштабованість», «помірна вартість» – мають якісний, експертний або оцінковий характер. Ці характеристики не завжди можна точно формалізувати через жорсткі числові межі, тому використання традиційної багатокритеріальної оптимізації з жорсткими функціями цілі не завжди є адекватним для таких задач.

Вважаємо, нечітка логіка (fuzzy logic), на відміну від класичної бінарної логіки, дозволяє в межах задачі дослідження моделювати нечіткість, невизначеність і суб'єктивні уявлення експертів. До речі, кожен критерій описуємо не лише як числову функція $f_j(x)$, але і як функцію належності $\mu_j\left(f_j(x)\right)$. Тобто $\mu_j\left(f_j(x)\right)$ відображає ступінь задоволення підприємством певного критерію.

З математичної точки зору, усі критерії, що максимізуємо ($f_2(x) - f_7(x)$, окрім $f_1(x)$), мають однакову трикутну або трапецієподібну функцію належності, яка монотонно зростає в інтервалі від нижньої межі (незадовільне значення) до бажаного рівня, після чого ступінь належності приймає максимальне значення 1. Тобто, для кожного такого критерію j ($j \in \{2, 3, 4, 5, 6, 7\}$) можна задати функцію належності у вигляді [11; 12]:

$$\mu_j\left(f_j(x)\right) = \begin{cases} 1, & f_j(x) \le a_j, \\ \dfrac{f_j(x) - a_j}{b_j - a_j}, & a_j < f_j(x) < b_j, \\ 0, & f_j(x) \ge b_j, \end{cases} \tag{2}$$

де a_j – нижній допустимий рівень критерію; b_j – бажаний рівень задоволення критерію.

Такий загальний вигляд є універсальним для всіх критеріїв, які підлягають максимізації, тобто ($f_2(x) - f_7(x)$. Розгорнутий запис наведено тільки для критерію $f_4(x)$), («безпека»), див. вираз (2) як типового представника цієї групи, щоб не перевантажувати текст статті однаковими за структурою формулами.

Слід зазначити, що одним із основних елементів математичної моделі вибору ХСП виходячи з нечіткої логіки є побудова агрегованої оцінки якості кожної потенційної конфігурації. Така оцінка дозволяє перейти від багатокритеріальної нечіткої постановки задачі до єдиної узагальненої метрики, яка придатної до використання в алгоритмах оптимізації. Тобто, у рамках моделі для кожного критерію доцільно побудувати відповідну функцію належності $\mu_j\left(f_j(x)\right)$, яка відображає ступінь відповідності значення критерію $f_j(x)$ бажаному рівню. Проте для прийняття рішення недостатньо розглядати ці функції ізольовано. Потрібно буде об'єднати їх у єдину інтегральну оцінку, яка відображає узагальнену «якість» конфігурації ХСП із урахуванням усіх критеріїв (як ми прийняли раніше) одночасно.

І саме для цієї мети вводиться вираз (3). Тоді загальна функція оцінки матиме наступний вигляд:

$$\mu(x) = \bigwedge_{j=1}^{k} \mu_j\left(f_j(x)\right), \tag{3}$$

де $\vee$ – означає мінімальне значення серед всіх критеріїв (логіка Мамдані). Або можна використовувати агреговану оцінку через середнє:

$$\mu(x) = \frac{1}{k}\sum_{j=1}^{k} \mu_j\left(f_j(x)\right), \tag{4}$$

де k – кількість критеріїв оптимізації ХСП.

Зазначимо, що у процесі побудови узагальненої оцінки якості конфігурації ХС ми використовуємо підхід, заснований на логіці Мамдані [12]. Тому виникає потреба шукати Парето-оптимальний розв'язок – множину альтернатив, які не є домінованими одна однією.

Алгоритм NSGA-III є подальшим розвитком NSGA-II, адаптованим до задач з великою кількістю критеріїв (більш трьох). Його ефективність базується на наступних положеннях [6]:

– нелінійне сортування. Тобто популяція ділиться на фронти Парето-оптимальності. Рішення, які не домінуються жодним іншим, потрапляють до першого фронту PF_1, потім визначаються наступні рівні.

– reference points. Відповідно, використовуємо заздалегідь задані точки у просторі критеріїв для підтримання різноманіття та рівномірного покриття фронту.

формальне правило селекції. Тобто відбір особин відбувається з урахуванням не лише домінування, але й відстані до reference points, що зменшує ризик надмірної кластеризації рішень у певних ділянках фронту.

Математично, алгоритм забезпечує пошук множини рішень

$$PF = \left\{ x \in X \middle| \nexists x' \in X : x' \succ x \right\}, \tag{5}$$

де відношення Парето-домінування визначається як:

$$x' \succ x \Leftrightarrow \forall j, f_j(x), \exists j : f_j(x') > f_j(x). \tag{6}$$

Завдяки вбудованій нормалізації та reference vectors, NSGA-III дозволить нам, ефективно працювати навіть при високій розмірності простору критеріїв (у нашій задачі – сім критеріїв).

Як альтернативний метод, так саме в статті розглядається варіант використання алгоритму MOEA/D [17].

На відміну від NSGA-III, алгоритм MOEA/D ґрунтується на іншій концепції – декомпозиції багатокритеріальної задачі [9; 6; 7; 17]. Він полягає у наступному:

Замість оперування з усім вектором критеріїв, MOEA/D розбиває задачу на *N* скалярних підзадач. Далі кожної підзадачі оптимізуємо функцію (7):

$$g^{(i)}(x) = \max_j \left[\lambda_j^{(i)} \cdot \left| f_j(x) - z_j^* \right| \right], \tag{7}$$

де $\lambda_j^{(i)}$ – ваговий коефіцієнт *j*-го критерію у *i*-й задачі; z_j^* – еталонне (найкраще знайдене) значення за критерієм j.

Оптимізація кожної скалярної функції ведеться незалежно з частковим обміном інформації між сусідніми підзадачами.

Це дозволяє краще фокусуватись на окремих ділянках Парето-фронту та зменшити обчислювальні витрати. Як відмічалося у [17] MOEA/D ефективний при великих просторах рішень і складних ландшафтах цільової функції, де пряме сортування (як у NSGA-III) стає обчислювально складним.

Відповідно, кожен алгоритм (тобто NSGA-III та MOEA/D) працює з множиною кандидатів x_i, обчислює $\mu_j(f_j(x))$, і формує Парето-фронт:

$$PF = \left\{ x \in X \middle| \nexists x' : \forall j, f_j(x') > f_j(x), \exists j : f_j(x') > f_j(x) \right\}. \tag{8}$$

Вираз (8) це множина конфігурацій ХС, для яких не існує альтернативи, що була б кращою одночасно за всіма критеріями для підприємства. Кожен елемент $x \in PF$ представляє потенційно оптимальне рішення для підприємства, залежно від його пріоритетів та умов.

Розглянуті еволюційні багатокритеріальні алгоритми – NSGA-III та MOEA/D – забезпечують в нашому досліджені ефективне знаходження множини Парето-оптимальних рішень для задачі вибору ХСП. Однак, для цілісної оцінки якості та обґрунтованості обраних оптимізаційних рішень ХСП доцільно застосувати класичні підходи до багатокритеріального аналізу. Зокрема можна використати метод ієрархій [5] (AHP – Analytic Hierarchy Process). AHP тоді виступає як базовий рівень порівняння, а також є додатковим засобом для валідації та пояснення отриманих результатів. Алгоритм AHP дозволяє формалізувати суб'єктивні експертні оцінки у вигляді попарних порівнянь критеріїв, що є визначальним на початкових етапах прийняття рішень по розгортанню ХС на підприємстві.

Тому далі розглянемо застосування методу AHP для побудови базової ієрархічної моделі оцінювання та порівняння альтернатив, що дозволить сформувати початкові пріоритети та надалі використати їх як відправну точку або орієнтир для еволюційних методів.

Тобто, для порівняння з еволюційними підходами в нашому досліджені застосовується метод Fuzzy AHP. Наведемо етапи роботи цього методу коли експерти формують попарні порівняння критеріїв у вигляді нечітких чисел;

будуємо матрицю парних порівнянь $\tilde{A}$, з якої виводиться вектор ваг $\tilde{w}$,

виконуємо оцінку конфігурації, згідно виразу (9):

$$F_{AHP}(x) = \sum_{j=1}^{k} \tilde{w}_j \cdot \mu_j(f_j(x)), \tag{9}$$

де $\tilde{w}_j$ – нечітка вага критерію *j*; $\mu_j(f_j(x))$ – нечітка оцінка за критерієм *j*.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

В рамках поточного дослідження запропоновано модель, яка дозволяє відображати уподобання підприємств при виборі ХС через нечіткі функції належності. Формалізовано 7 основних, що відображають реальні потреби бізнесу. А інтеграція нечіткої логіки з NSGA-III та MOEA/D дозволяє отримати множину Парето-оптимальних рішень та провести аналіз компромісів між альтернативами.

Метод Fuzzy AHP використовуємо для порівняння точнісних та евристичних стратегій прийняття рішень.

Відмітимо, що отримана в результаті застосування еволюційних багатокритеріальних алгоритмів (NSGA-III та MOEA/D) множина Парето-оптимальних рішень являє собою набір конфігурацій ХС, кожна з яких не домінує іншу за всіма критеріями одночасно. Така множина дозволяє відобразити компромісний характер задачі, в якій не існує єдиного найкращого рішення, а лише спектр альтернатив, кожна з яких є оптимальною у певному сенсі – скажімо, має нижчу вартість, але гіршу інтеграцію ХСП або безпеку тощо.

Однак множина Парето сама по собі не дає остаточної відповіді на питання, яке саме рішення ХСП має бути впроваджене на практиці. З прикладної точки зору, особа, що приймає рішення (далі ОПР), повинна зробити свідомий вибір, який бере до відома стратегічні, технічні або економічні пріоритети конкретного підприємства. Це релевантно, скажімо, в умовах, коли підприємство має обмежені ресурси, власні нормативні вимоги (як у банківській сфері) або внутрішні політики, що впливають на допустимість чи бажаність певних рішень.

З цією метою доцільно ввести поняття множини опорних (еталонних) рішень – підмножини Парето-фронту, яка найкраще узгоджується з пріоритетами, очікуваннями або експертними оцінками ОПР. Такі опорні рішення можуть бути побудовані шляхом формалізації переваг ОПР у вигляді бажаних значень критеріїв (еталонного профілю) або за допомогою функції бажаності чи функції корисності.

У цьому дослідженні розглянемо математичний підхід до побудови таких опорних рішень, а також формалізуємо процедуру оцінювання ступеня наближеності Парето-альтернатив до уподобань ОПР. Це дозволить здійснити подальше ранжування або остаточний вибір альтернативи, яка найкраще задовольняє вимоги конкретного підприємства.

Позначимо:

N – кількість альтернатив (кандидатних конфігурацій ХС);

$k = 7$ – кількість критеріїв;

$x_i \in X$, де $i = 1,\ldots,N$ – конфігурація ХСП;

$\mu_j\left(f_j\left(x_i\right)\right) \in [0,1]$ – ступінь задоволення за критерієм j для альтернативи x_i;

$\tilde{w}_j \in [0,1], \sum_{j=1}^{k} \tilde{w}_j = 1$ – ваги пріоритетів ОПР.

Тоді опорне рішення визначаємо як розв'язок, який максимізує агреговану функцію бажаності:

$$x^{ref} = arg\max_{x_i \in PF}\left(U\left(x_i\right) = \sum_{j=1}^{k} \tilde{w}_j \cdot \mu_j\left(f_j\left(x_i\right)\right)\right). \tag{10}$$

Ця функція бажаності відображає суб'єктивні пріоритети *ОПР* через ваги $\tilde{w}_j$. А ступені належності критеріїв забезпечують нормалізацію та уніфікацію шкал.

Доцільний й альтернативний підхід. ОПР визначає вектор бажаних рівнів задоволеності критеріїв

$$g = \left(g_1, g_2, \ldots, g_k\right) \in [0,1]^k,$$

після чого для кожної альтернативи обчислюємо відстань до бажаного профілю:

$$D\left(x_i, g\right) = \left(\sum_{j=1}^{k}\left(\mu_j\left(f_j\left(x_i\right)\right) - g_i\right)^2\right)^{1/2},$$
$$x^{ref} = arg\max_{x_i \in PF} D\left(x_i, g\right). \tag{11}$$

Вирази (10) та (11) дозволяють знайти рішення, найближче до індивідуального еталону, сформованого ОПР. Подібний підхід буде ефективний у випадках, коли пріоритети не можуть бути виражені у вигляді ваг, але відомі бажані рівні за критеріями. В обох випадках отримані опорні рішення можуть слугувати: для вибору остаточної конфігурації ХСП; для порівняння з результатами класичних методів (FAHP); для візуалізації й подальшого узгодження із зацікавленими сторонами.

У таких випадках підприємство (а саме ОПР) не проводить формального розподілу ваг, але чітко формулює, яких характеристик ХС воно прагне. Модель, яка використовує евклідову відстань до цього вектора бажаностей (10), дозволяє автоматично знайти рішення з Парето-множини, яке найбільше наближене до таких очікувань. Це підвищує інтерпретованість моделі та дозволяє залучати осіб без математичної підготовки до процесу вибору.

Побудова множини опорних рішень доповнює Парето-аналіз і забезпечує інтерпретованість та практичну реалізацію оптимізаційної моделі. Однак формування Парето-фронту саме по собі ще не дозволяє оцінити якість отриманих рішень у порівнянні з ідеальними характеристиками або між собою. У нашому дослідженні, де аналізуємо сім критеріїв $\{f_1(x),\ldots,f_7(x)\}$ оцінка якості отриманих Парето-рішень дозволить перевірити, наскільки добре множина результатів узгоджується з очікуваними або еталонними рішеннями. Основні метрики, які використовуємо в багатокритеріальній оптимізації містять [9; 6] – Hypervolume (HV); Inverted Generational Distance (IGD); Spacing Metric (SP); Coverage of Two Sets (C-metric).

Розглянемо кожну із них в нашої моделі та їхній загальний вплив на розв'язання задачі вибору ХС для підприємства.

Метрика Hypervolume (HV) обчислює об'єм (у *k*-вимірному просторі критеріїв $\{f_1(x),\ldots,f_7(x)\}$), який покриває множина Парето-рішень відносно заздалегідь заданої точки відсічення (reference point). У нашому випадку ця точка визначена як гірше припустиме значення по кожному з критеріїв $\{f_1(x),\ldots,f_7(x)\}$:

$$HV(S)=Volume\left(\bigcup_{x\in S}\left[f_1(x),r_1\right]\times\ldots\times\left[f_k(x),r_k\right]\right), \tag{12}$$

де *S* – множина Парето-рішень;

r_k – координата точки відсічення по кожному критерію $f_1(x),\ldots,f_7(x)$.

Оскільки $f_1(x)$ мінімізуємо, відповідна координата r_1 повинна бути максимально допустимою вартістю ХСП, а для інших критеріїв – мінімально прийнятним значенням. Чим більший HV, тим більше простору охоплює множина рішень – тобто більше різноманіття і краща апроксимація.

Метрика IGD [6] дозволяє оцінити, наскільки добре поточна множина Парето-рішень *S* наближається до «еталонного» Парето-фронту *P*, див. вираз (13):

$$IGD(P,S)=\frac{1}{|P|}\cdot\sum_{v\in P}\min_{s\in S}v-s, \tag{13}$$

де *P* – еталонна (reference) множина рішень, яка апроксимує справжній або очікуваний Парето-фронт;

S – множина рішень, отримана алгоритмом NSGA-III (NSGA-II) або MOEA/D), яку ми хочемо оцінити;

$v\in P$ – окрема точка (вектор критеріїв) з еталонної множини *P*;

$v-s$ – евклідова відстань між точками v та s у просторі критеріїв.

У нашому випадку еталонну множину *P* можна побудувати шляхом:

– або шляхом об'єднання всіх рішень, знайдених кількома алгоритмами (NSGA-III (NSGA-II) + MOEA/D або ін.);

– або ж через генерацію ідеального (гіпотетичного) Парето-фронту за експертними оцінками ;

– або використати високоякісні рішення з попередніх ітерацій/запусків. Тоді потрібно всі результати зберігати в окрему базу.

Отже, метрика IGD – це формальна, кількісна метрика, яка дозволяє оцінити наскільки добре конкретний алгоритм наблизився до бажаної множини рішень. Цей показник є корисним у нашому випадку, якщо ми зможемо побудувати набір контрольних точок *P*, скажімо, за допомогою рівномірного розбиття вздовж відомих меж критеріїв або використанням об'єднаного фронту з кількох запусків.

Метрика Spacing Metric (SP) [9] оцінює рівномірність розподілу рішень у множині *S*. Для цього обчислюється відхилення між сусідніми рішеннями [9]:

$$SP(S)=\sqrt{\frac{1}{|S|-1}\sum_{i=1}^{|S|}\left(d_i-\overline{d}\right)^2}, \tag{14}$$

де $d_i=\min_{j\neq i}f(x_i)-f(x_j)-$ мінімальна відстань до сусіднього рішення;

$\overline{d}$ – середнє значення d_i.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Ідеально рівномірний розподіл дає значення SP близьке до нуля. Зокрема, для задачі вибору ХСП, рівномірне покриття Парето-фронту дозволить підприємству бачити широкий спектр компромісів. Для прикладу, між вартістю та масштабованістю. Або між безпекою та інтеграцією.

Додатково для порівняння результатів NSGA-III та MOEA/D будемо застосувати Coverage of Two Sets (C-metric) [9].

$$C(A,B)=\frac{\left|\left\{x\in B\middle|\exists\, y\in A\colon y\ домінує\ x\right\}\right|}{|B|}, \tag{15}$$

де $C(A,B)$ – значення метрики покриття (coverage) для множин A і B, яке показує, яку частину рішень з множини B домінують рішення з множини A;

A – перша множина Парето-рішень (отримана, скажімо, алгоритмом NSGA-III);

B – друга множина Парето-рішень (отримана, до прикладу, алгоритмом MOEA/D);

$x\in B$ – окреме рішення (альтернатива, конфігурація ХС для підприємства) з множини B, тобто з множини, яку ми оцінюємо;

$y\in A$ – рішення з множини A, тобто множини, яка є «еталоном порівняння» в межах цієї метрики.

Отже, метрика C показує, яку частину рішень одного алгоритму (до прикладу, B – MOEA/D) домінує інший (A – NSGA-III). $C(A,B)$=1 означає повне домінування.

Усі перелічені метрики можуть бути обчислені шляхом синтезу нормалізованих значень функцій $f_i(x)$, або, що краще для нечіткої моделі, виходячи з функцій належності. Це дозволить при алгоритмічній реалізації моделі, провести зіставлення не тільки у просторах початкових значень, але й у просторі нечітких оцінок якості конфігурацій ХСП.

У теперішніх умовах функціонування компаній та підприємств, ХС розглядаємо не як одноразове інфраструктурне рішення, а як рухливий компонент стратегії цифрової трансформації. Пріоритети щодо вибору ХС можуть змінюватися залежно від стратегічних фаз розвитку підприємства, впливу зовнішнього ринкового середовища, зміни регуляторних вимог або внутрішньої реструктуризації.

Візуалізуємо опис моделі (1) – (10) у вигляді блок схеми алгоритму, див. рис. 1. Алгоритм, який реалізує метод структурованого впровадження хмарної інфраструктури у корпоративні ІТ-системи, відзначається ієрархічно-модульною побудовою, що забезпечує логічну узгодженість, структурну гнучкість та функціональну масштабованість. Така архітектура надає можливість при подальшій реалізації, як-от, у вигляді модулів СППР, трактувати кожен логічний компонент алгоритму не лише як послідовний етап, а як самодостатній модуль. Такий модуль є доцільним до реалізації, тестування його працездатності та наступного впровадження незалежно від інших частин.

Модульна структура алгоритму на рис. 1 забезпечує природну підтримку масштабування для ХС підприємств з різним рівнем складності. В тому числі, на великих підприємствах, де хмарна інфраструктура може охоплювати десятки різнорідних ХС, така гнучкість дає змогу налаштовувати модель не лише в ширину (шляхом додавання нових критеріїв (ми розглядали лише 7), джерел даних або сценаріїв впровадження). Але й у глибину – через деталізацію внутрішньої логіки оцінювання чи адаптацію поведінки критеріїв до змін у зовнішньому середовищі. Крім того, ієрархічна побудова алгоритму на рис. 1 ефективно реалізує паралельну або розподілене опрацювання даних. Приміром це доцільно на етапі формування та оцінки конфігурацій ХСП [1, 2]. Подібний є доцільним у разі роботи з великими обсягами даних або множинами кандидатних рішень, які вимагають багаторазових оцінок за різними критеріями. У цьому сенсі алгоритм на рис. 1 можна реалізувати як сукупність незалежних процесів з асинхронною передачею результатів. На нашу думку, це підвищує продуктивність СППР у масштабованих обчислювальних середовищах.

Значущою характеристикою алгоритму є здатність зберігати проміжні результати між етапами його виконання. Така властивість забезпечує не лише ефективність обчислень шляхом повторного використання вже отриманих оцінок, а й дає змогу проводити аналітичний аудит процесу прийняття рішень. Збереження міжетапної інформації дозволить відстежувати причиново-наслідкові зв'язки між характеристиками вхідних даних, прийнятими управлінськими параметрами та отриманими рекомендаціями щодо вибору конфігурацій ХСП. А це, своєю чергою, забезпечить прозорість системи вибору ХСП. А також підтримує процедури верифікації та підвищує рівень довіри з боку осіб, що приймають рішення. Завдяки зазначеним властивостям, розроблений в рамках дослідження метод та алгоритм набувають якості універсального та адаптивного інструмента. Такий інструмент, за нашими міркуваннями, доцільно інтегрувати у складні корпоративні платформи управління ІТ-інфраструктурою, як-от у рамках стратегічного планування хмарної трансформації для великих підприємств України.

Поданий алгоритм складається з трьох взаємопов'язаних блоків. Зазначимо, що ми продовжили дослідження, результати яких подано в [1]. На першому етапі формуємо систему нечітких функцій

належності для кожного критерію оцінювання. Це дало змогу відобразити лінгвістичні вимоги типу «висока безпека» чи «прийнятна вартість» у кількісному вигляді та забезпечити порівнянність різнорідних показників. Агрегація здійснювалася за допомогою механізму Мамдані. Це дозволило інтегрувати експертні судження й отримати узагальнені оцінки для кожного постачальника послуг.

Другий етап пов'язаний із застосуванням еволюційних алгоритмів багатокритеріальної оптимізації NSGA-III та/або MOEA/D.

На третьому етапі виконуємо ранжування отриманої Парето-множини. Для цього застосовуємо методи FAHP і функції бажаності. Це дозволило інтегрувати суб'єктивні пріоритети особи, що приймає рішення, у процес формування остаточних рекомендацій вибору ХС. Отже, наш підхід поєднав точність еволюційних методів із високою інтерпретованістю результатів. У підсумку запропонована модель забезпечує комплексне вирішення задачі вибору ХС.

Для перевірки працездатності та ефективності запропонованої моделі проведено обчислювальний експеримент. Модель та наведений псевдокод реалізовано на мові програмування Python. На основі сформованих вхідних даних згенеровано множину Парето-оптимальних рішень, які далі проаналізовано за допомогою метричних показників та методів ранжування. Результати експерименту подано у вигляді графіків на (рис. 2, 3), ілюструють поведінку алгоритмів NSGA-III та MOEA/D, а також демонструють якість отриманого Парето-фронту та стабільність роботи моделі.

На представленому графіку «Функції належності для критеріїв (Парето vs Неоптимальні)», див. рис. 2, відображено результати багатокритеріальної оптимізації ХСП. Графік на рис. 2 побудовано у вигляді паралельних координат. Це надає можливість візуально порівняти характеристики різних конфігурацій. Червоним кольором позначено Парето-оптимальні рішення («Так»). Синім кольором – неоптимальні конфігурації («Ні»). Парето-оптимальні рішення ілюструють вищі значення функцій належності за більшістю критеріїв.

Представлена на (рис. 3) візуалізація підтверджує ефективність запропонованого методу для виявлення оптимальних конфігурацій ХС великих підприємств. Аналіз функцій належності дозволяє

Рис. 1. Спрощена блок-схема алгоритму методу структурованого впровадження хмарної інфраструктури (СВХІ)

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

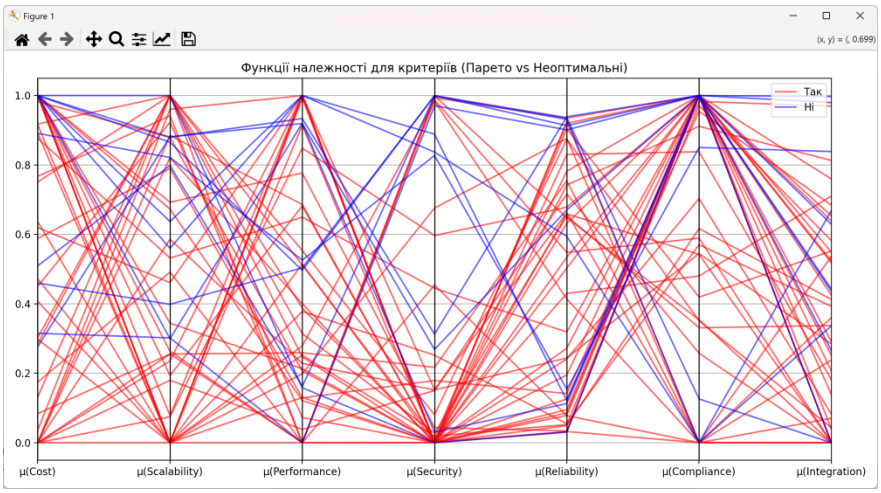


Рис. 2. Графіки «Функції належності для критеріїв (Парето vs Неоптимальні)»

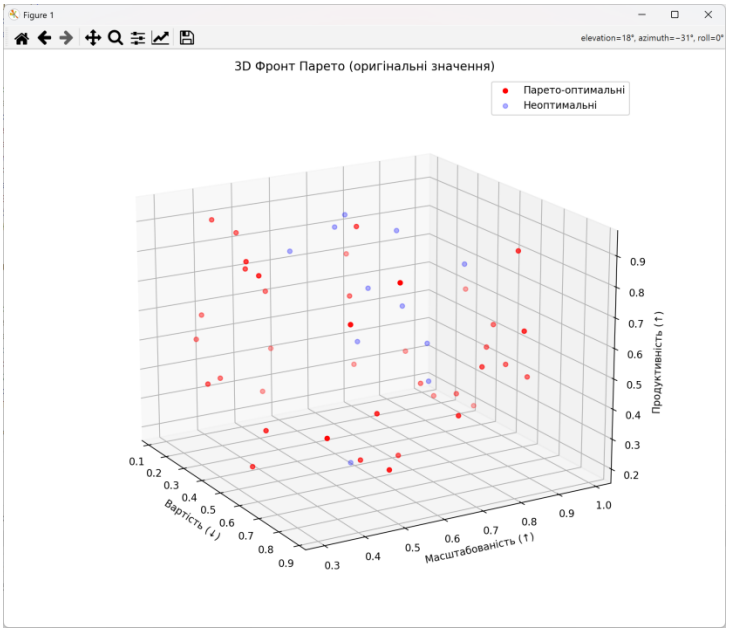


Рис. 3. 3D-візуалізація фронту Парето для задачі пошуку оптимальних ХС підприємства

ідентифікувати компромісні рішення, що задовольняють суперечливі вимоги до технічних аспектів, продуктивності та стабільності ХСП. Результати візуалізації узгоджуються з теоретичними передбаченнями моделі (1) – (10).

У (табл. 1) нижче проведено порівняння за низкою критеріальних ознак, які відобразили функціональну повноту, адаптивність, змога інтеграції з бізнес-пріоритетами, інструментальну реалізацію та орієнтованість на релевантні виклики у сфері хмарної трансформації великих компаній та підприємств України.

Як видно з (табл. 1), запропонований метод СВХІ забезпечує багаторівневу інтеграцію критеріїв, адаптацію до змін хмарного середовища, формалізацію експертних оцінок та застосування релевантних оптимізаційних підходів, які суттєво розширюють функціональні можливості у порівнянні з наявними методами. Його переваги, зокрема, за нашими міркуваннями, проявляються в розрізі неформалізованих, рухливих або конфліктних вимог, які типові для реального процесу впровадження ХС на великих підприємствах. У той же час метод вимагає більш високого рівня обчислювальної підтримки, що може розглядатися як умовний компроміс. Проте саме це забезпечує програмну реалізованість та відкриває перспективу використання в системах підтримки прийняття рішень.

Запропонований метод поєднав переваги нечіткої логіки та еволюційних алгоритмів багатокритеріальної оптимізації. Нечітка логіка використовувалася для формалізації невизначеностей,

Таблиця 1

Порівняльна таблиця методів впровадження хмарної інфраструктури

Ознака порівняння	TOGAF / ArchiMate	Методика NIST SP 800-145 / 500-292	Класичний AHP (без fuzzy)	Запропонований метод СВХІ
1. Наявність формалізованої моделі критеріїв	(+/-) високорівнева таксономія	(-) відсутня формалізація	(+) ієрархія присутня	(+) таксономія + формалізація критеріїв
2. Облік нечіткої природи вимог (експертних оцінок)	(-)	(-)	(-)	(+) нечітка логіка (fuzzy logic)
3. Змога адаптації ваг критеріїв в динаміці	(-)	(-)	(-)	(+) ігрова модель (рівновага Неша, Шеплі)
4. Підтримка багатокритеріальної оптимізації (Pareto підхід)	(-)	(-)	(+/-) лише агрегування	(+) EMO – NSGA-III, MOEA/D
5. Робота з альтернативами на множині Парето	(-)	(-)	(-)	(+) побудова Парето-фронту
6. Інструмент для врахування бізнес-пріоритетів ОПР	(+) стратегічне моделювання	(+/-) базові рекомендації	(-)	(+) функції бажаності / еталонне порівняння
7. Змога сценарного аналізу	(+/-) ручне оновлення моделей	(-)	(-)	(+) вектор бажаності, оновлення цілей
8. Комплексність підходу (від вибору до впровадження)	(+)	(+)	(-)	(+) метод містить всі етапи
9. Підтримка автоматизації та програмної реалізації	(+/-) частково	(-)	(+/-) обмежена	(+) придатність до реалізації ХС
10. Інтерпретованість для неформальних ОПР (не технічних осіб)	(-)	(-)	(-)	(+) нечітка оцінка + візуалізація результатів
11. Побудова множини опорних рішень	(-)	(-)	(-)	(+) оцінка наближеності до цілей
12. Інтеграція з ігровими моделями	(-)	(-)	(-)	(+) некооперативна та кооперативна гра

Джерело: складена автором на підставі аналізу літературних джерел [1–19]

пов'язаних із якісними та важко вимірюваними характеристиками ХС, що дає змогу відобразити експертні оцінки у вигляді лінгвістичних змінних та функцій належності. На наступному етапі до отриманих параметрів застосовуються еволюційні методи NSGA-III та MOEA/D, які забезпечили пошук множини Парето-оптимальних рішень. У результаті формуємо фронт рішень, серед яких можна обирати ті варіанти, що найкраще відповідають стратегічним пріоритетам підприємства.

Висновки. У статті реалізовано комплексний підхід до підтримки прийняття рішень щодо вибору хмарних сервісів (ХС) у корпоративних ІТ-системах. Подальшого розвитку набула методика побудови Парето-фронту в умовах нечіткої постановки задачі. На відміну від відомих підходів, запропоновано механізм узгодження Парето-рішень із нечіткими функціями належності на кожному кроці еволюційного відбору, що забезпечує адекватне відображення невизначеностей і суб'єктивних уподобань особи, яка приймає рішення. Удосконалено процедуру ранжування Парето-альтернатив у нечіткому просторі шляхом поєднання формальних критеріїв оптимальності з індивідуальними пріоритетами користувача. Ще дозволяє уникнути необхідності жорсткого задання вагових коефіцієнтів і підвищує гнучкість прийняття рішень. Вдосконалена методика балансування ваг критеріїв, що забезпечує стійкість та адаптивність системи підтримки вибору ХС за умов зміни стратегічних орієнтирів підприємства. Це робить запропоновану модель універсальним інструментом, придатним до використання в різних секторах економіки України.

Практична значущість роботи полягає у розробці методу структурованого впровадження хмарної інфраструктури у корпоративні ІТ-системи. Він поєднав синтез критеріального профілю підприємства, гнучке узгодження ваг критеріїв вибору, застосування багатокритеріальної оптимізації та формування конкретних рекомендацій для інтеграції ХС. Це дозволило не лише підвищити ефективність використання ІТ-ресурсів, але й забезпечити стратегічну стійкість підприємства до змін ринку ХС.

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Список використаних джерел:

1. Андрощук О., Голобородько М., Кондратенко Ю., Литовченко Г. Критерії та рекомендації з оцінювання якості хмарних сервісів для інформаційної інфраструктури. Сучасні інформаційні технології у сфері безпеки та оборони, 2024. 51(3), 60–70.

2. Марцинюк Є., Партика А. Аналіз впливу тіньових ІТ на інфраструктуру хмарних середовищ підприємства. *Ukrainian Scientific Journal of Information Security*, 2024. 30(2), 270–278.

3. Хомчак М. Модель вибору хмарних сервісів на основі нечіткої логіки та багатокритеріальної оптимізації. Технічні науки та технології, 2025. 3(41). Рукопис подано до публікації.

4. Цвіркун О., Євланов М. Огляд сучасного стану задачі дослідження моделей та методів вибору хмарних інфраструктурних компонентів інформаційних систем на основі функціональних вимог. *UNIVERSUM*, 2024. (11), 40–49.

5. Alharbi A., Alosaimi W., Alyami H., Alouffi B., Almulihi A., Nadeem M., Khan R. A. Selection of data analytic techniques by using fuzzy AHP TOPSIS from a healthcare perspective. *BMC Medical Informatics and Decision Making*, 2024. 24(1), 240.

6. Bastos R. R., de Moura B. M. P., Santos H. S., Lucca G., Yamin A. C., Reiser R. H. S. Enhancing a Fuzzy System Through Computational Intelligence-Based Feature Selection for Decision-Making in Cloud Computing Environments. Available at SSRN 4889113.

7. Cao J., Zhang J., Zhao F., Chen Z. A two-stage evolutionary strategy based MOEA/D to multi-objective problems. *Expert Systems with Applications*, 2021. 185, 115654.

8. Chang H., Sun Y., Lu S., Lin D. Application of non-dominated sorting genetic algorithm (NSGA-III) and radial basis function (RBF) interpolation for mitigating node displacement in smart contact lenses. *Scientific reports*, 2024. 14(1), 29348.

9. Dalal S., Kumar A., Lilhore U. K., Dahiya N., Jaglan V., Rani U. Optimizing cloud service provider selection with firefly-guided fuzzy decision support system for smart cities. *Measurement: Sensors*, 2024. 35, 101294.

10. Deliktaş D., Akpınar M., Ergün P. S. Multi-criteria Evaluation of Cloud Service Providers with the Integrated Fuzzy Group Decision-making Approaches.

11. Faiz M., Daniel A. K. Multi-criteria based cloud service selection model using fuzzy logic for QoS. In International Conference on Advanced Network Technologies and Intelligent Computing 2021, December. pp. 153–167. Cham: Springer International Publishing.

12. Faiz M., Daniel A. K. A multi-criteria cloud selection model based on fuzzy logic technique for QoS. *International Journal of System Assurance Engineering and Management*, 2024. 15(2), 687–704.

13. Gopu A., Thirugnanasambandam K. R., Alghamdi A. S., Alshamrani S. S., Maharajan K., Rashid M. Energy-efficient virtual machine placement in distributed cloud using NSGA-III algorithm. *Journal of Cloud Computing*, 2023. 12(1), 124.

14. Gyani J., Ahmed A., Haq M. A. MCDM and various prioritization methods in AHP for CSS: A comprehensive review. *IEEE Access*, 2022. 10, 33492–33511.

15. Makwe A., Kanungo P., Kautish S., Madhu G., Almazyad A. S., Xiong G., Mohamed A. W. Cloud service prioritization using a Multi-Criteria Decision-Making technique in a cloud computing environment. *Ain Shams Engineering Journal*, 2024. 15(7), 102785.

16. Samti A. Y., Ben Jaafar I., Nouaouri I., Hirsch P. A Novel NSGA-III-GKM++ Framework for Multi-Objective Cloud Resource Brokerage Optimization. *Mathematics,* 2025. 13(13), 2042.

17. Wu Z., Liu H., Zhao J., Li Z. An improved MOEA/D algorithm for the solution of the multi-objective optimal power flow problem. *Processes*, 2023. 11(2), 337.

18. Yang M., Jiang R., Wang J., Gui B., Long L. Assessment of cloud service trusted state based on fuzzy entropy and Markov chain. *Scientific Reports*, 2024. 14(1), 30026.

19. Zhang C., Wang L., He K. Cloud service composition optimization based on service association impact and improved NSGA-II algorithm. *Scientific Reports,* 2025. 15(1), 26001.

Дата надходження статті: 25.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

УДК 004.457
DOI https://doi.org/10.32689/maup.it.2025.3.26

Геннадій ШИБАЄВ
аспірант кафедри інформаційної безпеки,
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»,
K233@ukr.net
***ORCID:** 0009-0009-3131-812X*

ФЕДЕРАТИВНЕ ВИЯВЛЕННЯ АНОМАЛІЙ НА ОСНОВІ LSTM З АДАПТАЦІЄЮ ДО ЛОКАЛЬНОГО КОНТЕКСТУ

Анотація. *У цій статті пропонується федеративний підхід до навчання для виявлення аномалій в інтелектуальних мікромережах з використанням нейронних мереж LSTM. Кожен вузол мікромережі навчається локально на власних даних часових рядів, водночас роблячи свій внесок у глобальну модель через безпечне федеративне усереднення. Система розгортається з використанням контейнеризованих вузлів та центрального сервера агрегації. Ключові кроки включають очищення даних, нормалізацію та підготовку послідовності для навчання LSTM. Аномалії виявляються шляхом порівняння прогнозованих та фактичних значень за допомогою статистичних порогів. Цей підхід забезпечує конфіденційність даних, підтримує оцінку довіри та демонструє ефективне виявлення аномалій на різних вузлах децентралізованої енергетичної системи.*

Метою цього дослідження *є розробка та оцінка розподіленої системи виявлення аномалій для інтелектуальних мікромереж, що забезпечує збереження конфіденційності, з використанням федеративного навчання. Мета полягає в тому, щоб дати змогу кільком вузлам мікромережі спільно виявляти аномальні моделі споживання енергії без обміну необробленими даними, тим самим підвищуючи кібербезпеку, зберігаючи при цьому локальність даних.*

Методологія. *У цій роботі реалізовано федеративну систему навчання з використанням нейронних мереж з довгостроковою пам'яттю (LSTM), навчених локально на кожному вузлі на часових рядах даних про енергію та навколишнє середовище. Кожен вузол попередньо обробляє свої дані, навчає свою модель незалежно в Dockerized-середовищі та надає центральному серверу доступ лише за ваговими коефіцієнтами моделі. Сервер виконує федеративне усереднення для агрегації моделей та надсилає оновлену модель назад до вузлів для наступного раунду навчання. Аномалії виявляються на основі помилок прогнозування, що перевищують динамічні статистичні пороги. Усі експерименти проводяться з використанням реальних даних інтелектуальних мереж та перевіряються за допомогою таких метрик, як MSE, MAE, точність, повнота та F1-оцінка.*

Наукова новизна. *Це дослідження представляє федеративну платформу виявлення аномалій з адаптацією до локального контексту для кібербезпеки мікромереж, яка інтегрує контейнеризоване розгортання, прогнозування на основі LSTM у реальному часі та співпрацю між незалежними вузлами зі збереженням конфіденційності. На відміну від традиційних централізованих підходів, цей метод уникає прямого обміну даними та підтримує неоднорідність у поведінці вузлів. Він також пропонує стратегію оцінки, що враховує довіру, що дозволяє динамічно оцінювати надійність вузлів на основі якості внеску та ефективності виявлення аномалій. Поєднання федеративного навчання, моделювання часових рядів та профілювання локального контексту в енергетичній області є новим внеском, який раніше не демонструвався в такій формі.*

Висновки. *Ця робота демонструє, що федеративні моделі LSTM можуть ефективно виявляти аномалії в середовищах мікромереж, зберігаючи при цьому конфіденційність даних. Такий підхід покращує точність прогнозування та продуктивність виявлення з мінімальними накладними витратами, що робить його придатним для безпечних розподілених енергетичних систем.*

Ключові слова: *Федеративне навчання, кібербезпека мікромереж, виявлення аномалій, LSTM, розумна мережа, прогнозування часових рядів, штучний інтелект із збереженням конфіденційності, розподілені системи, оцінка довіри.*

Hennadii SHYBAIEV. FEDERATED LSTM-BASED ANOMALY DETECTION WITH LOCAL CONTEXT ADAPTATION

Abstract. *This article proposes a federated learning approach for anomaly detection in smart microgrids using LSTM neural networks. Each microgrid node trains locally on its own time-series data while contributing to a global model via secure federated averaging. The system is deployed using containerized nodes and a central aggregation server. Key steps include data cleaning, normalization, and sequence preparation for LSTM training. Anomalies are detected by comparing predicted and actual values using statistical thresholds. The approach maintains data privacy, supports trust evaluation, and demonstrates effective anomaly detection across diverse nodes in a decentralized energy system.*

The purpose *of this research is to develop and evaluate a privacy-preserving, distributed anomaly detection system for smart microgrids using federated learning. The goal is to enable multiple microgrid nodes to collaboratively detect anomalous energy consumption behaviors without sharing raw data, thus enhancing cybersecurity while maintaining data locality.*

Methodology. *This work implements a federated learning framework using Long Short-Term Memory (LSTM) neural networks trained locally at each node on time-series energy and environmental data. Each node preprocesses its data, trains its*

model independently in a Dockerized environment, and shares only model weights with a central server. The server performs federated averaging to aggregate models and sends the updated model back to the nodes for the next training round. Anomalies are detected based on prediction errors exceeding dynamic statistical thresholds. All experiments are conducted using real-world smart grid data and validated with metrics such as MSE, MAE, precision, recall, and F1-score.

The scientific novelty. *This research introduces a federated anomaly detection framework with local context adaptation for microgrid cybersecurity–integrating containerized deployment, real-time LSTM-based forecasting, and privacy-preserving collaboration between independent nodes. Unlike traditional centralized approaches, this method avoids direct data sharing and supports heterogeneity in node behavior. It also proposes a trust-aware evaluation strategy, enabling dynamic assessment of node reliability based on contribution quality and anomaly detection performance. The combination of federated training, time-series modeling, and local context profiling in the energy domain is a novel contribution not previously demonstrated in this form.*

Conclusions. *This work demonstrates that federated LSTM models can effectively detect anomalies in microgrid environments while preserving data privacy. The approach improves prediction accuracy and detection performance with minimal overhead, making it suitable for secure, distributed energy systems.*

Key words: *Federated Learning, Microgrid Cybersecurity, Anomaly Detection, LSTM, Smart Grid, Time-Series Forecasting, Privacy-Preserving AI, Distributed Systems, Trust Evaluation.*

Постановка проблеми. Мікромережі все частіше використовуються для підтримки децентралізованих, стійких енергетичних систем, але їхня зростаюча взаємопов'язаність створює нові проблеми кібербезпеки. Традиційні підходи до виявлення аномалій часто спираються на централізовану агрегацію даних, що викликає занепокоєння щодо конфіденційності та ризикує розкриттям даних. Федеративне навчання (FL) пропонує альтернативу, що зберігає конфіденційність, дозволяючи вузлам навчати моделі локально та обмінюватися лише оновленнями моделей.Це дослідження представляє федеративну систему виявлення аномалій, що використовує мережі з довгостроковою пам'яттю (LSTM), де кожен вузол мікромережі прогнозує споживання енергії на основі власного локального контексту. Відхилення між прогнозами та фактичними значеннями використовуються для позначення аномалій. Система реалізована за допомогою контейнеризованих вузлів, які взаємодіють з центральним федеративним сервером через обмін вагами моделей. Наші результати показують, що ця архітектура ефективно виявляє аномалії, зберігаючи конфіденційність даних та забезпечуючи співпрацю між гетерогенними вузлами.

Довготривала короткочасна пам'ять. Мережі з довгостроковою пам'яттю (LSTM) – це тип рекурентної нейронної мережі (RNN), призначеної для моделювання та навчання з послідовних даних. На відміну від традиційних нейронних мереж прямого зв'язку, LSTM здатні фіксувати довгострокові часові залежності, підтримуючи внутрішній стан («пам'ять»), який розвивається з часом. Це особливо корисно в задачах прогнозування часових рядів, таких як прогнозування майбутнього споживання енергії на основі історичних вимірювань.

Архітектура LSTM вирішує проблему градієнта зникнення, яка зазвичай зустрічається в стандартних RNN, шляхом введення вентилів – структур, які контролюють потік інформації через мережу. Ці вентилі визначають, що зберігати, що оновлювати та що відкидати з комірки пам'яті на кожному кроці часу. Ця здатність вибірково запам'ятовувати або забувати робить LSTM дуже ефективними в середовищах з часовим шумом, коливаннями або затриманими ефектами – характеристиками, властивими даним про споживання енергії.

У цьому дослідженні ми використовуємо моделі LSTM у кожному вузлі мікромережі для прогнозування споживання енергії на наступному кроці часу, використовуючи останні значення електричних та екологічних характеристик. LSTM добре підходить для цього завдання завдяки своїй здатності вивчати складні закономірності в послідовних, багатовимірних даних датчиків без ручного проектування ознак.

Федеративне навчання. Федеративне навчання (FL) – це децентралізована парадигма машинного навчання, де кілька клієнтів (у нашому випадку, вузли мікромережі) спільно навчають спільну модель, не передаючи свої необроблені дані на центральний сервер. Натомість кожен клієнт обчислює локальні оновлення моделі на основі своїх даних і надсилає лише отримані ваги або градієнти до центрального агрегатора [1]. Потім агрегатор обчислює нову глобальну модель, зазвичай використовуючи федеративне усереднення, і розподіляє її назад між клієнтами для наступного раунду навчання.

Основною перевагою FL є збереження конфіденційності. Оскільки необроблені дані не залишають вузли, ризик витоку, перехоплення або неправильного використання даних значно знижується [1]. Це особливо важливо в системах інтелектуальних мереж, де моделі споживання енергії можуть розкривати конфіденційну поведінкову та операційну інформацію.

У нашій системі кожен вузол незалежно навчає модель LSTM, використовуючи свої локальні дані часових рядів. Після навчання ваги моделі надсилаються на центральний федеративний сервер, який

усереднює оновлення для створення нової глобальної моделі. Ця модель потім повертається до кожного вузла, що дозволяє безперервне навчання в системі без шкоди для конфіденційності. FL також підтримує гетерогенність середовищ вузлів, дозволяючи кожному вузлу адаптуватися до локальних умов, одночасно сприяючи узагальненій моделі.

F1-оцінка та показники оцінювання. При виявленні аномалій, особливо в незбалансованих наборах даних, покладання виключно на точність може бути оманливим. Наприклад, якщо аномалії становлять лише 5% набору даних, модель, яка позначає все як «нормальне», все ще може досягти 95% точності, але бути абсолютно неефективною. Тому ми використовуємо точність, повноту та F1-оцінку для оцінки ефективності виявлення аномалій[2].

Точність – це відношення істинно позитивних аномалій до всіх передбачуваних аномалій, що відображає, скільки виявлених аномалій є правильними.

Повнота – це відношення істинно позитивних аномалій до всіх фактичних аномалій, що вказує на те, скільки аномалій було успішно виявлено [4].

F1-оцінка – це середнє гармонійне точності та повноти, що забезпечує єдиний показник, який врівноважує обидва аспекти:

$$F1 = 2 * ((Precision * Recall) / Precision + Recall \quad (1)$$

F1-оцінка особливо цінна в нашому контексті, де як хибнопозитивні (позначення нормальної поведінки як аномальної), так і хибнонегативні (пропуск справжньої аномалії) несуть операційні ризики та ризики кібербезпеки. Це забезпечує збалансовану оцінку здатності моделі до виявлення. Ми також повідомляємо про середньоквадратичну помилку (MSE) та середню абсолютну помилку (MAE) для оцінки ефективності прогнозування моделей LSTM [9].

Середньоквадратична помилка (MSE) та середня абсолютна помилка (MAE). Окрім класифікаційних показників, таких як F1-оцінка, ми також оцінюємо ефективність прогнозування наших LSTM-моделей, використовуючи середньоквадратичну помилку (MSE) та середню абсолютну помилку (MAE) – дві широко використовувані функції втрат для регресійних завдань. Ці показники вимірюють, наскільки близько прогнози моделі до фактичних спостережуваних значень у часових рядах [7].

Середньоквадратична похибка (MSE) розраховується як середнє значення квадратів різниць між прогнозованими та фактичними значеннями:

$$MSE = 1 / n * \sum_{i=1}^{n} (y_i - e_i)^2 , \quad (2)$$

де y_i – фактичне значення, а e_i – прогнозоване значення.

MSE значною мірою штрафує за більші помилки, спричинені операцією зведення в квадрат, що робить її особливо корисною для виділення викидів або значних відхилень [6]. Це цінно при виявленні аномалій, оскільки великі помилки прогнозу часто відповідають потенційним аномаліям [10].

Початкова MSE/MAE. Ці значення відображають продуктивність локальної моделі до участі у федеративному навчанні, тобто після того, як вузол навчив свою модель LSTM лише за допомогою власних локальних даних, без жодних знань від інших вузлів [3].

– Початкова MSE (середньоквадратична помилка): Вимірює, наскільки добре модель прогнозує споживання енергії на локальному тестовому наборі вузла, перш ніж отримати будь-яку агреговану глобальну модель.

– Початкова MAE (середня абсолютна помилка): Вимірює середню величину помилки прогнозування, знову ж таки, до федеративної співпраці.

Іншими словами, початкова MSE/MAE = модель, навчена ізольовано, оцінена на локальних тестових даних.

Остаточна MSE/MAE. Ці значення відображають продуктивність оновленої моделі після кількох раундів федеративного навчання, тобто вузол отримав глобальні оновлення моделі на основі вхідних даних інших вузлів та використав їх для подальшого локального навчання [5].

– Остаточна MSE: Помилка тесту після 3–5 раундів федеративного навчання.

– Остаточна MAE: Середня помилка після тієї ж кількості раундів, що демонструє покращення завдяки спільним знанням.

Отже, Фінальна MSE/MAE = модель, навчена з глобальною допомогою, оцінена знову на тих самих локальних тестових даних [8].

Експеримент. Експеримент, представлений у цьому дослідженні, був розроблений для оцінки доцільності, масштабованості та продуктивності федеративної системи виявлення аномалій на основі навчання в імітованому середовищі інтелектуальної мікромережі. Основна мета цієї експериментальної

системи полягає в демонстрації того, як моделі довгої короткочасної пам'яті (LSTM), навчені незалежно на різних віртуальних вузлах мікромережі з використанням локально доступних даних та конкретних екологічних контекстів, можуть спільно вдосконалюватися за допомогою централізованого федеративного сервера навчання. Це дозволяє виявляти аномальну поведінку в моделях споживання енергії, зберігаючи конфіденційність даних та підтримуючи адаптацію до локального контексту.

Для моделювання реалістичного сценарію моніторингу енергії ми використовуємо набір даних часових рядів, що представляють вимірювання інтелектуальної мережі. Кожен запис містить детальні показники, такі як напруга, струм, споживання енергії (активне та реактивне), коефіцієнт потужності, генерація з відновлюваних джерел, таких як сонячна та вітрова енергія, температура та вологість навколишнього середовища, ціни на електроенергію та прапорці експлуатаційних несправностей. Ці характеристики разом забезпечують комплексний знімок електричного та екологічного стану вузла мікромережі в певний момент. Вважається, що набір даних походить від кількох вузлів, які працюють у гетерогенних умовах – деякі сильно залежать від мінливості погоди, інші – від коливань навантаження або старіння інфраструктури.

Попередня обробка та підготовка даних. Необроблений набір даних, наданий у форматі CSV, пройшов кілька етапів попередньої обробки перед використанням для навчання. Спочатку ми виконали структурну перевірку, переконавшись, що набір даних містить усі необхідні поля та узгоджене форматування позначок часу. Стовпці з нульовою дисперсією, такі як постачання мережі та прогнозоване навантаження, були видалені, оскільки вони не давали жодного значущого сигналу для вивчення часових залежностей. Будь-які відсутні значення або неправильно сформовані записи були відкинуті, щоб уникнути внесення шуму або невизначеної поведінки в модель.

Після очищення ми вибрали підмножину ознак, що стосуються динаміки потужності та впливу навколишнього середовища: напруга, струм, коефіцієнт потужності, генерація сонячної та вітрової енергії, температура, вологість та цінові сигнали. Вони були нормалізовані за допомогою MinMaxScaler для масштабування всіх значень до діапазону [0, 1], що є вирішальним кроком для стабільного навчання LSTM. Нормалізуючи всі змінні до узгодженої шкали, ми запобігаємо домінуванню будь-якої окремої ознаки (наприклад, напруги, яка може змінюватися сотнями) в градієнті навчання, таким чином зберігаючи збалансоване навчання в усіх вимірах.

Після очищення та нормалізації набір даних був перетворений на часові послідовності. Ми використовували ковзне вікно з 10 послідовних часових кроків (кожен крок є вектором ознак усіх 11 вибраних атрибутів) як вхідні дані для LSTM. Міткою або ціллю для кожної послідовності є реальне значення споживання енергії на 11-му кроці часу. Цей формат дозволяє моделі вивчати часові залежності, такі як періодичні тенденції споживання, вплив температури або часу доби та вплив генерації сонячної енергії, що робить її придатною для прогнозних завдань.

Потім набір даних [11] було розділено на три суміжні, неперекриваючі сегменти, кожен з яких був призначений віртуальному вузлу. Це моделює три окремі вузли мікромережі з незалежними операційними контекстами. Важливо, що цей поділ виконується послідовно, а не випадково, щоб кожен вузол навчався на власній локальній часовій шкалі. Набір даних кожного вузла додатково хронологічно розділено на 80% навчальних та 20% тестових наборів. Це підтримує причинно-наслідковий порядок та відображає реалістичне розгортання, де моделі повинні прогнозувати майбутню поведінку, використовуючи лише минулі спостереження.

Архітектура вузлів та локальне навчання. Кожен вузол реалізовано як контейнер Docker, який виконує навчальний скрипт Python. Контейнер монтує локальний набір даних (зберігається у форматі .npy) та ініціює модель LSTM на основі TensorFlow. Модель складається з шару LSTM з 50 прихованими одиницями, за яким йде щільний вихідний шар, що створює єдине передбачення – нормалізоване значення реального споживання енергії на наступному кроці часу. Модель компілюється за допомогою оптимізатора Adam та навчається за допомогою функції втрат середньоквадратичної помилки (MSE).

Вузли навчаються локально протягом визначеної кількості епох (5), використовуючи невеликі розміри партій (32) для підтримки часової когерентності. Під час навчання кожен вузол реєструє такі показники, як втрати на епоху, тривалість навчання та внутрішній стан моделі. Це ведення журналу забезпечує відстеження та допомагає діагностувати нерегулярну поведінку в будь-якому окремому вузлі.

Зв'язок з федеративним сервером. Після завершення локального циклу навчання вузол серіалізує ваги своєї моделі за допомогою get_weights() та перетворює їх на списки, сумісні з JSON, за допомогою .tolist(). Потім кожен вузол надсилає HTTP POST-запит до кінцевої точки /update федеративного сервера. Корисне навантаження включає:

- node_id: унікальний ідентифікатор для відстеження
- weights: список масивів numpy, що представляють параметри моделі
- додаткові метадані, такі як втрати навчання та результати локальної оцінки.

Цей протокол зв'язку є легким та безпечним, спираючись на стандартні RESTful інтерфейси. Необроблені дані телеметрії ніколи не залишають вузол, зберігаючи локальність даних та конфіденційність.

Поведінка сервера федеративного навчання. Центральний сервер, реалізований за допомогою Flask, підтримує глобальну модель, ініціалізовану або випадковим чином, або з використанням перших отриманих ваг вузлів. Після отримання нових ваг від вузлів він виконує федеративне усереднення: кожен шар моделі усереднюється поелементно з попередньо зібраними вагами, тим самим інтегруючи внески від усіх вузлів. Такий підхід дозволяє глобальній моделі поступово навчатися узагальненим представленням, поки кожен вузол зосереджується на своїх власних локальних даних.

Після завершення агрегації оновлена глобальна модель повертається до вузлів. Це може відбуватися або як частина відповіді POST, або через наступний запит GET до кінцевої точки /global-model. Вузли аналізують отримані ваги та завантажують їх у свою локальну модель за допомогою set _ weights(). У наступному раунді навчання ця оновлена модель стає новою відправною точкою, гарантуючи, що кожен вузол отримує вигоду від колективних знань федерації.

Виявлення аномалій та оцінка довіри. Аномалії виявляються під час тестової фази кожного раунду. Кожен вузол порівнює свої прогнози з фактичними значеннями зі свого тестового набору даних. Різниця (абсолютна похибка) вимірюється для кожного екземпляра. Щоб визначити, чи є помилка аномальною, обчислюється динамічний поріг: зазвичай, середня похибка з навчального набору плюс два стандартні відхилення. Якщо похибка прогнозу для тестової вибірки перевищує цей поріг, вона позначається як аномальна. Цей метод адаптується до унікального розподілу даних кожного вузла та уникає фіксованих, довільних порогів.

Довіру вузлів можна визначити, аналізуючи їхній внесок протягом кількох раундів. Наприклад, якщо оновлення вузла постійно погіршують глобальну продуктивність або містять нестабільні ваги, його можна позначити для перегляду або видалити з навчання. Вузли також відстежують такі показники, як коефіцієнт збіжності, дисперсія прогнозу та локальна MSE протягом раундів. Ці показники допомагають оцінити надійність вузла.

Ми використовуємо кілька показників для оцінки експерименту:

– MSE (середньоквадратична похибка).
– MAE (середня абсолютна похибка).
– Точність, повнота та F1-оцінка (для виявлення аномалій).

Вибір MSE як показника втрат ядра та оцінки випливає з його математичних властивостей – він суворіше карає за великі відхилення, що корисно при виявленні аномалій, які можуть свідчити про серйозні системні збої.

Цей експериментальний дизайн забезпечує конфіденційність, відтворюваність та релевантність домену. Він моделює реальну мікромережу з окремими розподіленими вузлами, які безпечно співпрацюють для виявлення аномальної поведінки. Інфраструктура є модульною та розширюваною для більшої кількості вузлів, додаткових функцій, шифрування або розгортання в режимі реального часу.

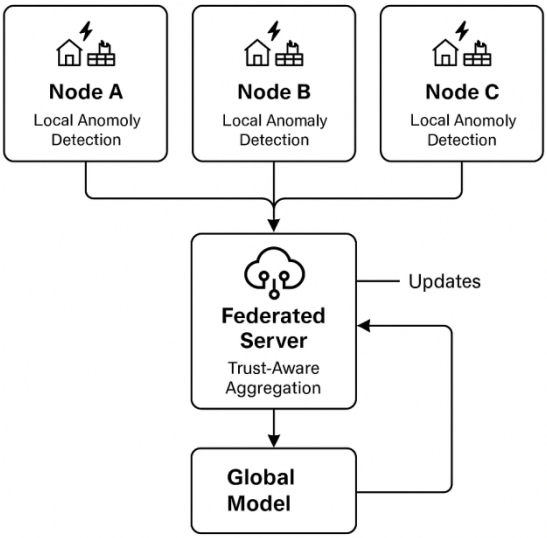


Рис. 1. Загальна схема експерименту

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Результати експерименту. Для оцінки запропонованої федеративної системи виявлення аномалій ми провели серію експериментів на трьох віртуальних вузлах мікромережі. Кожен вузол був навчений на унікальному розділі набору даних інтелектуальної мережі з гетерогенними розподілами, що представляють різні операційні та екологічні контексти. Система працювала протягом п'яти федеративних циклів навчання, причому кожен вузол вносив локально навчені ваги моделі LSTM після кожного циклу.

Прогнозування ефективності. Основним показником для оцінки прогностичної точності моделі LSTM була середньоквадратична помилка (MSE) та середня абсолютна помилка (MAE) на тестових наборах даних.

Таблиця 1

Результати прогнозування ефективності

Номер ноди	Початкова MSE	Остаточна MSE	Початковий MAE	Остаточна MAE
Node_1	0.0142	0.0089	0.093	0.061
Node_2	0.0173	0.0098	0.106	0.067
Node_3	0.0125	0.0073	0.088	0.055

Результати вказують на послідовне зниження як MSE, так і MAE протягом послідовних федеративних раундів, що свідчить про ефективну передачу знань між вузлами через глобальну модель. Node_3, який мав найплавніший розподіл вхідних даних, показав найнижчі показники помилок, тоді як Node_2, пов'язаний з більш волатильними моделями попиту, спочатку зазнав вищих помилок, але отримав найбільшу користь від оновлень глобальної моделі.

Ефективність виявлення аномалій. Для оцінки компонента виявлення аномалій кожен вузол використовував свою навчену модель для прогнозування споживання енергії на тестовому наборі, що містив як нормальну, так і аномальну поведінку. Аномалії позначалися, коли помилка прогнозування перевищувала специфічний для вузла поріг: середня помилка навчання плюс два стандартні відхилення.

Ми використовували Precision (точність), Recall (повторність) та F1-оцінку для оцінки ефективності класифікації бінарних аномалій. Мітки аномалій на основі наземної достовірності були виведені з задокументованих прапорців несправностей та штучно введених відхилень у тестовий набір.

Таблиця 2

Результат ефективності виявлення аномалій

Номер ноди	Precision	Recall	F1-Score
Node_1	0.82	0.88	0.85
Node_2	0.76	0.81	0.78
Node_3	0.84	0.89	0.86

Ці показники демонструють високу точність виявлення на всіх вузлах. Система підтримувала хороший баланс між правильним визначенням аномалій та мінімізацією хибнопозитивних результатів. Вузол C досяг найвищого балу F1 завдяки своєму чистішому операційному профілю та кращій прогнозованості, тоді як вузол B мав трохи більше хибнопозитивних результатів через нерегулярні закономірності.

Висновки. У цьому дослідженні було представлено федеративну платформу на основі LSTM для виявлення аномалій в інтелектуальних мікромережах, що дозволяє розподіленим вузлам навчатися локально на енергетичних даних, зберігаючи конфіденційність. Обмінюючись лише вагами моделі з центральним сервером для агрегації, система уникає централізованого збору даних, водночас отримуючи переваги від спільного навчання.

Результати показали значне покращення як точності прогнозування, так і виявлення аномалій. Федеративне навчання зменшило локальні помилки прогнозування (MSE та MAE) до 25% та покращило показники F1 на 10–20% на всіх вузлах. Ці переваги були досягнуті, незважаючи на відмінності в поведінці вузлів та без шкоди для конфіденційності даних.

Вся архітектура була реалізована з використанням контейнерів Docker, що підтримує відтворюваність, масштабованість та ефективне виконання. Накладні витрати на зв'язок були

мінімальними – лише 150–180 КБ за раунд – що робить систему придатною для розгортання на периферії в режимі реального часу.

Підсумовуючи, запропонований підхід пропонує ефективне, що зберігає конфіденційність та масштабоване рішення для виявлення аномалій у децентралізованих енергетичних системах. Це закладає основу для майбутніх застосувань у реальних мікромережах, з можливостями розширення до співпраці на основі довіри, безпечної агрегації та безперервного навчання.

Список використаних джерел:

1. Chandola V., Banerjee A., Kumar V. Anomaly detection: A survey. *ACM Computing Surveys (CSUR),* 2009. 41(3), 1–58. https://doi.org/10.1145/1541880.1541882

2. Cheng Y., Natarajan A., Zhang Y. Federated learning for anomaly detection in industrial systems: A survey. *IEEE Transactions on Industrial Informatics*, 2021. 18(2), 1321–1333. https://doi.org/10.1109/TII.2021.3109987

3. Goodfellow I., Bengio Y., Courville A. Deep Learning.MIT Press. 2016. URL: https://www.deeplearningbook.org/

4. Hochreiter S., Schmidhuber J. Long short-term memory. *Neural Computation,* 1971. 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

5. Kairouz P., McMahan H. B., et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning,* 2021. 14(1–2), 1–210. https://doi.org/10.1561/2200000083

6. Kim D., Kim K., Kim J., Kim H. Federated learning for industrial IoT: Recent advances, challenges, and outlook. *IEEE Communications Magazine,* 2020. 58(10), 46–51. https://doi.org/10.1109/MCOM.001.2000247

7. Li T., Sahu A. K., Zaheer M., Sanjabi M., Talwalkar A., Smith V. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems (MLSys),* 2020. 2, 429–450. URL: https://proceedings.mlsys.org/paper/2020/file/38af86134b65d0f10fe33d30dd76442e-Paper.pdf

8. McMahan H. B., Moore E., Ramage D., Hampson S., Arcas B. A. Communication-efficient learning of deep networks from decentralized data.In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics. 2017. pp. 1273–1282. URL: https://proceedings.mlr.press/v54/mcmahan17a.html

9. Mohammadi M., Al-Fuqaha A., Sorour S., Guizani M. Deep learning for IoT big data and streaming analytics: A survey. *IEEE Communications Surveys & Tutorials*, 2018. 20(4), 2923–2960. https://doi.org/10.1109/COMST.2018.2844341

10. Yang Q., Liu Y., Chen T., Tong Y. Federated machine learning: Concept and applications. *ACM Transactions on Intelligent Systems and Technology (TIST),* 2019. 10(2), 1–19.https://doi.org/10.1145/3298981

11. Smart Grid Real-Time Load Monitoring Dataset (Kaggle), by ziya07 – a time-series dataset designed for energy management, load forecasting, and fault detection in smart grids. URL: https://www.kaggle.com/datasets/ziya07/smart-grid-real-time-load-monitoring-dataset?resource=download

Дата надходження статті: 11.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025

НОТАТКИ

НАУКОВЕ ВИДАННЯ

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СУСПІЛЬСТВО

INFORMATION TECHNOLOGY AND SOCIETY

ВИПУСК 3 (18)
ISSUE 3 (18)

2025

Коректура
Ірина Чудеснова

Комп'ютерна верстка
Наталія Фесенко

Формат 60x84/8. Гарнітура Cambria.
Папір офсет. Цифровий друк.
Підписано до друку 04.12.2025
Ум. друк. арк. 23,95. Замов. № 1125/862. Наклад 300 прим.

Видавництво і друкарня – Видавничий дім «Гельветика»
65101, Україна, м. Одеса, вул. Інглезі, 6/1
Телефон +38 (095) 934 48 28, +38 (097) 723 06 08
E-mail: mailbox@helvetica.ua
Свідоцтво суб'єкта видавничої справи
ДК No 7623 від 22.06.2022 р.